

Storytelling with Animated Interactive Objects in Real-time 3D Maps

Raimund Schnürer

2023



DISS. ETH NO. 29884

Storytelling with Animated Interactive Objects in Real-time 3D Maps

A thesis submitted to attain the degree of
DOCTOR OF SCIENCES
(Dr. sc. ETH Zurich)

presented by

Raimund Thies Schnürer

M.Sc. Geoinformatics, University of Münster
born on 16.04.1987

accepted on the recommendation of

Prof. Dr. Lorenz Hurni

Prof. Dr. Temenoujka Bandrova

Prof. Dr. A. Cengiz Öztireli

2023

Abstract

Digital story maps present geographic information embedded in a narrative structure and often supplemented by multimedia elements. The currently predominant extrinsic approach, however, impedes following the storyline since the reader's attention is split between maps and additional illustrative content in the user interface. Incoherent representations between abstract map elements and realistic multimedia elements can be seen as another shortcoming. To close these gaps, an intrinsically oriented approach, which is inspired by historical and contemporary pictorial maps, is proposed in this dissertation: Figurative objects, which act as storytellers, are introduced into the map. By spatially anchoring the complementary entities within 3D maps in particular, the connection between the cartographic model and geographic reality will be strengthened. The overall goal of this work is to automatically turn static objects from prevalent 2D pictorial maps into animated objects for interactive 3D story maps.

Artificial neural networks, primarily convolutional neural networks (CNNs), are applied in a sequence of discriminative and generative tasks to achieve the goal. For each task, data is prepared to train the networks in a supervised manner. Firstly, pictorial maps are identified from publicly available images on the internet by CNNs for classification since metadata of the images is not always present or reliable. Different strategies are investigated to input the images into the CNNs. Secondly, bounding boxes of objects on pictorial maps are detected using the example of sailing ships. Although map descriptions may include the occurrences of objects, their positions and sizes are usually unknown. To determine these two measures, CNNs for object detection are examined while modifying their hyperparameters. Thirdly, silhouettes of pictorial objects are recognised, exemplified by human figures. As the manual preparation of training data would be too labour-intensive, combinations of figurative and realistic entities are evaluated. Following, body parts and pose points are extracted from the silhouettes of the figures, which is a prerequisite for skeletal animations and the insertion of speech bubbles at head positions. CNNs with varying numbers of skip connections are compared for this task. Lastly, 3D figures are derived from their 2D counterparts based on the outputs of the previous step by a series of networks. This significantly facilitates and accelerates the 3D modelling process. The figures are represented by implicit surfaces, which are advantageous for curved surfaces, and rendered in real time by a ray tracing algorithm. Quantitative metrics, such as accuracy and rendering speed, and qualitative results are reported for each task.

The inferred 3D pictorial objects may guide readers through the map while providing background information and offering interaction possibilities like quizzes. This is especially suitable for atlases, touristic or educative applications, even in augmented and virtual reality. Animated interactive objects are intended to engage map readers through an immersive storytelling approach, increase their map literacy and frequency of use, and create long-lasting memories.

Zusammenfassung

Digitale Story Maps zeigen geografische Informationen, die in eine narrative Struktur eingebettet und meist mit multimedialen Elementen ergänzt sind. Der derzeit vorherrschende extrinsische Ansatz erschwert jedoch dem Handlungsverlauf zu folgen, da die Aufmerksamkeit des Lesenden zwischen Karten und illustrativen Zusatzinhalten in der Benutzeroberfläche geteilt ist. Ein weiterer Nachteil ist die inkohärente Darstellung zwischen abstrakten Kartenelementen und realistischen multimedialen Elementen. Um diese Lücken zu schliessen, wird in der vorliegenden Dissertation ein intrinsisch motivierter Ansatz vorgeschlagen, der von historischen und gegenwärtigen Bildkarten inspiriert ist: Figürliche Objekte, die eine Erzählfunktion ausüben, sollen der Karte hinzugefügt werden. Durch die räumliche Einbettung der zusätzlichen Entitäten in insbesondere 3D Karten wird die Verbindung zwischen kartografischem Modell und geografischer Realität gestärkt. Das Gesamtziel der Arbeit ist es, statische Objekte aus gängigen 2D Bildkarten automatisch in animierte Objekte für interaktive 3D Story Maps umzuwandeln.

Künstliche neuronale Netzwerke, vorwiegend Convolutional Neural Networks (CNNs), werden in einer Reihe von diskriminativen und generativen Aufgaben angewendet, um das Ziel zu erreichen. Für jede Aufgabe werden Daten aufbereitet, um die Netzwerke überwacht zu trainieren. Zuerst werden Bildkarten aus öffentlich zugänglichen Bildern im Internet durch klassifizierende CNNs identifiziert, da Metadaten der Bilder nicht immer vorhanden oder verlässlich sind. Hierbei werden verschiedene Vorgehensweisen verglichen, um die Bilder den CNNs zuzuführen. Als Zweites werden umschliessende Rechtecke von Objekten auf Bildkarten am Beispiel von Segelschiffen detektiert. Obwohl Objekte in Kartenbeschreibungen aufgeführt sein können, sind deren Positionen und Abmessungen oft nicht bekannt. Um diese beiden Angaben zu erhalten, werden CNNs zur Objekterkennung untersucht, wobei deren Hyperparameter modifiziert werden. Als Drittes werden Umriss von bildhaften Objekten, beispielsweise menschlichen Figuren, erkannt. Da die manuelle Aufbereitung der Trainingsdaten zu arbeitsaufwändig wäre, werden Kombinationen von figürlichen und realistischen Entitäten ausgewertet. Anschliessend werden Körperteile und Haltungspunkte von den Umrissen der Figuren extrahiert, was eine Voraussetzung für Skelettanimationen und das Einfügen von Sprechblasen an Kopfpositionen bildet. Bei dieser Aufgabe werden CNNs mit einer unterschiedlichen Anzahl von Brückenverbindungen untersucht. Zuletzt werden basierend auf den Ergebnissen des vorherigen Schritts 3D-Figuren von ihren 2D-Pendants mithilfe einer Reihe von Netzwerken abgeleitet. Dies vereinfacht und beschleunigt den 3D-Modellierungsprozess signifikant. Die Figuren werden durch implizite Flächen repräsentiert, welche für gekrümmte Flächen vorteilhaft sind, und mit einem Raytracing-Algorithmus in Echtzeit gerendert. Quantitative Metriken, zum Beispiel Genauigkeit und Rendering-Geschwindigkeit, und qualitative Ergebnisse werden bei jeder Aufgabe angeführt.

Die erzeugten bildlichen 3D-Objekte können Lesende durch die Karte führen, während sie Hintergrundinformationen liefern und Interaktionsmöglichkeiten wie Ratespiele anbieten. Dies ist speziell für Atlanten, touristische oder pädagogische Anwendungen geeignet, sogar in Augmented oder Virtual Reality. Die animierten interaktiven Objekte sollen Lesende durch immersives Storytelling ermuntern, deren Kartenkompetenzen und Nutzungshäufigkeit steigern sowie langanhaltende Erinnerungen schaffen.

For my grandfather

Table of Contents

Abstract	I
Zusammenfassung	II
Table of Contents	V
List of Tables	VII
List of Figures.....	VIII
List of Acronyms	XI
1. Introduction	1
1.1. Motivation	1
1.2. Problem statement and research questions	2
1.3. Methodology and technologies.....	3
1.4. Relevance to science and society.....	4
1.5. Structure.....	4
2. Background	5
2.1. Storytelling.....	5
2.1.1. Pictorial maps.....	6
2.1.2. Multimedia cartography	7
2.1.3. Narrative cartography	8
2.1.4. Story maps.....	10
2.2. Real-time 3D rendering.....	13
2.2.1. Virtual globes	15
2.2.2. Topographic 3D maps	15
2.2.3. Thematic 3D maps.....	17
2.2.4. Animated 3D maps.....	18
2.2.5. Interactive 3D maps.....	19
2.3. Machine learning	22
2.3.1. Supervised learning	22
2.3.2. Artificial neural networks	24
3. Detection of Pictorial Map Objects with Convolutional Neural Networks.....	29
Abstract	30
3.1. Introduction	31
3.2. Related work.....	33
3.3. Experiments.....	34
3.3.1. Classification of maps vs. non-maps.....	34
3.3.2. Classification of pictorial maps vs. non-pictorial maps.....	39
3.3.3. Detection of sailing ships on maps.....	44
3.4. Discussion.....	49
3.5. Summary and future work.....	50
Appendix.....	52
References.....	53
4. Instance Segmentation, Body Part Parsing, and Pose Estimation of Human Figures in Pictorial Maps.....	59
Abstract	60
4.1. Introduction	61
4.2. Data	62
4.3. Methods.....	66
4.4. Results	69

4.5.	Discussion.....	73
4.6.	Conclusion and future work.....	75
	Appendix.....	76
	References.....	80
5.	Inferring Implicit 3D Representations from Human Figures on Pictorial Maps....	83
	Abstract	84
5.1.	Introduction	85
5.2.	Related work.....	86
5.3.	Data	87
5.4.	Methods.....	89
5.4.1.	3D pose estimation	89
5.4.2.	3D body part inference.....	90
5.4.3.	UV coordinates prediction.....	92
5.4.4.	Texture inpainting and enhancement	94
5.4.5.	Real-time rendering	96
5.5.	Use case	97
5.6.	Discussion.....	99
5.7.	Summary and future work.....	101
	Appendix.....	103
	References.....	105
6.	Conclusion	109
7.	Outlook.....	113
7.1.	Technology.....	113
7.2.	Concepts.....	114
7.3.	Usability.....	115
	Acknowledgements	117
	Curriculum vitae	119
	References.....	121

List of Tables

Table 2.1: Differences between texts and maps.....	5
Table 3.1: Correct classifications of maps and non-maps for the examined CNNs and image input options	37
Table 3.2: Correct classifications of pictorial maps and non-pictorial maps for the examined CNNs and image input options.....	42
Table 3.3: Average COCO metrics of the best Faster R-CNN models of three runs for different scales	47
Table 3.4: Average COCO metrics of the best RetinaNet models of three runs for different scales	47
Table 3.5: Average COCO metrics of the best Faster R-CNN models for small objects of three runs configuration and different scales	47
Table 3.6: Average COCO metrics of the best RetinaNet models for small objects of three runs configuration and different scales	47
Table 3.7: Digital libraries from which historic maps with sailing ships were retrieved for training Faster R-CNN and RetinaNet.....	52
Table 4.1: Frequency of occurrence of body part configurations in our test dataset, which consists of 387 pictorial figures.....	65
Table 4.2: Averaged COCO metrics of retrained Mask R-CNN models for different datasets and scales.....	69
Table 4.3: Averaged COCO metrics for body parts and pose keypoints for different datasets and architectures.....	71
Table 4.4: Possibilities to consider and decisions made for our CNN versions on parsing body parts and estimating poses.....	74
Table 5.1: Average root mean squared errors and percentages of correct keypoints of pose points for estimating depth coordinates of human poses using their and our data as well as different projections and number of pose points.....	90
Table 5.2: Average root mean squared errors and intersections over unions on our validation data for inferring 3D SDFs of hands from 2D masks	91
Table 5.3: Average root mean squared errors and intersections over unions on our validation data for inferring 3D SDFs of different body parts from 2D masks using DISN two-stream with pose points.....	91
Table 5.4: Mean absolute errors and root mean squared errors for predicting UV coordinates of pictorial human figures from different input data	93
Table 5.5: Application of storytelling concepts to our sketched story map	98
Table 5.6: Sources of the 3D story map with pictorial figures.....	104
Table 6.1: Correspondence of research questions and research articles.....	110

List of Figures

Figure 2.1: Different pictorial objects and textual passages are depicted on 'The story map of Spain'	6
Figure 2.2: Various narratives are communicated implicitly by the map 'Nova Totius Terrarum Orbis Tabula'	8
Figure 2.3: The map 'Are there Tsunamis in Switzerland?' is an example of intrinsic storytelling in the Atlas of Switzerland.....	11
Figure 2.4: Graphics rendering pipeline	13
Figure 2.5: Ray tracing.....	14
Figure 2.6: Geographic-grid tiling structure of a virtual globe	15
Figure 2.7: Simple topographic 3D map of a city.....	16
Figure 2.8: Different visualisation techniques for 3D objects and 3D charts on thematic maps.....	17
Figure 2.9: Nested hemispheres representing river runoffs in different seasons and time periods.....	18
Figure 2.10: Temporal animation of 3D objects on the terrain.....	19
Figure 2.11: Non-temporal animation of one of the stacked cuboids representing glacier volumes in different years	19
Figure 2.12: Common high-level interactions in 3D atlases and virtual globes.....	20
Figure 2.13: Interactive local terrain deformation by control point placement	20
Figure 2.14: Exemplary arrangements of variables/states and observations for different machine learning models represented by directed and undirected graphs.....	23
Figure 2.15: An exemplary Multi-Layer Perceptron consisting of three input variables, two hidden layers having four neurons each, and two output values.....	24
Figure 2.16: Scheme of a simple Convolutional Neural Network.....	25
Figure 2.17: Segmented building footprints by a CNN.....	27
Figure 3.1: ROC curves and auc scores for the tested CNNs and image evaluation options to classify maps and non-maps	38
Figure 3.2: The three most frequently misclassified maps by both CNN models for resized and average over grid image evaluation options.....	39
Figure 3.3: The three most frequently misclassified non-maps by both CNN models for resized and average over grid image evaluation options.....	39
Figure 3.4: ROC curves and auc scores for the tested CNNs and image evaluation options to classify pictorial maps and non-pictorial maps	42
Figure 3.5: Selection of three frequently misclassified pictorial maps by both CNN models for resized and average over grid image evaluation options	43
Figure 3.6: Selection of three frequently misclassified non-pictorial maps by both CNN models for resized and average over grid image evaluation options	43
Figure 3.7: Ground truth and detected bounding boxes with the best trained Faster R-CNN model for large, freestanding ships	48
Figure 3.8: Ground truth and detected bounding boxes with the best trained RetinaNet model for occluded and small ships.....	48
Figure 4.1: Exemplary pictorial map with human figures on the sides	62
Figure 4.2: Manually annotated body parts and skeletons of pictorial figures in the web application Supervisely on a touristic map	63
Figure 4.3: Real and synthetic entities randomly scaled and placed on a map	64

Figure 4.4: A real person, its body part mask and skeleton as well as a real object from the PASCAL-Part dataset; a synthetic person, its body part mask and skeleton from our SVG figure generator as well as an icon object from Iconfinder.....	65
Figure 4.5: Our custom CNN architectures to parse body parts and estimate poses simultaneously	68
Figure 4.6: One-hot encoded masks for body parts and keypoints of a human figure from the test dataset.....	68
Figure 4.7: Comparison of Mask R-CNN results for few, several, and many human figures on maps, between the original COCO model with an AP of 15.57% and the best retrained model with an AP of 19.38%.....	70
Figure 4.8: Selection of success cases, moderate failure cases, and severe failure cases for simultaneous body parsing and pose estimation for the best retrained model on our Simple UNet.....	72
Figure 4.9: Selection of pictorial maps from Pinterest.....	76
Figure 4.10: Selection of human figures extracted from pictorial maps from Pinterest.....	76
Figure 4.11: Selection of real persons extracted from photos from the PASCAL-part dataset	77
Figure 4.12: Selection of real objects extracted from photos from the PASCAL-part dataset	77
Figure 4.13: Selection of synthetically generated human figures by our custom web application with the MPII Human Pose dataset	78
Figure 4.14: Selection of icon objects from Iconfinder.....	78
Figure 4.15: Selection of maps without any human figures from Pinterest	79
Figure 4.16: Selection of maps without any human figures, enriched with real or synthetic entities	79
Figure 5.1: Our data processing workflow	88
Figure 5.2: Estimated 3D poses from 2D poses of pictorial figures from our test data after training the network of Martinez et al. with our data.....	90
Figure 5.3: SDF around a hand viewed from the top, back and left, and in oblique perspective.....	91
Figure 5.4: Inferred body part SDFs from masks by DISN one-stream with concatenated pose points.....	92
Figure 5.5: Network architecture for predicting UV coordinates from a depth map	93
Figure 5.6: Predicted UV coordinates of pictorial human figures from a depth image and body part masks by our fully convolutional network	93
Figure 5.7: Generated textures of two pictorial figures from a given image as well as source and target UV maps by the inpainting network	94
Figure 5.8: UV coordinates in the inpainted texture and in the colour wheel at the position of the left eye of a pictorial figure	94
Figure 5.9: Network architecture for enhancing textures of inpainted heads.....	95
Figure 5.10: Denoised and resynthesised textures of inpainted heads from different views.....	95
Figure 5.11: Remeshed pictorial 3D figure placed on the original map in Blender.....	96
Figure 5.12: Remeshed pictorial 3D figure in the virtual globe CesiumJS.....	96
Figure 5.13: Sketched story map for children about Charles Darwin in Patagonia.....	98
Figure 5.14: Failure cases of inferred body parts.....	100
Figure 5.15: All pictorial human figures on maps from our test dataset and our inferred 3D models from different views	102
Figure 6.1: Workflow pursued in the research articles	111

Figure 6.2: An inferred human figure rendered via sphere tracing in a virtual globe. 112
Figure 7.1: Cartographic storytelling as a multisensory experience including different types of verbal and non-verbal communication..... 114
Figure 7.2: Comparison between intrinsic and extrinsic cartographic storytelling 115

List of Acronyms

ANN	Artificial Neural Network
API	Application Programming Interface
AP	Average Precision
AR	Augmented Reality
AUC	Area under the ROC Curve
BC	Before Christ
BN	Bayesian Network
CAD	Computer-Aided Design
CNN	Convolutional Neural Network
COCO	Common Objects in Context
COVID-19	Coronavirus Disease 2019
CPU	Central Processing Unit
CRF	Conditional Random Field
CZML	Cesium Markup Language
DT	Decision Tree
ESRI	Environmental Systems Research Institute
FCN	Fully Convolutional Network
FIFA	Fédération Internationale de Football Association
FPS	Frames per Second
GAN	Generative Adversarial Network
GIS	Geographic Information System
GML	Geography Markup Language
GNN	Graph Neural Network
GPS	Global Positioning System
GPU	Graphics Processing Unit
HMM	Hidden Markov Model
HTML	Hypertext Markup Language
ICA	International Cartographic Association
ISO	International Organisation for Standardisation
IoU	Intersection over Union
JPEG	Joint Photographic Experts Group
JSON	JavaScript Object Notation
KML	Keyhole Markup Language
k-NN	k-Nearest Neighbours
LOD	Level of Detail
MAE	Mean Absolute Error
MLP	Multi-Layer Perceptron
NB	Naïve Bayes
ND	N-dimensional ($N \in 1, 2, 2.5, 3, 4, 5$)
OBJ	File Format by Wavefront Technologies
OGC	Open Geospatial Consortium
PASCAL	Pattern Analysis, Statistical Modelling and Computational Learning
PCK	Percentage of Correct Keypoints
R-CNN	Region-based Convolutional Neural Network
RF	Random Forest
RGB	Red Green Blue

RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristics
SDF	Signed Distance Field/Function
SMPL-X	Skinned Multi-Person Linear Model (incl. expressions)
SVG	Scalable Vector Graphics
SVM	Support Vector Machine
TIN	Triangulated Irregular Network
UI	User Interface
US	United States (of America)
VR	Virtual Reality
VRML	Virtual Reality Modelling Language
X3D	Extensible 3D
XML	Extensible Markup Language

Note: Acronyms of artificial neural network architectures are not listed.

1. Introduction

1.1. Motivation

Storytelling is a popular concept applied in various domains including cartography. In recent years, so-called 'story maps', which are maps enriched with multimedia elements like texts, photos, or videos, have gained popularity (Cartwright & Field, 2015). Study projects have shown that story maps are well-suited for educational purposes, for instance, to teach spatiotemporal phenomena (Marta & Osso, 2015). Generally, it is aimed to make maps easier accessible and comprehensible for a broader audience through storytelling (ESRI, 2012). Inspired by historic pictorial maps, where sea monsters, ships, or other figures give an impression of the real or imagined world (Child, 1956), this dissertation will focus on transforming illustrative objects into storytellers in digital maps. These visualisations may gain the empathy of the map reader because they are entertaining, and in turn, engaging (Borkin et al., 2013). Next to conveying personal stories, background information, or peculiarities of a thematic map, also interactive map functions or quizzes may be offered while complementing the figurative objects with speech bubbles or other overlays. When being animated additionally, pictorial objects may guide the map reader to interesting places and special events on the map, or serve as avatars for other users.

Stories with illustrative objects can be told on 2D and 3D maps. Although producing 3D maps is technically more demanding, it enables the development of cartographic applications using virtual globes (Sieber et al., 2016), virtual reality (Lütjens et al., 2019), or augmented reality (Schnürer et al., 2020). While objects in current 3D maps are mainly tessellated and processed within the rasterisation pipeline, this dissertation will examine in particular ray tracing algorithms for real-time rendering. Being favoured by recent advancements in performing parallel operations on graphics processing units (GPUs), ray tracing can simulate complex effects in the scene via sending rays starting from the camera or light sources. Ray tracing also facilitates the rendering of 3D objects as implicit surfaces (Hart, 1996), which are described by mathematical functions in Euclidean space satisfying the equation $f(x, y, z) = 0$. Implicit surfaces support exact representations of geometric primitives (e.g. spheres), constructive solid geometry operations (e.g. difference), and deformations (e.g. morphing). For instance, a penguin (Abgottspon, 2011), a rabbit (Tomczak, 2012) and a dinosaur (Quilez, 2015) were modelled as implicit surfaces and rendered in real-time, which is a prerequisite for further animations and interactions with these objects.

As the manual construction of pictorial 3D objects for maps is cumbersome, this dissertation will investigate machine learning methods to automate this task as well as the analysis of decorative objects on existing maps. Machine learning, also known as artificial intelligence, exists for several decades (Rosenblatt, 1958) to solve mainly optimisation problems by approximating functions. Artificial neural networks (ANNs) are one of the computational models of machine learning, which mimic neuronal connections and activations in the human brain (Dayhoff, 1990, as cited in Merwin et al., 2009). With the help of ANNs, for example, values can be interpolated for areas, where source and target zones are different (Merwin et al., 2009). In previous years, transferring operations from central processing units (CPUs) to GPUs reduced largely training times

of ANNs, such as convolutional neural networks (CNNs) (Scherer et al., 2010). Further adaptations of the network architectures led to significant improvements in accuracy, for instance for labelling images (Krizhevsky et al., 2012), recognizing objects (Girshick, 2015), or generating 3D models (Saito et al., 2019). During the time of writing this dissertation, these new techniques in machine learning, which are referred to as deep learning due to the large network sizes, have been also applied to solve cartographic problems.

1.2. Problem statement and research questions

Maps are increasingly available in digital form and published via the internet. An automated categorisation of the map content would be beneficial for indexing maps in search engines or archiving them in libraries to not rely only on enclosed captions, which may be erroneous or missing. Similar tasks have been tackled in other domains than cartography, for example, remote sensing (Zhu et al., 2017), where deep learning methods are used with promising accuracy rates. In this dissertation, it is aimed to identify maps, in particular illustrative maps, in image collections by ANNs. A precondition before training the networks will be to clarify semantical issues by formulating map definitions.

RQ1a: How accurately can pictorial maps be distinguished from other maps and images using artificial neural networks?

Many digital maps are scans of printed maps or created in raster graphics editors. These maps are often annotated with metadata (e.g. title, author, creation date), but information about their content is largely missing due to the effort in listing all map elements. Automating this task would help to offer additional filter options for the advanced search of digital map libraries or social media websites. Since bounding boxes (Girshick, 2015) and silhouettes (He et al., 2017) of various real-world objects could be detected in photos by CNNs, it is examined in this dissertation how to transfer and adapt these networks to identify illustrative objects in maps. Additionally, it is intended to segment parts and locate key points of objects for skeletal animation on the original map or on other maps.

RQ1b: How accurately can pictorial objects including parts and key points be detected on maps using artificial neural networks?

In this dissertation, it is planned to automatically create similarly looking 3D models based on the given 2D templates. This task is also known as single-view 3D reconstruction (Saito et al., 2019) and has been facilitated by ANNs. Since multiple versions are possible due to occlusions, only the plausibility of poses and shapes as well as the visual quality of textures of the resulting figures can be assessed. The constructed models can be inserted into 3D maps suited for children, tourists, museum visitors, or atlas users. Surprisingly, illustrative objects are sparsely used in 3D maps for this target audience. One reason for the absence might be explained by the technical challenges,

such as handling massive datasets (Cozzi & Ring, 2011), which keep map editors from spending thoughts on illustrative elements and their interaction with the map reader.

RQ2: How effectively can the detected pictorial objects be converted into 3D models, suited for skeletal animation, using artificial neural networks?

3D models usually consist of vertices, edges, and faces, having colours or textures assigned, and are rendered in real-time with the graphics rendering pipeline. However, 3D representations by implicit surfaces, which can be rendered with ray tracing algorithms, are rather unexplored, especially in cartography (Schnürer et al., 2017). Implicit surfaces are based on mathematical functions, which first need to be solved to determine the location of the objects (Fryazinov & Pasko, 2008; Singh & Narayanan, 2010). Visual noise may appear as the steps to trace object surfaces have to stop at some time if the algorithm is to be executed in real-time. In this dissertation, the performance of one of the ray tracing algorithms will be assessed to render animatable 3D objects.

RQ3: How efficiently can the derived pictorial 3D models be rendered by ray tracing implicit surfaces while enabling interactive cartographic storytelling?

1.3. Methodology and technologies

An in-depth literature survey is conducted to gain insights into storytelling, 3D real-time rendering, and machine learning. Starting from an initial set of research articles, related articles are identified in Google Scholar¹ and Scopus² by a forward and backward reference search. Additionally, a custom search engine based on ElasticSearch³ is implemented to find literature specifically in cartographic journals. It is to be noted that the boundary between cartography and GIS/geovisualisation is quite vague: Cartography tends towards presentation and orientation, whereas GIS/geovisualisation rather targets analysis and exploration.

For supervised learning with ANNs, it is essential to gather large amounts of training data. Unlike general computer vision, where real-world photos have been annotated in large-scale datasets, datasets involving maps to solve cartographic problems have been non-existing yet. Therefore, data is crawled from web collections, synthetically generated, or adapted from real-world datasets. The data is fed into ANNs specialised in the following tasks: classification, object detection, instance segmentation, semantic segmentation, key point detection, single-view 3D reconstruction, and texture inpainting. ANNs based on existing architectures are trained via transfer learning, when pre-trained models on real-world images are available, whereas others are trained from scratch. Existing ANN architectures are customised or own architectures are developed by assembling common building blocks. The results are evaluated quantitatively using

¹ <https://scholar.google.com/>

² <https://www.scopus.com/>

³ <https://www.elastic.co/elasticsearch/>

existing metrics and qualitatively by assessing success and failure cases. ANNs are implemented in Python by mainly the TensorFlow⁴ framework.

The generated 3D models are represented implicitly by signed distance functions (SDFs), which indicate whether a point lies inside, on, or outside a surface. Intersections of rays with objects can then be determined by sphere tracing (Hart, 1996). Textures are mapped to the surfaces using UV coordinates. The ray tracing algorithm is implemented with Numba⁵ in Python.

1.4. Relevance to science and society

Machine learning, especially deep learning in computer vision, has been one of the most rapidly evolving research fields in the previous years. This dissertation supports these advances by applying network architectures to essential cartographic tasks, such as map identification, object detection, or 3D construction. These tasks are hard to automate by conventional algorithms and time-consuming when being performed manually. Machine learning systems, however, establish models by optimisation. To train and validate the models, datasets are created in the scope of this dissertation, which may be reused by other scientists. As theoretical foundations, the definition of a map is revisited and the concept of the cartoverse (= the cartographic metaverse) is introduced.

Although pictorial objects are mainly focused in the given work, experiments and findings may be transferrable to other map elements (e.g. buildings). So far, mainly static pictorial objects have been depicted on maps, but the addition of animated interactive objects is rather unexplored. Especially in 3D maps, these supplemental objects may unfold their potential by enlivening landscape and urban scenes. Heretofore research in 3D cartography has been specialised in visualisation techniques for different map topics (e.g. Schnürer et al., 2015), though means of presentation and user experience were mostly neglected. Therefore, this dissertation proposes to endow objects with storytelling capabilities to make the content of maps more graspable, coherent and distinguishable for the audience, particularly in educational settings.

1.5. Structure

This introductory *Chapter 1* has stated the motivation, research gaps, methodology, and relevance of this dissertation. *Chapter 2* provides an overview of core definitions and methodologies for storytelling, 3D real-time rendering, and machine learning. Applications and relations of these three topics to cartography are outlined additionally. *Chapters 3 to 5* comprise the three research articles for this cumulative thesis. The first two articles are about detecting pictorial objects in historic and contemporary maps. In the third article, animatable 3D models are inferred from the extracted pictorial figures. *Chapter 6* summarises and appraises the results of these articles. *Chapter 7* anticipates and discusses future work.

⁴ <https://www.tensorflow.org/>

⁵ <https://numba.pydata.org/>

2. Background

2.1. Storytelling

Storytelling is “[t]he action or activity of telling stories” (OED Online, 2022b). The term ‘story’ has different meanings (OED Online, 2022b): On the one hand, stories can be based on real events or incidents in the past or present. On the other hand, stories can be fictitious (e.g. myths, legends) and sometimes negatively connoted (e.g. assertions, lies). Stories usually have a subject (e.g. persons, countries, institutions) and can be conveyed by objects (e.g. images). Stories can be told personally, and are ideally interesting or entertaining for the receiver. Most stories are rather short but can be the foundation for larger works (e.g. novel, play, film) or other entities (e.g. newspapers, businesses).

The plan or scheme of stories is called the plot or storyline, which is an ordered sequence of main events or principal stages (OED Online, 2022b). According to Freytag’s model of dramatic action, a plot typically consists of five parts: exposition, build-up, climax, peripeteia, and catastrophe (Balme, 2005). Narratives unfold in one or more parts, and connect the events (OED Online, 2022b). An orientation aid to construct scenes and narratives for the plot is to answer the 5 W’s questions: Who? (e.g. protagonists, supportive characters), What? (e.g. dialogues, actions), When? (i.e. time spans), Where? (e.g. places, routes), Why? (e.g. motivation of people, natural laws).

	Text	Map
Dimensionality	1D (= linear structure)	2D, 3D
Animability	only static	possibly dynamic
Thematic aspects	in local passages	in layers
Spatial relations	need to be explicitly mentioned	implicitly contained
Spatial context	can be contradictory or missing	given by topology
Scale	flexible	fixed
Completeness	open-world assumption	closed-world assumption
Representations	words	graphic symbols
Perception	usually linearly along the reading direction	recognised at a glance and in arbitrary order
Spatial search	tediously word by word or by a register	easily by a spatial index or by a grid-based gazetteer
Appraisal	rather subjective and emotional	rather objective and factual
Atmosphere	conveyed by descriptions	conveyed by design
Generalisation	individual summaries or automatable by natural language processing	custom aggregations/simplifications or automatable by algorithms following standardised rules

Table 2.1: Differences between texts and maps (adapted from Mocnik & Fairbairn, 2018)

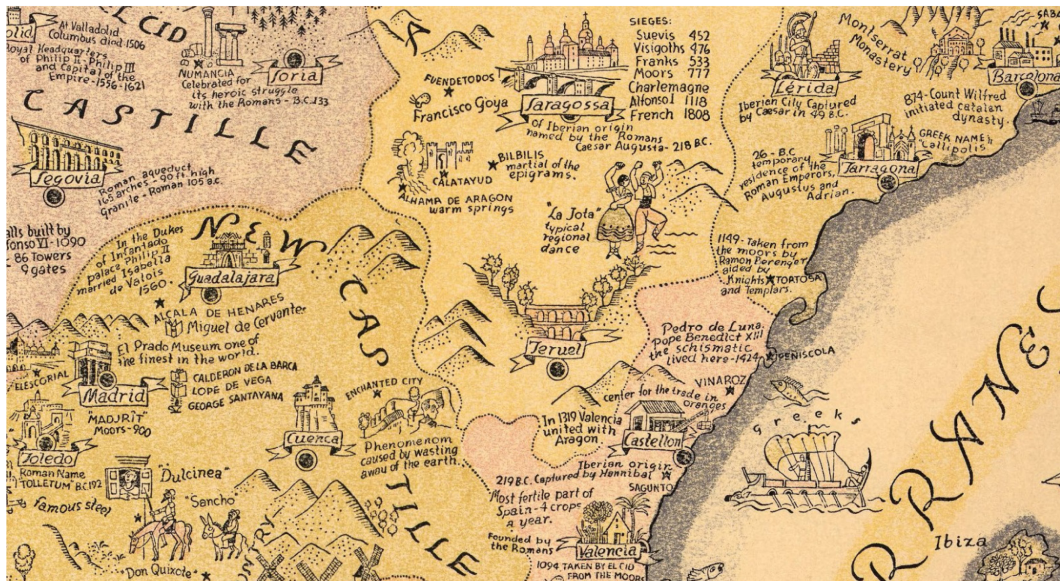


Figure 2.1: Different pictorial objects and textual passages are depicted on 'The story map of Spain' (Diego, 1935).

Traditional stories are transmitted in oral, written, and audio-visual forms (Caquard & Cartwright, 2014). In recent years, storytelling concepts and techniques have been also adapted in data visualisation disciplines such as cartography. This is not a straightforward endeavour because stories, transcribed as text, and maps have different properties (Table 2.1). As a visual coalescence, for instance, shorter textual passages can be placed on a map (Figure 2.1). Besides text, also illustrations can be added to maps to convey certain narratives. These kinds of maps are known as pictorial maps. Stories then unfold within the map (i.e. intrinsically) by included map objects, in contrast to stories emerging from outside the map (i.e. extrinsically) by enclosed texts and visual elements (Bonassi & Sieber, 2017).

2.1.1. Pictorial maps

The origins of pictorial maps lie in European maps from the Middle Ages, the Age of Exploration, and the Age of Enlightenment, where pictorial objects, such as ships and imaginary monsters, are embedded into the map. In other cases, large pictorial figures are merged with topographic elements; for instance, a lion frames Belgium and the Netherlands in the map 'Leo Belgicus' produced by Aitzinger in 1583. Pictorial maps are often surrounded by decorative borders, for example, different coats of arms and cities enclose a later version, ca. 1611, of the map 'Leo Belgicus' by Visscher. Next to terrestrial, also celestial pictorial maps were produced with heavenly figures and symbolic features. (Kanas, 2019)

Another peak of pictorial maps was in the middle of the 20th century, when maps were included in newspapers and magazines, and when popular culture was closely related to technologies such as cars and aeroplanes (Cosgrove, 2005). For instance, a scenic route is illustrated by photos, sketches and texts in Owens's 1929 map 'Have you seen the Pinnacles National Monument?'. Another example of this period is an atlas map, created by White in 1935, which shows cartoonish humans, such as the Pilgrim Fathers

or cowboys, in the US (Griffin, 2017). Pictorial and cartoon maps are especially appealing to students of elementary schools as these maps simultaneously amuse and teach (Price, 1937). A literary genre, which gained popularity among teenagers and adults in the 20th century, concerns fantasy novels. Some of these imaginary settings, such as Howard's 'Hyborian Age', are illustrated with maps including pictorial embellishments (Ekman, 2013). Pictorial objects are not only drawn in two dimensions but also shown perspectively, such as the axonometric buildings in Bollmann's 'New York City Picture Map' (Cosgrove, 2005).

Tourists and children are also the main user groups of modern pictorial maps. For example, the 2014 map 'Beer City Ale Trail' by the Grand Rapids Convention and Visitor Bureau represents numerous breweries by pictorial beer bottles and glasses to create "an inviting, warm, [...] slightly juvenile, cutesy feel" (Feeney, 2017, p. 19). Interactions and animations can be introduced into digital pictorial maps, such as those published by InsideAsia Tours (K. Clarke, 2016). Scenes and objects can be depicted also three-dimensionally, for instance by applying a toon shader to simplified sights and mountains (Naz, 2005). Oblique views of pictorial maps are likewise suited to familiarise children at early ages with wayfinding (Sigurjónsson et al., 2020).

Pictorial maps may hold certain narratives as exemplified in section 2.1.3. Further information about the history of pictorial maps is given in section 3.1.

2.1.2. Multimedia cartography

"Multimedia uses different media to convey information as text, audio, graphics, animation, and video, all done interactively" (p. 1). In contrast to printed cartographic products, multimedia cartography enables interactions such as changing the scale, querying the underlying data, and following links to external media (Cartwright & Peterson, 2007). Next to desktop computers, the term 'multimedia cartography' is nowadays also associated with maps on mobile devices for augmented and virtual reality (Medyńska-Gulij et al., 2021).

Media are called hypermedia when being connected via the World Wide Web (Cartwright & Peterson, 2007). An advantage of linking media in cartography is that information in digital atlases, for example, can be kept up-to-date and external services can be provided (Richard, 2000). In addition to the editorial staff, who provides content to websites or web applications, also users may contribute media via the Web, which is also referred to as 'Web 2.0'. Participatory processes in urban and regional planning (e.g. land management), where users can share their ideas, opinions, and emotions (Rocca, 2013), are a cartographic example of the Web 2.0 phenomenon.

Multimedia cartography can thus be seen as a predecessor of story maps lacking narrative aspects.

2.1.3. Narrative cartography

On the one hand, narratives and maps are interconnected because maps are means of communication. Narratives can be regarded as an additional dynamic layer on top of the static topographic layer, whereas the main narrative can be split into a series of smaller narratives (Vannieuwenhuyze, 2020). For example, the narratives conveyed by different elements in a world map by Blaeu in 1648 (Figure 2.2) can be interpreted as “pro-Dutch [...], Euro-superior, male-dominated, in favour of Copernicanism, and optimistic about the progress of knowledge since classical times” (Netten, 2020, p. 1). Next to being provided with contextual information, the interpretation of maps and included narratives depends on various factors like symbol literacy, domain knowledge, or the cultural background of the reader, which may result in different conclusions.

On the other hand, narratives and maps are interlinked because stories are set in certain places (Caquard, 2011). In that sense, narrative cartography depicts “spatiotemporal structures of stories and their relationships to places” (Caquard & Cartwright, 2014, p. 101) on maps. Stories may originate from fiction, such as literature and films, or reality, possibly perceived from a personal perspective. Examples of mapping fiction are given by tracing and visualizing implicit and explicit connections of places in Storm’s novella ‘Der Schimmelreiter’ (Reuschel & Hurni, 2011) or in the movie ‘Ararat’ (Caquard & Fiset, 2014). An example of mapping a subjective reality is given by showing the travel route of a soldier during the First World War, which is illustrated by photos, videos, diary entries, and military forms (Cartwright & Field, 2015).



Figure 2.2: Various narratives are communicated implicitly by the map ‘Nova Totius Terrarum Orbis Tabula’ (Blaeu, 1648).

In the following, conceptual aspects of narrative cartography, which is also known as 'cartographic storytelling' or 'geographic storytelling', are presented. Design aspects and technical details of the resulting maps, so-called 'story maps', are described in the following section.

Concepts of narrative cartography may be transferred from theatre. The extent of the landscape can be seen as the stage and elements on the landscape as actors, while the script prescribes the processes of elements within the landscape. In a broader sense, the 4D geographical space can represent the stage, map editors can be directors, and users can act as players, who either follow the script of the directors (e.g. playing animations) or improvise (e.g. explore the application interactively). (Cartwright, 2009)

Another inspiration to narrative cartography is cinema - not only since maps in films are often technically highly advanced. A theoretic framework to be adapted in cinema and cartography is the model of eloquence, which defines certain form factors ranging on a continuum between simplicity and complexity. For example, a base map with a minimal number of symbolised elements may be recognised as simple, while a base map with a shaded relief may be perceived as complex. Overall, narratives act as a glue between descriptive and persuasive features in films and maps. (Muehlenhaus, 2014)

A concept of narrative cartography originating from literature is 'story focus', where "[e]verything that is irrelevant [...] remains unrepresented" (p. 50). The relevance of features and events may change by following the storyline, for instance, when characters move in space and time. Cartographic means to vary the focus are "layers of content, levels of detail, scale, precision and uncertainty, emotion and mood" (p. 51). For example, only certain map parts may be revealed, some parts may be enlarged, or the struggles of protagonists may be represented (Mocnik & Fairbairn, 2018). When centring the story and emotions, even non-Euclidean maps may be created (Olmedo & Caquard, 2022).

Branching stories is a technique from computer games (e.g. role-playing games). Compared to linear stories, which are presented in sequential order (e.g. by time), non-linear stories depend on the decisions of the user, for example, multiple-choice questions or the order of navigating to certain places on the map. Non-linear structures may lead to different endings of a story and users are tempted to explore alternative options. To keep an overview, the flow of storylines can be sketched on a storyboard and divided into single scenes, which inform about the characters, theme, setting (i.e. time and place), and point of view (i.e. of the narrator). (Thöny et al., 2018)

Other inspirations from computer games for cartographic storytelling are interactive simulations to foster strategic thinking, puzzles to challenge the user, or experience systems to provide incentives to return (Cartwright, 2004). Since many computer games are rendered in 3D, visualisation techniques (e.g. meteorological effects, streamlines) and graphic techniques (e.g. reflection, glow) can create certain atmospheres and effects, especially for stories depicted on 3D maps. Interactive navigation and depth perception may lead to an immersive experience for the user in 3D story maps, being even more reinforced in virtual reality (Thöny et al., 2018).

2.1.4. Story maps

In the 1930s, a series of pictorial maps entitled '(The) story map of ...' (Figure 2.1) was created by several authors (Chase et al., 1935). The maps were published by the company 'Colortext' (Clinton, 2022) and mostly featured a country in Europe, North America or Central America. Local customs, food and drinks, animals, and famous personalities were portrayed next to topographic features such as cities, which were represented by iconic buildings. Most of the pictorial objects were labelled, and also historic events and episodes were mentioned on some of these story maps. Similar maps have been produced in the follow-up years and also nowadays pictorial maps with textual descriptions can be found. The map series 'The Appendix Guide to ...' (Cannon, 2013), for instance, illustrates travel routes of pioneering expeditions. Other maps with textual passages originate from literary works (Lewis-Jones, 2018), such as Melville's 'Moby-Dick', and from art installations, for example, Picton's 'London 1940 panels' (Streifeneder & Piatti, 2021).

With the rise of multimedia cartography, interactive map applications have been created. An early example is the Atlas of Indigenous Perspectives (Caquard et al., 2009), where stories have been introduced in the form of treaties, travel reports, spiritual theories, or interviews. In the 2010s, story maps have been adapted and coined by the geospatial software company 'ESRI'. The company provides several user interface (UI) templates (e.g. photo gallery, journal, cascade) with different layouts (e.g. tabs, accordion) and tools (e.g. swipe, spyglass). ESRI (2012) also formulates principles on how to create effective story maps, such as having a defined target audience, an ice-breaking beginning or title, and a well-balanced map with a simple story. Beyond, the company offers an easy-to-use authoring tool and grants hosting space so that a multitude of story maps have been created. A series of ESRI story maps have been published by Varvara Antoniou, for instance about the Greek peninsula Methana (V. Antoniou et al., 2018). Other tools to create interactive story maps are Google Tour Builder, Tripline, Mapstory, Atlascine, or Neatline (Caquard & Dimitrovass, 2017).

Cartographic stories can be told with 2D and 3D maps (e.g. V. Antoniou et al., 2018). A special 3D map is the space-time cube, where the time axis is orthogonal to a 2D map or the surface of a 3D map. The space-time cube can be annotated with images (Eccles et al., 2008; Kraak & Kveladze, 2017) or comic-like scenes (A. B. Moore et al., 2018) for storytelling. 3D maps are suited for virtual reality applications, possibly in combination with gamification techniques, for example, to present a former industrial site (Edler et al., 2019) or a historic castle (Matthys et al., 2021). Virtual 3D objects, which are relevant to the story, can enrich the real world in augmented reality applications, for instance for cultural heritage (Koutsabasis et al., 2022). On mobile devices, other visual information (Lu & Arikawa, 2013) or sounds (Indans et al., 2019) related to the story can be revealed once a user reaches a certain location in the real world.

As a general classification for story maps, it can be distinguished between extrinsic and intrinsic storytelling (Bonassi & Sieber, 2017; Denil, 2017). In extrinsic storytelling, maps illustrate and support a story, thus maps are one of many multimedia elements. To advance the story, actions are triggered from outside the map, for example, the page is scrolled, a photo next to the map is clicked, or the time slider is moved. This is the case for many of the ESRI story maps. In intrinsic storytelling, the story is transported via the map itself, while other multimedia elements support the map (Figure 2.3). The reader

constructs a story, which not necessarily matches with the editor's intentions, from the proposed facts and relationships of the map. Examples of intrinsic storytelling are static maps (e.g. containing symbols and labels), animated maps (e.g. following a route), and interactive maps (e.g. clicking on elements inside the map).

An analysis of the content of digital story maps revealed several storytelling genres (Roth, 2021), which can be understood as templates having certain map or UI elements to advance the storyline. Examples are static visual stories (e.g. by numbering), multimedia visual experiences (e.g. hyperlinks), or compilations (e.g. of real-time events). Design aspects of story maps can be summarised in a couple of storytelling tropes (Roth, 2021), for instance, continuity (e.g. navigation), dosing (e.g. partitioning), or voice (e.g. typography). Surveys have been conducted on particular storytelling topics, for example, climate change or the COVID-19 virus. Authors reported that they highlighted key data and reduced embellishments to raise the reader's attention. Additionally, they employed metaphors by size comparisons and less abstract representations to reduce complexity (Fish, 2020). Authors of story maps followed mostly longform infographics with scrollbars and piqued curiosity by colour and novelty (Prestby, 2022).

Space and time are inherent properties of story maps. For a series of photos accompanying a map, walkthrough, panoramic views, and focus on single objects can be identified as spatial patterns, and cycles and retrospection as temporal patterns (Fujita & Arikawa, 2011). Besides slideshows and animations, multiple coordinated views (e.g. for data brushing), layer superimposition (e.g. by colour-coding), and layer juxtaposition (i.e. small multiples) are possible methods to depict spatiotemporal data and to explore story maps (Mayr & Windhager, 2018).



Figure 2.3: The map 'Are there Tsunamis in Switzerland?' is an example of intrinsic storytelling in the Atlas of Switzerland. Stories are triggered based on the camera at different zoom levels.

The underlying spatiotemporal data of story maps, such as place names in border regions (Mościcka & Kuźma, 2018) or movable cultural monuments (Mościcka & Zirowicz-Rutkowska, 2018), can be stored in relational or graph databases. The latter, namely network-based structures, which are also known as ontologies, are particularly suited to link related stories based on thematic connections (Zanda et al., 2019) as well as to acquire and integrate data from different sources (Mai et al., 2022). The stored data can be algorithmically evaluated to automatically generate certain parts of story maps, for instance, events can be analysed to visualise flows and to produce text passages (Tateosian et al., 2020).

Concluding, decisive factors of contemporary story maps are historic roots (e.g. pictorial objects) and technological advances (e.g. AR/VR, machine learning), links to narrative disciplines (e.g. literary/theatre science) and other sciences (i.e. topic-related), general storytelling concepts (e.g. extrinsic/intrinsic, gamification) and scene-specific methods (e.g. settings, narratives), data (e.g. correlations, causations) and anecdotes (e.g. emotions, fantasies) as well as cartographic visualisations (e.g. space-time cube, animations) and user interface features (e.g. photo galleries, parallax effect).

Note

This section mainly focused on transferring storytelling concepts from literature and related sciences to maps, but only marginally on mapping literary works, which has been the subject of another dissertation (Weber, 2014).

2.2. Real-time 3D rendering

Real-time 3D rendering enables smooth animations and allows users to navigate in the digital space and to interact with its content in a seamless way. Although basic interactivity starts at 6 frames per second (FPS), video games are targeted at producing scenes with smooth transitions between rendered images at 30 FPS or higher (Haines et al., 2018).

Traditionally, a sequence of operations, the so-called *graphics rendering pipeline*, is performed to render 3D scenes in real time. The graphics rendering pipeline consists of four main stages (Figure 2.4) and is often referred to simply as rasterisation. In the application stage, user inputs from different devices (e.g. keyboard, mouse) are handled, collisions between objects are detected, and culling algorithms (e.g. back-faces, occlusions) are applied, amongst others. In the geometry processing stage, vertex positions and vertex data (e.g. normal coordinates for lighting) are transformed from model to world space, and further to view space based on given camera parameters like projection (e.g. perspective, orthographic). Additionally, geometries are clipped at the window boundary, normalised and scaled to the screen size. In the rasterisation stage, all pixels within a geometric primitive, commonly a triangle, are detected. In the pixel processing stage, depth and colour values are interpolated and textures applied. (Haines et al., 2018)



Figure 2.4: Graphics rendering pipeline

The second main 3D rendering technique is *ray tracing*. Compared to rasterisation, ray tracing is more efficient for calculating reflections, refractions, and shadows. However, ray-traced lines may appear sharp and aliased (Slusallek et al., 2005). Ray tracing can be performed in real-time with nowadays graphic boards, although a denoising step may be necessary for post-processing. In classical ray tracing, imaginary rays are fired from the camera through a pixel grid into the scene (Figure 2.5). Each ray is tested for intersections with scene objects and if there are any, the nearest object is identified. From the hit location on this object, rays to the light sources are calculated and mixed with the material properties such as colour. When considering mirroring surfaces and translucent objects, secondary rays for reflections and refractions may be calculated recursively (Haines et al., 2018). There are different variations of ray tracing such as ray casting, which involves only primary rays, path tracing, which samples rays stochastically at hit locations, or sphere tracing, which is based on signed distance fields.

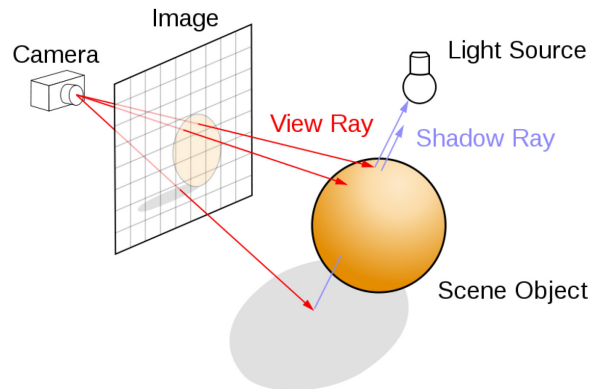


Figure 2.5: Ray tracing (Henrik, 2008)

One of the first 3D maps being rendered in real-time depicts terrain models and extruded polygons illuminated by an artificial light source (Moellering, 1980). A decade later, shaded terrain models and simple houses, forests, and roads including colours are presented on an interactive 3D map (Kraak, 1994). These early works make use of graphic and animation languages, such as IRIS GL, but it is often not clear at which speed the scenes were established and which algorithms were used. Lindstrom et al. (1996), for instance, report rendering a virtual flight over a wireframe of a multiresolution terrain surface at 20 FPS.

Around the millennium, programmable shaders for vertices and pixels were introduced for graphic boards, favouring the rasterisation approach (Góralski, 2009), whereas ray tracing is applied only in a few cases (see section 2.2.1). Low-level APIs such as OpenGL (Döllner & Kersting, 2000), DirectX (Lorenz & Döllner, 2008), or WebGL (Christen et al., 2012) offer full control of the capabilities of graphic boards to render performant 3D map scenes. With high-level APIs like Java3D (Hobona et al., 2006), OpenSceneGraph (Gaitán et al., 2006), or three.js (Limberger et al., 2017), it is easier to import different data formats, to handle events, or to create light sources, amongst others.

Game engines, for example, Unreal (Germanchis et al., 2004), CryEngine (Germanchis et al., 2007), or Unity (Laksono & Aditya, 2019), offer even more features than high-level APIs. Those products facilitate physics-based rendering, the creation of virtual characters, and the sharing of assets and plugins with others. Scenes and objects can be created in game engines via a user interface, however, georeferenced data is mostly not supported. Virtual globe engines provide fewer features than game engines but inherently support the tiling of terrain models and georeferenced data into multiple levels of detail. Available software development kits for virtual globes are osgEarth (Sieber et al., 2016), Cesium (Gede & Jeney, 2017), and the ArcGIS API for JavaScript (Stähli et al., 2018). More details about virtual globes are given in the next section.

Suitable formats for storing and styling 3D map data are VRML (K. Moore, 1999), KML (Sandvik, 2008), CityGML (Kolbe, 2009), X3D (von Reumont et al., 2013), CZML (Gede & Jeney, 2017), or 3D Tiles (B. Mao et al., 2020). The first four formats are XML documents, CZML is based on JSON, whereas 3D Tiles consist mainly of binary encoded files. All formats define (geographic) geometries and graphic properties, some support also attribute data and animations.

2.2.1. Virtual globes

The concept of a 'Digital Earth' has been shaped in a speech by a former US vice president (Gore, 1998). The term has been broadened to virtual globes to enclose also other planets than Earth and to rather refer to software-based representations (Harvey, 2009). A virtual globe is defined as "a scale-bound, structured model of a celestial body (respectively firmament) presented in its undistorted three-dimensional wholeness" (Riedl, 2000). Examples of virtual globes are Google Earth⁶ and ArcGIS Earth⁷, which offer a terrain model, a selection of base maps, additional 2D and 3D map content, and simple analytical functions.

The geometry of a virtual globe is a sphere in its simplest case, which may be already a sufficient model for digitised historic globes (Gede, 2015). In more advanced virtual globes, the geometry is an ellipsoid whose surface is hierarchically subdivided to display the terrain, base imagery, and additional data at multiple resolutions depending on the current view. Possible tiling structures (Figure 2.6) are tetrahedrons, triangles, cubes projected to ellipsoids, or tessellated geographic grids (Cozzi & Ring, 2011). Alternative structures to reduce shape distortions at the poles are based on hexagons (Sherlock et al., 2021; M. Zhou et al., 2013).

Attempts have been made to render virtual globes via ray tracing in real time. The potential of ray tracing lies in effectively and efficiently rendering huge datasets, including environmental features like clouds, water, or vegetation. As preconditions, a high-speed data transfer from server to client as well as high-end graphic equipment are required (Christen, 2008). To increase the frame rate, ray misses can be reduced by replacing the bounding box of the ellipsoid with a viewport-aligned polygon (Cozzi & Stoner, 2010). Models, for example, 3D pipelines, being composed of implicit surfaces can be accurately rendered in a virtual globe using ray casting (Z. Wu et al., 2019).

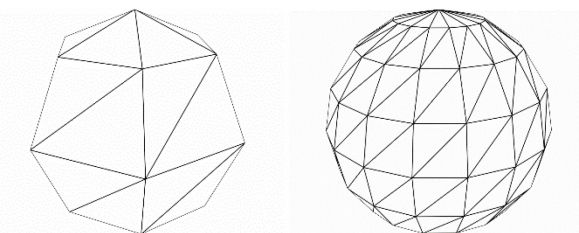


Figure 2.6: Geographic-grid tiling structure of a virtual globe (Cozzi & Ring, 2011)

2.2.2. Topographic 3D maps

The digital terrain surface of virtual globes and other cartographic 3D scenes consists mostly of a uniform triangle mesh, which is simple to store and manipulate, or an irregular triangle mesh, which has a lower number of triangles. To keep the loading and visualizing time at a minimum, the terrain is subdivided at different levels of detail (LODs) in structures like quad trees or triangle bins (Pajarola & Gobbetti, 2007). The tiles are usually preprocessed to reduce the load of the graphics board (Bösch et al., 2009) and

⁶ <https://earth.google.com/>

⁷ <https://www.esri.com/en-us/arcgis/products/arcgis-earth/overview>

requested on demand according to a geometrical error (Crocì et al., 2022). The smaller the distance between the camera and the terrain gets, usually the higher the LOD is. Raster textures can be applied to the terrain for shading mountainous regions or clamping thematic data (Döllner & Hinrichs, 2000). The terrain may be bent to avoid occlusions of important features in panorama or guide maps (Takahashi et al., 2002). In an extreme case, the terrain can be continually elevated to create a hybrid view between a 3D map in the foreground and a 2D map in the background (Lorenz et al., 2008).

On the terrain, 2D vector data (e.g. roads) can be clamped, 3D objects (e.g. buildings) or 3D vector data (e.g. transmission lines) can be placed, or billboard labels (e.g. place names) can be inserted (Figure 2.7). Similarly to the terrain model, vector data draped on terrain can be provided in multiple LODs (Wartell et al., 2003). Several optimisations such as an adaptive subdivision of the TIN to the mapped vector data (Schneider et al., 2005), a detailed rendering of line joins, outlines, and line intersections (Vaaraniemi et al., 2011), and the consideration of physical constraints such as gorges (She et al., 2020) have been proposed to ensure a high-quality cartographic appearance. 3D buildings can be rendered abstractly, for example in a cartoon-like style (Döllner & Walther, 2003), or realistically by applying photo textures (Buchholz, 2006). Also, raster data can be possibly visualised at the facades of the 3D buildings (Trapp & Dollner, 2009). Simplified geometries of buildings, for instance, cylinders, cuboids, or extruded arcs (Maass & Döllner, 2008), or the silhouettes of buildings (Lehmann & Döllner, 2014) may guide the dynamic placement of labels depending on the current view. Alternatively, labels can be attached directly to the facades and roofs of 3D buildings while resolving conflicts of overlapping labels (She et al., 2019).

2D and 3D vector data may be subject of cartographic generalisation. Polylines (Amiraghdam et al., 2020) and polygons (Amiraghdam et al., 2022) clamped on the terrain may be simplified at different LODs in real time. Generalisation tasks for 3D buildings comprise the segmentation and recognition of building structures as well as model and graphic generalisation (Meng & Forberg, 2007). Five LODs for 3D buildings were defined in the first version of the CityGML standard (Kolbe, 2009) and researchers aim at providing smooth transitions between the different levels (Kada et al., 2015). Neighbourhood buildings may be aggregated to uniform cell blocks, which fully enclose the buildings (Glander & Döllner, 2009). Besides generalisation, deformation methods, for instance broadening of routes and scaling of buildings, to avoid occlusions of routes (Qu et al., 2009) and abstraction techniques of topographic map elements have been applied to guide the reader's attention (Semmo et al., 2012).

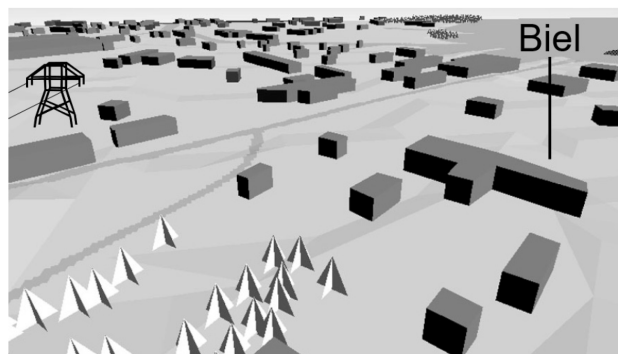


Figure 2.7: Simple topographic 3D map of a city (adapted from Terribilini, 1999)

2.2.3. Thematic 3D maps

Thematic maps represent data by applying visual variables to geometric primitives. On the one hand, points, lines, or polygons can be inserted into the 3D map space. For instance, population density can be depicted as a point cloud, the number of flight passengers can be mapped to curved lines, and employment rates for administrative units can be assigned to floating polygons (Sieber et al., 2013). Afterwards, the geometries are styled (e.g. by size, colour) according to their attribute values. For multivariate data, 2D charts - for example, bar charts, ring charts, or area charts - can be implemented as billboards with anchor lines (Schnürer et al., 2014). Concerning large amounts of linear data, paths of migration or commuter streams may be bundled while avoiding intersections with the terrain and including a LOD system (Thöny & Pajarola, 2015). Lines residing on the ground can be extruded to create wall-like visualisations, for instance, to depict cycles of traffic congestion on certain roads (Tominski & Schulz, 2012). High densities of traffic incidents can be shown as a statistical surface, optionally with a cutting plane (Herman et al., 2018). To visualise quantitative data having a worldwide coverage, different kinds of bars and circles can be added to globes as a whole (Satriadi et al., 2021). Due to the perspective view, it is difficult for the map reader to compare the size of objects in 3D maps. Therefore, a reference grid or scale bars may serve as visual aids (Bleisch, 2011).

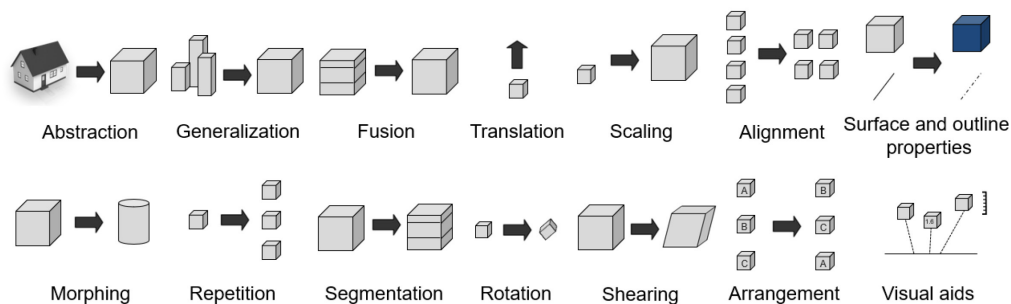


Figure 2.8: Different visualisation techniques for 3D objects and 3D charts on thematic maps (adapted from Schnürer et al., 2015, 2017)

On the other hand, 3D objects can be included in the 3D map space (Figure 2.8). For example, polygons representing different administrative units can be extruded according to the number of irrigation systems and grid cells of a raster layer can be extruded proportionally to the population density, while varying illumination properties (Krisp & Fronzek, 2003). In another work, CO₂ emissions of different countries were represented by scaled cylinders, amongst others (Sandvik, 2008). Changes in size of extruded 2D or 3D geometric primitives are also suited to visualise traffic offences (Herman et al., 2018). Alternatively, visual variables can be applied to parts of existing topographic 3D objects, such as facades and roofs of buildings (Lorenz & Döllner, 2010; Trapp & Dollner, 2009), which may be hierarchically aggregated taking object- and scene-specific properties into account (Vollmer et al., 2018). Voxel maps can be seen as a 3D equivalent of grid maps, and pie charts with individually extruded sectors can be regarded as a 3D equivalent of 2D pie charts (Sieber et al., 2013). Abacus charts, stacked pyramid frustums, and nested hemispheres (Figure 2.9) have been implemented as further types of 3D charts (Schnürer et al., 2015).

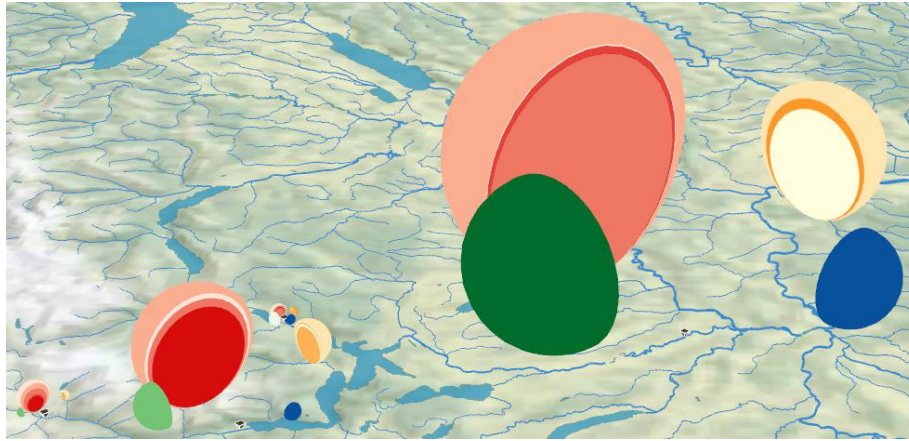


Figure 2.9: Nested hemispheres representing river runoffs in different seasons and time periods (Schnürer et al., 2015)

Not only the earth's surface is of interest for thematic mapping, but also the ocean, the underground, the air, or even other planets. In the ocean, the sea floor can be coloured according to its height and 3D models of remotely operating vehicles can be displayed alongside with their trajectories (McCann, 2004). In the underground, geological layers can be coloured according to their type and possibly sliced to fence diagrams, whereas drill holes can be depicted as 3D cylindrical objects (von Reumont et al., 2013). In the air, flight routes, jet streams, turbulence areas, and clouds can be represented by semi-transparent pipes and polyhedrons (Stähli et al., 2018). On planet Mars, stratigraphic analyses have been conducted (Traxler et al., 2022).

2.2.4. Animated 3D maps

Animations can be divided into temporal animations, depicting the change of spatial data over a period of time, and non-temporal animations, showing spatial data in different graphical representations at one point in time (Dransch, 1997). Potential targets for animations in 3D maps are the location and orientation of 3D objects as well as their size, shape, colour, transparency, or texture. Beyond, light sources and cameras can be animated (Hardisty et al., 2001).

A river flooding a landscape is an example of a temporal animation, that has been implemented by assigning a sequence of textures to the terrain (Döllner & Kersting, 2000). Voxels - having assigned a position, a direction, a velocity, and a volume - have been used to simulate overland flow and soil erosion (Shen et al., 2006). A temporal animation of rigid 3D objects (e.g. cars) requires to detect collisions, for instance, by computing intersections of the object, which can be simplified by hierarchical bounding boxes, and the terrain, which can be represented by a triangle subdivision bi-tree (Shenghua et al., 2008). In a later work (Figure 2.10), also 3D objects with movable parts (e.g. horses) have been animated on the terrain surface (Evangelidis et al., 2018). Temporal animations can occur in real-time, for instance when tracking ships represented by 3D models on the ocean (Ray et al., 2011), or simulating railway infrastructure, such as moving trains and loading cargo (Bogunov & Istomin, 2023).

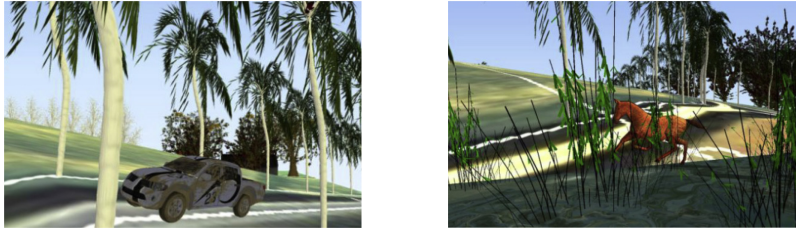


Figure 2.10: Temporal animation (i.e. translation) of 3D objects on the terrain (Evangelidis et al., 2018)

An example of a non-temporal animation is the change in the level of representation of a castle and a forest based on the viewing distance: the closer the distance, the more realistic the representation (Döllner & Kersting, 2000). Non-temporal animations have been also applied to trajectories in combination with colours and textures to depict the direction and speed of aeroplanes (Buschmann et al., 2014). Taking another example of a thematic map object, temperatures on different days in a year have been represented by a revolving 3D helix chart (Kennelly & League, 2015). Further types of animated 3D charts (Figure 2.11), which show transitions of chart-specific and attribute-based styling properties, have been implemented by sphere tracing implicitly defined geometries (Schnürer et al., 2017).

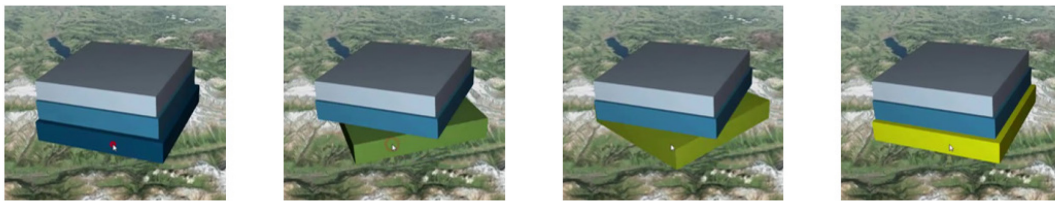


Figure 2.11: Non-temporal animation (i.e. colour, rotation) of one of the stacked cuboids representing glacier volumes in different years (Schnürer et al., 2017)

2.2.5. Interactive 3D maps

Besides monitors for desktop computers, 3D maps can be displayed on the screens of mobile and heads-up devices, which also allow the use of augmented reality (Dickmann et al., 2021) and virtual reality (Keil et al., 2021). Generally, users can interact with the map content via input devices such as a mouse, a keyboard, a touch screen, wired gloves, etc. The input events are registered by the system and corresponding actions are performed (Figure 2.12).

Possible interactions for 3D maps are the change of camera position, the query (i.e. picking) and change of feature attributes, geometric changes (e.g. shift) and feature manipulations (e.g. group), the change of surface properties (e.g. colour), and snapping operations (Fuhrmann et al., 2001). Further interactions are the change of illumination (e.g. azimuth), the overlay of thematic data, and the visualisation of the topic on the Z-axis (Persson et al., 2006). Some interactions such as spatial navigation can be performed directly on the 3D map, whereas others are mostly triggered via a graphical user interface (Cron, 2006). The interactions are possibly supported by visual aids like a 2D mini-map (Zagata et al., 2021).

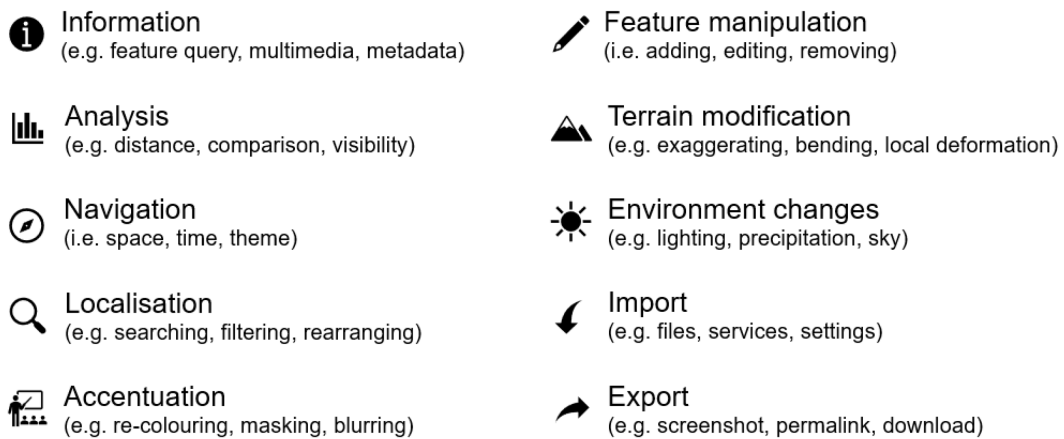


Figure 2.12: Common high-level interactions in 3D atlases and virtual globes (compiled from Cron, 2006; Fuhrmann et al., 2001; Roth, 2012)

Concerning the terrain, curves for bending the surface can be modified in an interactive plot to introduce a progressive projection, which is characterised by a steep viewing angle in the foreground and a flat viewing angle in the background (Jenny et al., 2010). By placing control points on the terrain (Figure 2.13), local regions can be emphasised using an inverse distance interpolation or the moving least squares algorithm (Jenny et al., 2011).

Data lenses, which either highlight or alter the map content within a circular area, are an interactive tool to be applied to the surface of the terrain. These lenses enable showing other thematic data (Döllner & Hinrichs, 2000), different LODs of buildings (Trapp et al., 2008), aggregated temporal information (Tominski et al., 2012), or occluded terrain data (Röhlig et al., 2017).

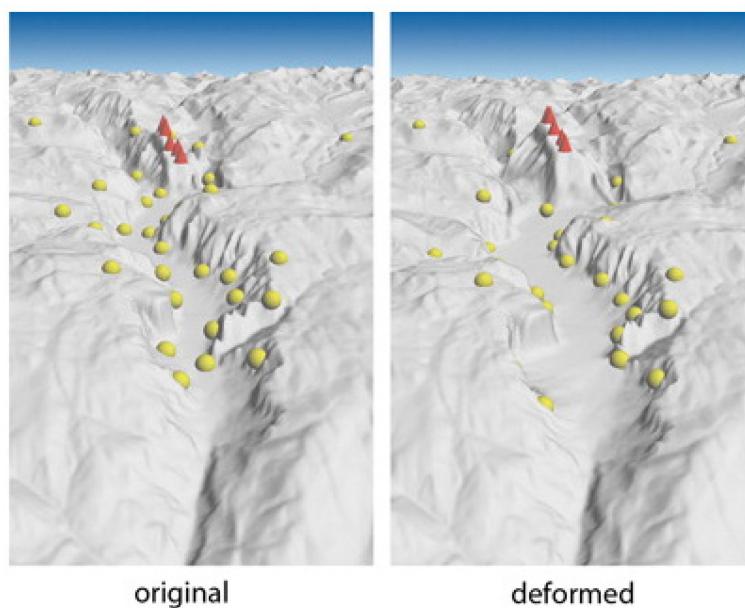


Figure 2.13: Interactive local terrain deformation by control point placement (Jenny et al., 2011)

3D buildings residing on the terrain can be also the subject of interactions. They can be highlighted by changing their colour or reducing the transparency of surrounding buildings (Nebiker, 2003). Other highlighting techniques are outlines, focus-blur effects, or adding glyphs on top (Trapp et al., 2011). Single buildings, certain parts (e.g. floors, roofs) or groups of buildings can be focused by modifying the granularity, the level of abstraction, or the style of textures (Semmo & Döllner, 2015).

Use cases for interactive 3D map environments are manifold. Application fields comprise spatial planning, for instance finding an optimal location for wind turbines (Fuhrmann et al., 2001). Other applications are digital atlases and exploratory data analysis software (Persson et al., 2006), educational games including virtual flights to different cities (Oleggini et al., 2009), or displays for museums, which enable exploring historic castles and churches by interactive dialogues (Matthys et al., 2021).

Note

This section highlighted technical aspects of 3D real-time rendering, but it did not include any cognitive experiments since none were conducted in the scope of this dissertation.

2.3. Machine learning

Models resulting from machine learning algorithms are established by analysing patterns of given data samples. The samples are usually split into training, validation, and test data by a certain ratio (e.g. 60:20:20). The model is created by only inspecting the training data, whereas validation and test data help to evaluate the performance during and after training (Glassner, 2021). The training data is passed usually several times (i.e. epochs) to the system; typically, with multiple samples (i.e. a batch) at once. To increase the training data, augmentation techniques (e.g. rotation, blurring) can be applied to the given data or synthetic samples can be produced.

Three main categories of machine learning are usually distinguished: supervised learning, unsupervised learning, and reinforcement learning. In *supervised learning*, a function is created that maps input to output data, for example, images are assigned categories. The objective during training is to reduce differences between target outputs, also known as labels, and predicted outputs. *Unsupervised learning* methods, such as clustering, do not use labels and exploit the inherent structure of the data. In *reinforcement learning*, which is needed for dynamic environments such as games, agents perform actions. The agents learn to carry out the actions efficiently by receiving either rewards or penalties. (Russell & Norvig, 2016)

Cartographic problems, such as conflating different geometries of geographic data, may be tackled with all three types of machine learning. In a supervised setting, a recurrent neural network can be trained to snap GPS trajectories to roads (J. Feng et al., 2022). It is also possible to match unregistered images by a deformation neural network in an unsupervised manner (S. Wu, Schnürer, et al., 2022). Alternatively, vector data can be aligned stepwise with raster images by reinforcement learning (Duan et al., 2020).

This dissertation is concerned mainly with supervised learning (section 2.6.1), in particular ANNs (section 2.6.2). Machine learning is referred to as deep learning when ANNs consist of numerous layers. The development of deep learning algorithms was supported by leveraging operations on the GPU instead of performing them on the CPU.

2.3.1. Supervised learning

This section provides a selection of frequently used supervised machine learning algorithms for cartographic problems.

The *k-Nearest Neighbours* (k-NN) lookup is suited for classification, whose output is a categorical value, and regression, whose output is a numeric value. A set of labelled data points serves as the source for distance calculations. For classification tasks, the class membership is determined by the plurality of closest neighbours to a query point, whereas the mean or median of the neighbours is calculated for regression tasks (Russell & Norvig, 2016). For instance, the heat vulnerability in cities can be assessed using the k-NN algorithm (Carter & Rinner, 2014).

The *Support Vector Machine* (SVM) is a methodology that constructs maximum-margin hyperplanes to separate classes of higher dimensional features from given example points (Russell & Norvig, 2016). Query points will be binarily classified according to the established decision boundaries, which are often linear. By using SVMs in cartography,

buildings can be classified according to their geometric properties, amongst others (Steiniger et al., 2010).

A hierarchical structure consisting of splitting rules, such as comparisons of attribute values, can be established by a *Decision Tree (DT)*, whose leaf nodes contain the final classes (Mather & Tso, 2009). The C4.5 algorithm is one of several construction possibilities of DTs, where splitting criteria are learned from the training samples according to the highest information gain. An ensemble of multiple DTs, where each of them has access only to a random subset of training data, is called a *Random Forest (RF)*. DTs have been applied for generalisation tasks like selecting settlements for small-scale maps (Karsznia & Weibel, 2018). DTs trained with the C4.5 algorithm helped to identify grid patterns in road networks (Tian et al., 2016). Traffic regulators in GPS trajectories have been detected by RFs (Golze et al., 2020).

A series of supervised learning algorithms is based on probabilistic models (Figure 2.14). *Naïve Bayes (NB)* considers class probabilities, which are conditioned on various attributes while assuming the independence of these variables (Russell & Norvig, 2016). In contrast, a *Bayesian Network (BN)* learns to represent conditional dependencies among variables. A *Hidden Markov Model (HMM)* is a special case of a BN assuming a Markov process, which is a series of events where the probability of an event depends on previous events (Beyerer et al., 2017). The states of an HMM at certain time steps are only indirectly observable by emissions, which appear with certain probabilities. In a *Conditional Random Field (CRF)*, states do not only depend on previous states as in HMMs, which gives more freedom (Russell & Norvig, 2016).

In geoinformation science, locations were extracted from geo-tagged tweets using NB (Eligüzel et al., 2020) and socioeconomic attributes of public transit passengers were derived from the trip attributes by a BN (Faroqi et al., 2020). GPS trajectories were matched to roads by an HMM (Cui et al., 2021) and addresses were linked to a database using a CRF (Comber & Arribas-Bel, 2019).

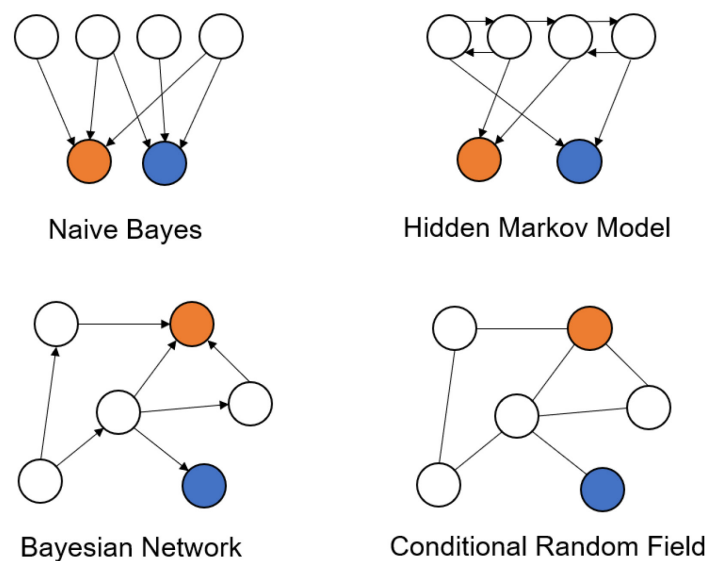


Figure 2.14: Exemplary arrangements of variables/states (= white circles) and observations (= orange and blue circles) for different machine learning models represented by directed (= lines with arrows) and undirected (= lines without arrows) graphs

k-NNs, SVMs, and DTs are non-parametric models, whereas NB, BNs, HMMs, and CRFs are parametric models. Non-parametric models are only based on the given data, while additional parameters, which influence state transitions, are learned in parametric models. Non-parametric models grow with the number of examples, whereas parametric models are restricted to the number of parameters. (Russell & Norvig, 2016)

2.3.2. Artificial neural networks

Originally, an artificial neural network (ANN) consists of neurons, which behave similarly to nerve cells in the human brain. At each neuron, the weighted sum (w) of a number (n) of input values (x) is calculated, a bias value (b) is added, and an activation function (φ) is applied:

$$\text{Formula 1: } N(x) = \varphi\left(\sum_{i=1}^n w_i x_i + b_i\right)$$

According to the universal approximation theorem, any continuous function can be represented by a sufficiently high number of neurons (Nielsen, 2015), which makes ANNs applicable to a broad range of use cases. ANNs are parametric models as they learn weights and biases mainly in a supervised manner. Weights and biases are usually initialised randomly and iteratively adjusted during training to turn the input into the desired output (Glassner, 2021). The difference between predicted and actual target values is indicated by a loss function (e.g. mean-squared error, binary cross-entropy), while adjustments are made by an optimiser (e.g. Gradient Descent, Adam). The speed of learning is known as the learning rate, which can be favoured by normalizing the input and intermediate results. The Rectified Linear Unit (ReLU) is one of the most common activation functions besides sigmoid (σ) and hyperbolic tangent (\tanh), which are mainly used to convert the output to a range between 0 and 1, or -1 and 1. In recent years, different types of ANNs have been elaborated.

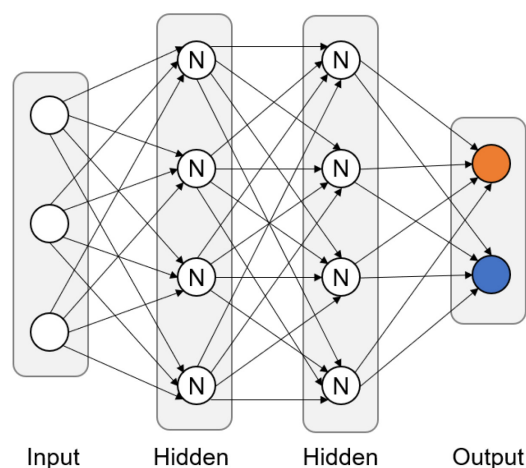


Figure 2.15: An exemplary Multi-Layer Perceptron consisting of three input variables (= white circles), two hidden layers having four neurons each (= white circles with the letter 'N'), and two output values (= orange and blue circles). Values are passed from one unit to the other (= lines with arrows). Formula 1 is applied to each neuron.

A *Multi-Layer Perceptron* (MLP) consists of an input layer, hidden layers, and an output layer, which are formed of multiple neurons (Glassner, 2021). Neurons between layers are fully interconnected for MLPs, but not within a layer (Figure 2.15). The MLP is based on the perceptron, which historically consists of a single neuron with a binary comparison as an activation function without a bias. MLPs are called feed-forward networks when information flows in one direction only (i.e. from input to output). To adjust the network parameters more efficiently, errors are propagated in the opposite direction (i.e. from output to input). MLPs trained with the latter algorithm are also known as *Backpropagation Neural Networks* (BPNNs). During training, neurons in hidden layers are sometimes discarded, which is also known as dropout regularisation, for better generalisability.

A common task of MLPs in cartography is generalisation, such as omission of roads (Q. Zhou & Li, 2017), simplification of administrative boundaries (Olszewski et al., 2018), building simplification (M. Yang, Yuan, et al., 2022), or label displacement (Lan et al., 2022). Other cartographic problems having been examined with MLPs are georeferencing a river on a historic map (Gullu & Narin, 2019), classifying roads based on the users' needs (Mohammadi & Sedaghat, 2021), categorising street vending locations based on attributes and geometries (Barreda Luna et al., 2022), or modelling water richness (Pal & Sarda, 2022) and gully erosion (Saha et al., 2022).

A *Convolutional Neural Network* (CNN) (Figure 2.16) is based on the convolution operation which is the weighted sum of the surrounding pixels of a pixel of an image I :

$$\text{Formula 2: } C(x, y) = \sum_{j=0}^{k-1} \sum_{i=0}^{k-1} I \left[x + i - \left\lfloor \frac{k-1}{2} \right\rfloor, y + j - \left\lfloor \frac{k-1}{2} \right\rfloor \right] * K[i, j]$$

The weights are termed kernel K with size k and act as filters to identify high-level features, such as object parts, and low-level features in images, such as edges. This process is also known as feature extraction and intermediate arrays are also known as feature maps. CNNs also involve downsampling operations to reduce the resolution of the input image and feature maps. Possible downsampling operations are pooling, where the maximum is taken or an average value is calculated, or strided convolutions, where the distance between kernel locations is increased (usually by a step size of 2). The flatten operation transforms a multi-dimensional array into a 1D array, which is passed to a series of fully connected layers, similar to a MLP. (Elgendy, 2020)

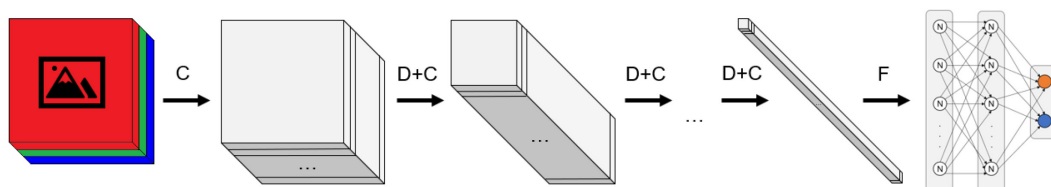


Figure 2.16: Scheme of a simple Convolutional Neural Network. An image with three channels (i.e. RGB) is passed to a series of convolution (C) and downsampling (D) operations as well as a flatten (F) operation, followed by two fully connected layers, which output two categorical values. Formula 2 is applied for the convolution operation.

Deconvolution or transposed convolutions are the inverse operations of a convolution. Deconvolutions with a stride of 2 are inserted into a CNN to upsample feature maps when it is aimed to output an image. Alternatively, feature maps can be bilinearly interpolated during upsampling. The upsampling part is also known as a decoder and the downsampling part as an encoder. Networks trying to reconstruct the input are also known as *Autoencoders*, which consist of an encoder and a decoder. The most compressed feature map is referred to as bottleneck or latent space.

'ResNet' is an exemplary CNN architecture for image classification, where each image is assigned a category, while 'U-Net' is specialised in semantic segmentation, where each pixel in an image is assigned a categorical value. Both networks contain residual or skip connections, which add or concatenate intermediate results of previous layers to preserve identity mappings. A *region-based Convolutional Neural Network* (R-CNN) identifies multiple targets of one or more categories in an image (Elgendy, 2020). Exemplary architectures of R-CNNs are 'Faster R-CNN' for determining bounding boxes, also known as object detection, and 'Mask R-CNN' for pinpointing silhouettes of objects, also known as instance segmentation. Both networks make use of so-called backbone networks for feature extraction (e.g. ResNet), which are subnetworks at the beginning of a network. Subnetworks put at the end are called head networks.

In cartography, CNNs are mainly used for map identification, metadata retrieval, feature extraction, and partially for generalisation, abstraction, feature matching and similarity.

Concerning map identification, maps were distinguished from images by a CNN (J. Li, 2022) and maps were located in images by Faster R-CNN (Oh, 2020). The following metadata elements were retrieved by CNNs or R-CNNs:

- map types (X. Zhou et al., 2018)
- map scale (Touya et al., 2020)
- map extent and state names (Hu et al., 2022)
- regions, map projections, and map themes (J. Li, 2022)

Using CNNs, the following features were extracted from mostly historic maps and DEMs (Figure 2.17):

- points: mountain peaks (Torres et al., 2018), road intersections (Saeedimoghaddam & Stepinski, 2020), symbols (Vassányi & Gede, 2021), spot elevations (Arundel et al., 2022), landmarks (Potié et al., 2022)
- lines: railroads (Chiang et al., 2020), drainage networks (X. Mao et al., 2021), roads (Petitpierre et al., 2021), borders (Ran et al., 2022)
- areas: settlements (Uhl et al., 2017), buildings (Heitzler & Hurni, 2020), surface mines (Maxwell et al., 2020), city structures (Y. Chen et al., 2021), archaeological features (Garcia-Molsosa et al., 2021), hydrographic features (S. Wu, Heitzler, et al., 2022), landforms (Farmakis-Serebryakova et al., 2022)
- labels: various map elements (Weinman et al., 2019)

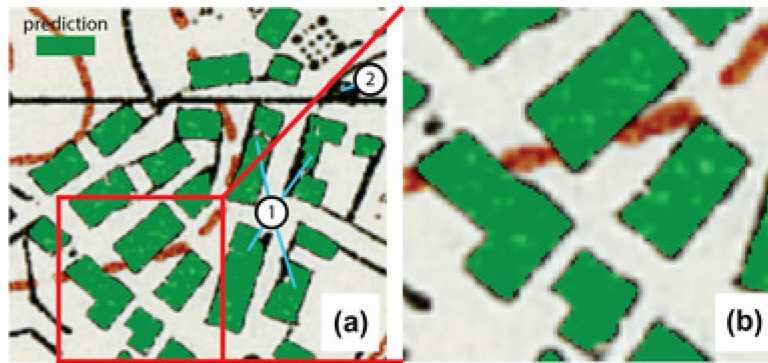


Figure 2.17: Segmented building footprints by a CNN (Heitzler & Hurni, 2020)

Generalisation tasks for CNNs involved the simplification of 2D buildings (Y. Feng et al., 2019), 3D buildings (Y. Wu et al., 2019), roads (Courtial et al., 2020), trajectories (Ruan et al., 2020), coastlines (Du et al., 2022), and contour lines (W. Yu & Chen, 2022). Furthermore, CNNs helped to place labels on areal features such as buildings (Y. Li et al., 2020) and classify the legibility of labels on wayfinding maps (Harrie et al., 2022). Beyond, stylised and abstracted icons for sights were produced by a CNN (Techt, 2020).

Vector railroads and waterlines were aligned to raster maps (Duan et al., 2021), and GPS points were matched to roads (Z. Liu et al., 2022) by CNNs. Lastly, CNNs allowed researchers to calculate similarity scores for cities based on land use (Dobesova, 2020), and between sketch maps and topographic maps (Guo et al., 2022).

A *Generative Adversarial Network* (GAN) usually consists of two parts: 1) A generator produces new observations similar to the original ones, 2) a discriminator learns to distinguish between real and generated observations. Both network parts compete against each other to improve their accuracy during training. MLPs or CNNs may serve as a generator or a discriminator. An exemplary GAN architecture is 'Pix2Pix', which can turn one image into another one while modifying certain properties. Pix2Pix is a type of conditional GAN, where the generator and discriminator are provided with an additional label, which is, in that case, an image to be transformed. (Elgendy, 2020)

Cartographic purposes of GANs are mainly style transfer and inpainting, and partially generalisation.

The following styles were transferred by GANs:

- orthophotos to Google Maps and vice versa (Isola et al., 2017)
- simply styled vector data from OpenStreetMap to Google Maps (Kang et al., 2019)
- CAD drawings to masterplans (X. Ye et al., 2022)
- orthophotos and topographic maps to historical maps and vice versa (Christophe et al., 2022)
- digital elevation models to shaded reliefs (S. Li et al., 2022)

GANs facilitated to complete holes in digital elevation models (Gavriil et al., 2019) and roads (Fang et al., 2022) as well as to generate building footprints (A. N. Wu & Biljecki, 2022). The simplification of coastlines (Du & Wu, 2022), urban areas (Courtial et al.,

2021), and trajectories (X. Yang et al., 2021) were performed as generalisation operations by GANs.

A *Graph Neural Network* (GNN) is particularly suited for graph-structured data consisting of nodes and edges. In contrast to data being arranged in a regular grid, nodes in a graph can have an arbitrary number of neighbours. Data is usually stored in a feature matrix for nodes and an adjacency matrix for the connections of nodes (i.e. the graph structure). There are two typical operations for GNNs: 1) The filtering operation modifies the node features but not the graph structure, whereas 2) the pooling operation coarsens the graph including the node features (Ma & Tang, 2021). A special type of GNN is the *Graph Convolution Network*, which uses convolutional layers for filtering.

In cartography, GNNs are mainly deployed for pattern recognition tasks for buildings, such as differentiating groups (Yan et al., 2019), shapes (Y. Li et al., 2022), and functionalities (X. Xie et al., 2022). Detecting junctions of roads are another application area of GNNs (M. Yang, Jiang, et al., 2022). GNNs are partially used for generalisation problems, such as simplifying roads (Zheng et al., 2021) and buildings (Z. Zhou et al., 2022) as well as selecting points of interest (H. Xie et al., 2022).

A *Recurrent Neural Network* (RNN) is specialised in processing sequential data. RNN units are chained and able to forget information, remember information, and select information. The units store information about frequent changes (i.e. short-term memory) and keep information acquired at earlier timestamps (i.e. long-term memory). Exemplary RNN cells are *Long Short-Term Memory* and the *Gated Recurrent Unit*. (Glassner, 2021)

Cartographic use cases for RNNs are mainly matching toponyms (Santos et al., 2018), addresses (Y. Lin et al., 2020), and location descriptions of roads (Cheng & Chen, 2021).

A *Transformer* is a network originally developed for natural language processing. Relevant parts in sequential data like sentences are being focused in self-attention layers, which are comparable to database queries involving keys and values (Glassner, 2021). Popular Transformer models such as 'BERT' or 'GPT' enable to complete gaps in sentences or to predict follow-up sentences. Recently, Transformers have been applied also for visual tasks like image generation or inpainting.

Related tasks for Transformers in cartography are matching toponyms (Alexis et al., 2020) and addresses (Qian et al., 2020) as well as trajectories of taxis (Jin et al., 2022).

Note

Only pioneering works of artificial neural networks applied in cartography are cited due to the high research activity.

3. Detection of Pictorial Map Objects with Convolutional Neural Networks

Raimund Schnürer¹, René Sieber², Jost Schmid-Lanter³, A. Cengiz Öztireli⁴, Lorenz Hurni⁵

^{1,2,5} Institute of Cartography and Geoinformation, ETH Zurich, Zurich, Switzerland

³ Abteilung Karten und Panoramen, Zentralbibliothek Zürich, Zurich, Switzerland

⁴ Department of Computer Science and Technology, University of Cambridge, Cambridge, United Kingdom

Peer-reviewed journal article

Published online: 11 September 2020

The Cartographic Journal

<https://doi.org/10.1080/00087041.2020.1738112>

Key findings

- Maps can be distinguished from images by Convolutional Neural Networks at very good accuracy rates, though there are some borderline cases in terms of map definition.
- Pictorial maps can be distinguished from other maps by Convolutional Neural Networks at good accuracy rates, though the recognition of very small pictorial objects on maps remains challenging.
- Bounding boxes of sailing ships can be detected on maps by Convolutional Neural Networks at satisfying accuracy rates, though the differentiation of single sailing ships in crowds remains challenging.
- Image input options and hyperparameters affect the accuracy of Convolutional Neural Networks for classifying maps and detecting pictorial objects.

Author contributions

Conceptualisation^{1,2,5}, Methodology¹, Software¹, Investigation¹, Data curation¹, Writing – Original draft¹, Writing – Review & Editing^{1,2,3,4,5}, Visualisation¹, Supervision^{2,4,5}, Project administration^{1,5}, Funding acquisition^{1,5}

Modifications to the original article

Typographic corrections, Citation updates of preprinted to published versions

Abstract

In this work, realistically drawn objects are identified on digital maps by convolutional neural networks. For the first two experiments, 6200 images were retrieved from Pinterest. While alternating image input options, two binary classifiers based on Xception and InceptionResNetV2 were trained to separate maps and pictorial maps. Results showed that the accuracy is 95-97% to distinguish maps from other images, whereas maps with pictorial objects are correctly classified at rates of 87-92%. For a third experiment, bounding boxes of 3200 sailing ships were annotated in historic maps from different digital libraries. Faster R-CNN and RetinaNet were compared to determine the box coordinates while adjusting anchor scales and examining configurations for small objects. A resulting average precision of 32% was obtained for Faster R-CNN and of 36% for RetinaNet. Research outcomes are relevant for crawling map images on the internet and for enhancing the advanced search of digital map catalogues.

3.1. Introduction

An increasing number of maps are available in digital form. Contemporary maps are created almost exclusively using software applications and distributed via electronic devices. Keeping track of these newly published maps is a challenging endeavour: maps need to be identified and distinguished from other types of digital images, such as some curators compile new maps in books (e.g. V. Clarke, 2015) and on blogs (Agarwal, 2019). However, the selection of maps largely depends on personal preference and the amount of depicted maps is limited since the collection process is undertaken manually. Detecting maps automatically would revivify attempts of developing a global search engine for maps (A. Goel et al., 2011). Another challenge is to categorise maps according to thematic, stylistic, or other criteria. Pictorial maps, for example, can be found in specialised books (e.g. A. Antoniou & Kotmair, 2015) or by keyword search on social media websites (e.g. #pictorialmaps, n.d.). In the latter case, when being annotated by laymen, tags might be incorrect though. Recommendation systems for tags (Nguyen et al., 2017) could support the users in creating or validating map metadata.

Libraries digitise their stocks of historic printed maps to preserve these unique documents and to make them accessible over the internet. Historic maps are not only acknowledged as a heritage but also recognised as a rich data source of topographic and socio-geographic features. Map readers, such as historians, may be interested in these features, for example, place names (Jordan et al., 2009), land cover (Fuchs et al., 2015), or illustrations like sea monsters (Duzer, 2014). Given the 'Iconic Turn' (Alloa, 2016) in history research, illustrations within maps and on map borders get into the focus of investigations more often. In many library catalogues, however, the presence of pictorial objects in maps can only be deduced from the title, description, map type, or keywords, if at all. An additional search filter option would enhance these catalogues and facilitate researchers answering semiotic and iconic questions (e.g. Baumgärtner et al., 2019). Considering the ongoing trend of storytelling in cartography (Caquard & Cartwright, 2014), another use case would be to add the detected pictorial objects as protagonists to modern maps. Map objects, like persons or animals, could tell a personal story, give background information to a topic or highlight interesting places on a map, for example for touristic purposes (Graça & Fiori, 2015).

Convolutional neural networks (CNNs) are a promising technology to tackle these challenges. CNNs are a type of artificial neural network (ANN), which is a computational model inspired by the biological neural network of the human brain (Dayhoff, 1990, as cited in Merwin et al., 2009). ANNs are more suited to fulfil complex perceptual tasks than other machine learning methods like support vector machines (Bengio & LeCun, 2007). The increase in parallel computing capabilities of the graphic processing units reduced the training times of ANNs, including CNNs, largely in the past years (Scherer et al., 2010). This enabled a growing number of researchers to experiment with different architectures, for instance, to classify images (Krizhevsky et al., 2012). Other tasks include object detection and image segmentation in domains like autonomous driving (Siam et al., 2017), remote sensing (Zhu et al., 2017), or medicine (Ronneberger et al., 2015). CNNs take images, usually encoded as three-dimensional arrays (i.e. height, width, colour channels), as input. They output arrays of different dimensions and sizes, for example, one-hot encoded categories (i.e. an array containing only zeros except for a single one value), or arrays of the same dimension and size, as it is the case for

autoencoders (Goodfellow et al., 2016). In between, CNNs perform mathematical operations in intermediate layers, such as convolutional, pooling, or fully connected layers (O'Shea & Nash, 2015), to detect patterns in images. Layer parameters, for example, values of convolution matrices (aka kernels), are gradually adapted to minimise differences between the actual and desired outputs. Previous research focused mainly on the recognition of objects in natural images like photos (Girshick et al., 2014) and marginally on man-made images like artwork (Gonthier et al., 2019) or mangas (Yanagisawa et al., 2018), but only scarcely on maps (see Related work).

In this paper, we examine whether CNNs are able to detect objects in pictorial maps. Some of the oldest maps contain pictorial objects, that are realistically depicted symbols and illustrations. The Bedolina map, for example, a rock engraving created in Val Camonica around 1500 BC, shows houses as well as fields with humans and animals as pictograms (Turconi, 1997). Pictorial maps flourished in the Middle Ages and Renaissance when painting and mapmaking were closely related (Rees, 1980, as cited in Kent, 2012). It was the Age of Discovery where monsters, for instance on Ortelius' (1585) *Islandia* map, symbolised the dangers of the sea and the fear of the unknown. In the following decades, pictorial maps declined due to different map materials and production techniques (Wallis & Robinson, 1987), and the growing sense of accurate geographic information (Child, 1956). The next heyday of pictorial maps was in the twentieth century, especially in the United States of America (Hornsby, 2017), which yielded Goodman and Neuhaus' (1930) map of Berkeley, for instance. Illustrative objects enlivened the map by portraying local customs and typical actions. On the other side of the coin, large menacing figures appeared on propaganda maps during the two world wars (Mason, 2016). Today, pictorial objects are often used in maps for tourism and leisure time. One of their purposes is to support underlying topographic features, like cars driving on roads. Graça and Fiori (2015) recommend the usage of pictorial symbols in tourist maps which shall encourage the reader to visit the represented locations. Sarjakoski et al. (2009) give an example of a project using comic-like icons on mobile maps for a national park to "possibly invoke positive emotions" (p. 113). Moreover, pictorial objects are used nowadays to teach map literacy to children. For example, different animals and other agricultural products are represented as pictograms in an agriculture map of a Bulgarian school atlas (Bandrova, 2003). Lastly, maps of fantasy books or in computer games, mimicking the style of the Middle Ages and Renaissance, may also contain pictorial map objects (Lamb & Johnson, 2014).

Pictorial maps follow the typical design process of maps. Child (1956) lists, for example, the purpose of the map, the method of reproduction, colour, projection, and lettering as typical decisions which authors should consider when creating a pictorial map. Child further emphasises and exemplifies pictorial symbols for cultural or manmade features, water, relief, and vegetation. Those "should be clear and simple and if possible recognisable on sight" and the "size of the symbol should not be too large for the scale of the map" (Child, 1956, p. 68). Holmes (1991) states that an outline may not be enough for pictorial objects; therefore, internal shading and patterns may be added complementary. Moreover, Holmes recommends the use of characteristic attributes, for example flying birds, to recognise pictorial symbols. According to Roman (2015), illustrative objects in maps shall be arranged based on the ABC-rule: (A) elements visible at first glance, (B) elements which support A, and (C) elements which support the overall image. Beyond, Roman endorses the four I's as design functions: Identification (= what is the map about), Image (= visual relation to the map), Information (= additional literal

facts), and Incidentals (= engagement of the map reader for the first three functions). While simplicity, distinct outlines and a clear layout may facilitate the recognition of pictorial objects by CNNs, different drawing styles may counteract it.

The overall goal of this work is to provide training datasets and first baselines to detect pictorial objects in maps with CNNs. As one dataset contains also depictions other than maps, maps are first separated from these non-maps with two state-of-the-art CNNs for image classification. To justify this division, we establish a modern definition of maps, which incorporates digital developments of recent years, thus contributing to the ICA Research Agenda (Virrantaus et al., 2009). Moreover, the trained CNNs may help to recognise maps when crawling images on the web. With the same two classifier CNNs, maps are next distinguished according to their level of abstraction, pictorial maps being those with less abstract objects. This differentiation may be used to reveal inconsistencies in pictorial map tags on social media websites, amongst others. For the definition, we relate pictorial maps to decorative, illustrative, and figurative maps. Finally, as an example of a frequently encountered pictorial object on historic maps, sailing ships are identified with two CNNs targeted at object detection. These CNNs output bounding box coordinates of individual ancient ships. These detection results may further enhance the advanced search of digital map libraries when adding sailing ships to the filter options. The development of customised CNN architectures will be the subject of future work to improve the accuracy of the individual tasks.

3.2. Related work

ANNs, which have been often applied in cartography, are self-organizing maps and backpropagation neural networks. Sen et al. (2014), for instance, used a self-organizing map for line generalisation, in particular for omitting rivers at certain scales. Merwin et al. (2009) interpolated values, such as population counts, of areas, where source and target zones are differently subdivided, with a backpropagation neural network. Other examples of ANNs in cartography comprise a particle swarm optimisation neural network (Y. Wang et al., 2015) or a multilayer perceptron and radial basis function network combined with the Weighted Effective Area algorithm (Olszewski et al., 2018). CNNs though have been hardly used in cartography. Duan et al. (2018) extracted railroads and waterlines from historical topographic maps with a fully convolutional network, a CNN which discards fully connected layers. The authors achieved promising results with accuracy rates of 85-93%. Feng et al. (2019) trained CNNs to generalise building footprints at different scales. The authors were successful in producing visually pleasing results and they plan to preserve also rectangularity and parallelisms of buildings in the future. Kang et al. (2019) established a classifier for style-transferred maps which evaluates whether the design characteristics of the original map were preserved. Considering indoor mapping, CNNs helped to find walls (Dodge et al., 2017) and junctions (C. Liu et al., 2017), and to detect objects like doors (Dodge et al., 2017; Ziran & Marinai, 2018) in floor plans. Due to the scarcity of related work regarding maps, we give a brief overview of popular CNNs for the classification of and object detection in natural images since we used some of them in our experiments.

Classification is a task for CNNs aiming to tell what is depicted in an image. One of the first CNNs which solved this task adequately was *AlexNet* (Krizhevsky et al., 2012) consisting of five convolutional layers and three fully connected layers. An improvement

over AlexNet is a series of VGG networks (Simonyan & Zisserman, 2015) which use smaller kernels in the convolutional layers. The most popular versions of these CNNs are *VGG16* and *VGG19*, where 16 and 19 correspond to the total number of convolutional and fully connected layers. The so-called *Inception* modules (Szegedy et al., 2015) were one of the next advancements, which have convolutional layers with different kernel sizes and a pooling layer in parallel. In the following, an identity mapping in parallel to convolutional layers was introduced by *ResNet* (He et al., 2015), leading to better optimisations of changes (= residuals) from input to output. Both networks were combined to *InceptionResNetV2* (Szegedy et al., 2017), which further improved the classification accuracy. An alternative with fewer layers but about the same effectiveness is given by *Xception* (Chollet, 2017). Overall, the accuracy (top 1, single model and single crop) of distinguishing 1000 categories of the ImageNet (Stanford Vision Lab, 2016) dataset on natural images was improved from 62.5% in AlexNet to 79% in Xception and 80.1% in InceptionResNetV2, as reported in the cited articles.

Another task of CNNs is to detect the locations of objects in images. At this, *R-CNN* (Girshick et al., 2013) pioneered by feeding resized bounding box proposals, obtained by the selective search algorithm (Uijlings et al., 2013), into AlexNet. The performance was increased by *Fast R-CNN* (Girshick, 2015), where the image on the whole next to the bounding box proposals are taken as inputs for a classification CNN, namely VGG16. In *Faster R-CNN* (Ren et al., 2017), another iteration, the region proposals are not pre-generated, but predicted by the CNN from anchor boxes. In parallel to the detectors above where the image is processed in multiple stages, CNNs have been developed which output labelled regions in one stage. Prominent examples of these one-stage detectors are *SSD* (Liu et al. 2016), *RetinaNet* (T.-Y. Lin et al., 2020) and *YOLO* (Redmon & Farhadi, 2018). RetinaNet, for instance, concatenates intermediate ResNet layers of different resolutions and upsamples those with lower resolutions. Of the described architectures, YOLO is the fastest (<50 ms), but RetinaNet is the most precise on average (37.8%).

3.3. Experiments

3.3.1. Classification of maps vs. non-maps

Definitions

Over the years, a multitude of map definitions has been established. In our work, we like to apply a modern definition that includes also trends like maps of fictional spaces, indoor maps, and 3D visualisations. Conventional definitions, however, do not take these current developments into account. For example, according to the International Cartographic Association (2003, as cited in Cartwright, 2014), “a map is a symbolized image of geographical reality, representing selected features or characteristics, resulting from the creative effort of its author’s execution of choices, and is designed for use when spatial relationships are of primary relevance” (p. 528). Based on this definition, maps are (static) images which clearly neglect the interactivity introduced by digital mapping. Therefore, Kraak and Fabrikant (2017) tried to establish a new definition by collecting responses of cartographers in a survey. They agreed on the least

common denominator of their suggestions and proposed the definition: “A map is a visual representation of an environment” (p. 6). Clearly, this definition is not as restrictive as previous ones; however, we would deduce that photos, paintings, circuit diagrams, and visualisations of non-spatial environments (e.g. social relationships) would be also counted as maps. For this reason, we would like to introduce a narrower definition of our work:

A map is a scaled-down 2D or 3D representation - optionally animated and interactive - of macroscopic spaces - possibly with additional temporal and thematic information - where features are symbolised and relationships between them are mainly preserved.

As the definition is formed of different aspects, we like to explain briefly our intentions in the following:

Scaled-down: Map scales shall be always smaller than the identical scale (1:1). A map of a model railroad set, for example, would have a very large scale (e.g. 1:5). Upscaled representations, such as circuit diagrams of computers, shall be excluded.

2D or 3D: Maps shall cover 2D planes (e.g. printed map sheets) or 3D spaces (e.g. Augmented Reality maps). We would count 2.5D representations to 3D. 1D representations, however, like stops of a certain bus line or a list of waypoints for route navigation, shall be excluded. We would refer the number of dimensions only to space, separately from time and theme (see additional temporal and thematic information) which are seen by some as 4D and nD representations. We do not distinguish between pseudo- and true 3D, and we would count cartographic 3D representations on 2D surfaces (e.g. on computer screens) as 3D maps.

Representations: Maps shall depict spatial entities in a certain manner (see also Features are symbolised), but they are not those entities themselves.

Animated and interactive: Maps shall include temporal (e.g. glacier motions) and non-temporal animations (e.g. adaptive generalisation when zooming in). Interactivity, which changes the map content by user inputs (e.g. dropdown menu selection), is especially relevant for digital maps.

Macroscopic: Maps shall depict spaces visible to the human eye. Maps of outer space (i.e. celestial maps) and indoor spaces (e.g. flats) would thus be included. Microscopic spaces on a cellular or atomic level would be excluded. Illustrations like the interior of a car or a wardrobe would be a borderline case.

Spaces: Maps shall depict real-world and fictional spaces (e.g. books, computer games, and dreams).

Additional temporal and thematic information: Space-time cubes and thematic maps shall be included. Timelines and mind maps to a certain topic shall be excluded.

Features are symbolised: The creation process of maps from the data model to the visualisation shall follow certain rules and conventions (e.g. styling, generalisation, and projection). This shall exclude paintings, where the painter has more freedom, and photos, which are not abstracted.

Relationships are mainly preserved: The topology of features shall be primarily maintained to allow orientation in space; however, some distortions shall be possible (e.g. cartograms and small displacements). Depictions where features are arranged by other attributes than location, for example when sorting country shapes alphabetically, would rather be infographics.

Data

In total, 3100 maps and 3100 non-maps were collected from Pinterest (n.d.). On this social media website, people can share memorable images. A preview of the image is then shown, and in many cases, a link to the original image source is given. Among those images are a large number of maps varying in time, spatial extent, theme, and style. Since a method to query by text is not offered by the application programming interface (API) of Pinterest, we used Google's (2019) Custom Search API instead to retrieve about 8000 images with the keyword 'illustrated map' and having the site restricted to Pinterest. Maps were then separated manually from non-maps to create training and validation data for the CNNs. We categorised an image as a map when all non-optional criteria of the above definition were fulfilled. In case one of the mandatory requirements was violated, we classified the image as a non-map. Mixtures between maps and non-maps, which are maps or map-related products appearing in the real world or real-world objects placed on maps, were excluded because they fit into both categories and their amount was about nine times less than the collected images of the other two categories. As the number of maps was higher than the number of non-maps, non-maps were enriched with 141 images having the keywords 'illustration', 'sketch', and 'painting'. Another 1569 random non-maps were added by the keyword 'pinimg' since this string is contained in all URLs of Pinterest images. Too closely zoomed maps, duplicate, and very similar images were removed from the search results. The remaining images have a width of 566 pixels and a height of 552 pixels on average. We split the images with a ratio of 60:40 into training and validation sets for the CNNs.

Procedure

We examined the CNNs Xception and InceptionResNetV2 to classify images as either maps or non-maps. These networks take RGB images with a size of 299×299 px as input. As our images exceed the size in either height or width, we tested three methods for feeding images into the networks:

- Resized: Images are resized to the input size without maintaining the aspect ratio.
- Middle random crop: The smaller image side is downscaled to 299px while maintaining the aspect ratio. In case the smaller image side is already less than 299px, this image side is upscaled to 299px while maintaining the aspect ratio. Afterwards, in both cases, a random crop is carried out along the larger image side to reduce the side to 299px.
- Random crop: A 299×299 px random patch is cropped from the image. In case the smaller image side is less than 299px, this image side is first upscaled to 299px while maintaining the aspect ratio and the other image side is reduced to 299px (which is identical to the second middle random crop case).

The Lanczos filter is used for resizing the images. We assume an equal performance of those methods since while the whole image is processed for the first option, undistorted details of images are taken into account in the third option. The second option is a mixture of the first and third options. During training, crops are randomised for each image in each epoch.

Both CNNs are initialised with weights from models pre-trained on the ImageNet dataset and fed with images in batches of 16. The models are retrained for 40 epochs with a learning rate of 10^{-5} , binary crossentropy loss and the Adam optimiser. Retraining one model took about 90 min with an NVIDIA GeForce GTX 1080 graphics board. We used the software library TensorFlow (n.d.) for Python with its high-level API Keras, where Xception and InceptionResNetV2 are pre-implemented.

Results

We averaged the validation results of three Xception and InceptionResNetV2 models, which have achieved the highest accuracy during a training run while changing image options (Table 3.1). For our classification tasks, we define accuracy as the number of all correct predictions divided by the number of all predictions. A prediction is counted as correct when its class score is higher than 0.5. Overall, the accuracies are quite high and only marginally different between the different input options in our experiment. As the accuracies for the random crop are lower than the other two input options, we calculated the accuracy additionally when splitting the image into cells of 299×299 px along a regular grid and averaging results from these cells by applying the retrained model from the random crop. While this approach is more time-consuming, the accuracy is slightly higher than in the first two approaches. We also tried to train the models from scratch instead of initializing them with ImageNet weights; however, this resulted in a significantly lower accuracy. When using a higher learning rate, the loss did not converge that smoothly. Using a 70:30 split between training and validation images led to an alternating loss.

	Xception	InceptionResNetV2
Resized	96.47%	96.60%
Middle random crop	96.52%	96.41%
Random crop		
- random crop	95.50%	95.89%
- average over grid	96.63%	96.76%

Table 3.1: Correct classifications of maps and non-maps for the examined CNNs and image input options (as explained in Procedure). The values are averages of validation accuracies of three retrained models having achieved the highest accuracy during training.

The classification results for a threshold of 0.5 are nearly consistent in the areas under the Receiver Operating Characteristics (ROC) curves for the different image input options (Figure 3.1). ROC curves show the relationship between the true- and false-positive rates for varying classification thresholds. The area under the curve (auc) is 1 in an ideal case. To distinguish between maps and non-maps, averaging the score of image grid cells leads to the largest auc (0.994) and the random crop to the smallest auc (0.991) for both classification models. Interestingly, Xception is slightly more performant than InceptionResNetV2 for the average over the grid calculation considering the auc metric.

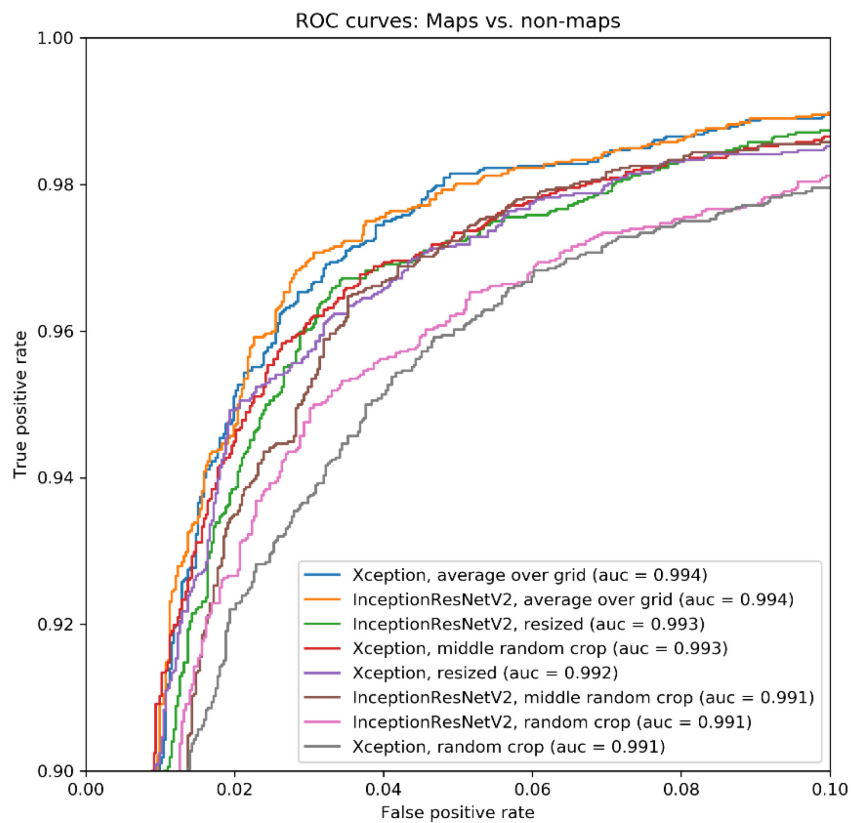


Figure 3.1: ROC curves (enlarged) and auc scores for the tested CNNs and image evaluation options to classify maps and non-maps

As the CNNs achieve a high categorisation accuracy between maps and non-maps, we only show the failure cases as qualitative results. An artistically styled world map, a street and store map of Los Angeles, and a perspective city map of Torun (Figure 3.2) were misclassified in all 12 runs of the two networks for resizing and averaging over the grid. We restricted ourselves to these two methods as they do not involve any randomisation during evaluation. According to our definition, the first example is considered a map because it maintains shapes and spatial relationships between the continents, while the latter example annotates a 3D scene with enlarged buildings. The second example would be counted as a map even with a more conservative definition. Regarding non-maps, the CNNs wrongly categorised a graph showing relations between painters, a collage of letters and telegrams, and US states shaped like a heart (Figure 3.3) in 11 out of 12 runs. As thematic and not spatial relationships are depicted in the first case, and as topology is not preserved in the latter case, we do not count them as maps based on our definition. The second case is clearly no map, even with a broader definition.

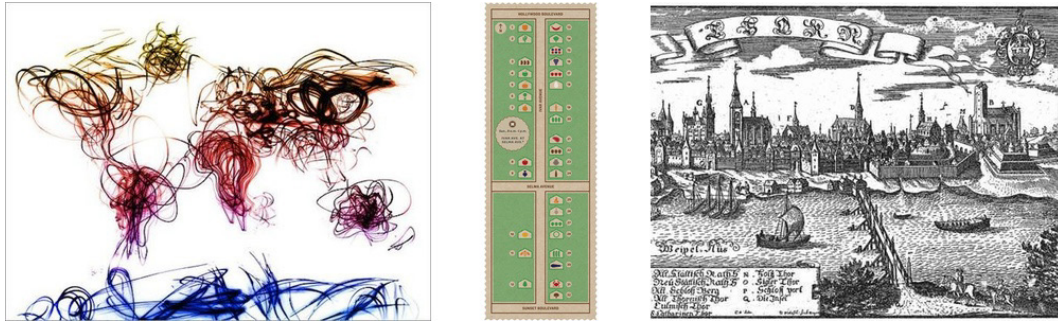


Figure 3.2: The three most frequently misclassified maps by both CNN models for resized and average over grid image evaluation options (image sources: Pinterest^{8, 9, 10})

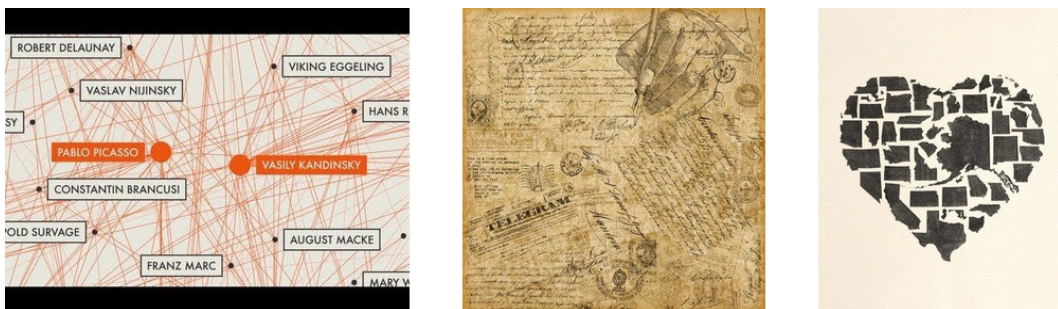


Figure 3.3: The three most frequently misclassified non-maps by both CNN models for resized and average over grid image evaluation options (image sources: Pinterest^{11, 12, 13})

3.3.2. Classification of pictorial maps vs. non-pictorial maps

Definitions

In this section, we characterise pictorial maps and relate them to decorated/decorative, illustrated/illustrative, and figurative maps. “Pictorial’ as opposed to ‘decorative’ maps might be defined as those which are intended primarily for instruction conveyed by means of naturalistic or realistically drawn details” (Child, 1956, p. 6). Wallis and Robinson (1987) state that “topographical information is delineated by more or less realistic drawings, illustrations of features in elevation and small bird’s eye view sketches” (p. 43) in pictorial maps. Both definitions have in common choosing verisimilar (i.e. very close to reality) or indexed (i.e. parametric) representations (Bodum, 2005) for map features. These realistically drawn objects – or alternatively pictures (i.e. paintings or photos) in panels – could be placed inside the map, on the map border, or beside the map. Anthropomorphic maps, where figures coalesce with the map content, would also

⁸ <https://i.pinimg.com/736x/9d/40/93/9d4093cebf375ef698c2022857b83de4--world-map-canvas-world-map-art.jpg>

⁹ <https://i.pinimg.com/736x/5e/94/a1/5e94a1054a88227364c82db48cbbd747--easter--food-design.jpg>

¹⁰ <https://i.pinimg.com/736x/f5/7d/38/f57d38e1acddcf874dcb529fee617290--maps.jpg>

¹¹ <https://i.pinimg.com/736x/bb/4c/52/bb4c5218917d2368937279be05deb528--moma-org-the-artist.jpg>

¹² <https://i.pinimg.com/736x/bb/8d/1b/bb8d1b57149f189eaf3deed58e4a7482.jpg>

¹³ <https://i.pinimg.com/736x/c1/4c/0a/c14c0a9ec176a79addef054c1e134e95--heart-map-my-heart.jpg>

count as pictorial maps. With instruction, Child seems to refer to educative maps for students or interested adults which explain certain topics (e.g. animals of the world, the cuisine of a country, and industries of a city) by pictorial symbols. These thematic maps are missing in Wallis and Robinson's definition as only topographic information is mentioned. In return, it is indicated that pictorial maps could be drawn from an oblique angle, thus panoramic/perspective maps would also be a subset of pictorial maps.

According to Child (1956), maps are decorative when "the composition, lettering and embellishments have all been considered as parts of the design" (p. 32). While we think that composition is important for all kinds of maps, flourishes in lettering may be a distinct property of decorative maps. In addition, embellishments like cartouches and ornamentation seem to occur frequently in decorative maps. Other authors of books (e.g. Barron, 1990; Skelton, 1966) give many examples of decorative maps, but not a clear definition. Theoretically, people may declare maps as decorative when they beautify a place, for instance, a wall map decorating a living room. Illustrated maps are also only vaguely defined. According to Roman (2015), illustrated maps "compress and distort the reality to fit the mental image of a place" (p. 6). A corresponding example would be a touristic map showing selected landmarks of a city. Moreover, illustrative objects could support topographic features, such as parasols on a beach. Illustrated maps are rather artistic than technical because they are created mainly by painters, architects, designers, geographers, historians, or reporters (A. Antoniou & Kotmair, 2015). Lastly, the term figurative map is coined by the title of Minard's (1869) map of Napoleon's Russian Campaign. The map contains numbers (= figures) indicating troop levels but not any images. In other maps like 'The figurative map of Adriaen Block' (1614), a decorated scale bar and compass rose are present. Van Bleyswijck's (n.d.) 'Kart Figuratief' depicts pictorial objects like houses and ships as well as images of places of interest in the city of Delft beside the map. As the term figurative has different meanings, what is reflected in these titles, we can assume that some but not all figurative maps are pictorial maps.

Concluding, we would define pictorial maps as those with verisimilar and indexed representations, which are rather individual than typified. Non-pictorial maps would be those with more abstract representations, which are icons/pictograms, geometric shapes, and labels referring to Bodum (2005). We would use the terms illustrated/illustrative maps synonymously with pictorial maps. We would refer to decorated/decorative maps when labels or elements like title, legend, north arrow, scale bar, and map frame are embellished. When referring to symbols, all maps would be figurative as they convey a certain meaning and they are not meant to be interpreted literally. When referring to images, pictorial maps would be figurative. When referring to creatures, only maps with humans/humanoids, animals, or mythological creatures would be figurative.

Data

From the dataset used in the first experiment, we selected 1500 pictorial maps and 1500 non-pictorial maps. Pictorial maps contain realistically drawn objects (e.g. persons, cars, and houses) and show space in 2D projection or 3D perspective. Non-pictorial maps are 2D representations which include abstract geometries (e.g. points, lines, and polygons), icons, and labels. Maps of both types vary in creation dates, scales, locations, themes,

and styles. We split the maps into training and validation sets with a ratio of 60:40. The maps have a width of 586 pixels and a height of 553 pixels on average.

As we are mainly interested in finding pictorial objects, we excluded 100 of the maps from the first dataset for this experiment. Those are anthropomorphic maps (e.g. Eytzinger and Hogenberg's (1583) *Leo Belgicus*), where pictorial objects cover a large area of the map, and maps showing 3D reliefs without any other pictorial objects (e.g. Berann's (1989) Yosemite panorama). Maps depicting mountains in a molehill manner (e.g. Coronelli's (1690) Abyssinia map) were also excluded as we see molehills as a mixture of an iconic and parametric representation.

Procedure

Similar to the first experiment, we retrained models for Xception and InceptionResNetV2 to categorise a map as either pictorial or non-pictorial. The CNNs use the same hyperparameters (i.e. weights, batch size, learning rate, loss function, and optimiser) as in the first experiment. For feeding the images into the networks, two input options were compared:

- Resized: Map images are resized to 299×299px without maintaining the aspect ratio.
- Manual gridded crop: Map images are partitioned along a regular grid into cells of 299×299px, at which cells may overlap and image sides smaller than 299px are upsampled to this size. Next, cells were manually identified where pictorial objects are present. Only those cells are taken into account as training data for pictorial maps, whereas all grid cells are available as candidates for non-pictorial maps. In every epoch, one cell is selected randomly for each of the pictorial and non-pictorial maps.

Again, the Lanczos filter is used for resizing the images. A middle and a complete random crop are not possible for this experiment since this may lead to regions which do not contain any pictorial objects.

Results

Again, we evaluated three Xception and InceptionResNetV2 models, which have reached the highest validation accuracy during a training run, and averaged their validation results while altering image options (Table 3.2). Overall, the number of correct categorisations between pictorial and non-pictorial maps is 88-92%, hence at a high level, though it is lower than that in the first experiment. In all evaluation options but one (i.e. random crop), Xception is more accurate than InceptionResNetV2. Retraining the classification models with manually identified grid cells containing pictorial objects and applying the retrained models to images with randomly cropped cells led to a decreased accuracy compared to the resizing option. Considering a map as a pictorial one when at least one of the grid cells contains pictorial objects, then the accuracy is similar to the resizing option. When averaging the classification results of image grid cells, the accuracy improved by about 2% in relation to those two options.

	Xception	InceptionResNetV2
Resized	89.64%	88.61%
Manual gridded crop		
- random crop	87.69%	88.00%
- one pictorial cell within grid	89.14%	88.67%
- average over grid	91.89%	90.83%

Table 3.2: Correct classifications of pictorial maps and non-pictorial maps for the examined CNNs and image input options (as explained in Procedure). The values are averages of validation accuracies of three retrained models having achieved the highest accuracy during training.

According to the ROC curves (Figure 3.4), the evaluation option to declare maps as pictorial when at least one of the image grid cells is predicted as pictorial seems to be more suited for higher classification thresholds. Higher thresholds reduce the number of positive outcomes, thus eventually lead to more true negatives but also more false positives. In contrast to a threshold of 0.5, the one pictorial cell within the grid option has a similar performance to averaging the scores over the grid considering the auc scores of the two CNNs. The options to resize or to crop a random part of an image perform similarly to the previous metric. Overall, the auc scores of Xception are higher than those of InceptionResNetV2 for all matching image evaluation options.

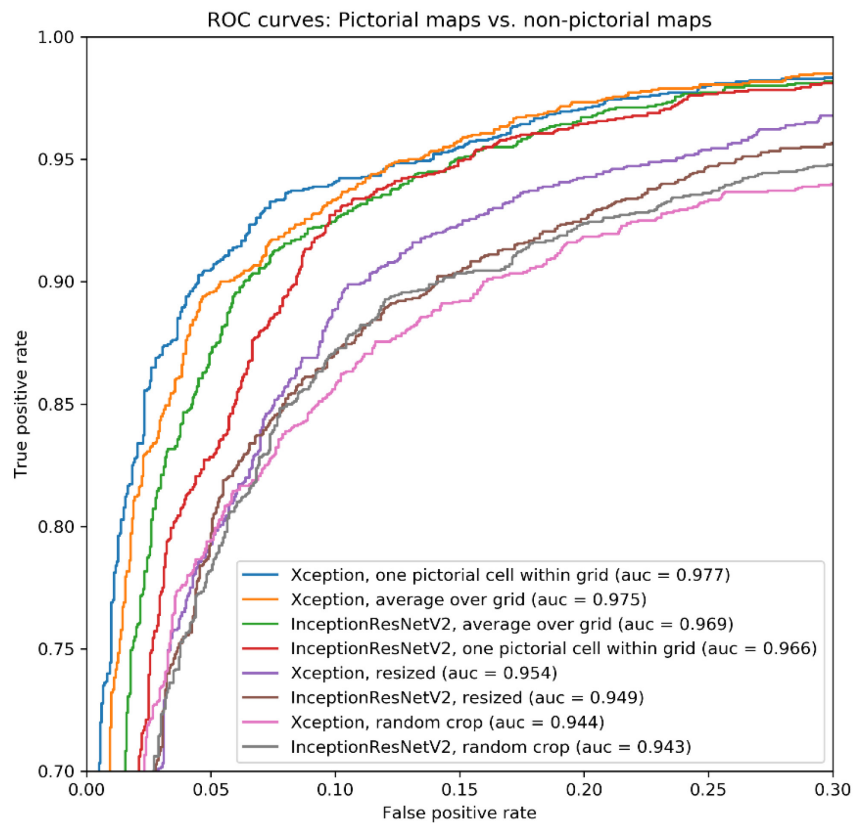


Figure 3.4: ROC curves (enlarged) and auc scores for the tested CNNs and image evaluation options to classify pictorial maps and non-pictorial maps

As the CNNs categorised pictorial and non-pictorial maps mostly successfully, we selected only some failure cases, which occurred in all 12 runs. For evaluation, we selected the same options - resizing and averaging over the grid - as in the first experiment. Examples of frequently misclassified pictorial maps (Figure 3.5) are a subway illustration on a Tokyo metro map, photos on a Beijing city map, and pictorial objects (e.g. lighthouse, rainbow and horse) on an Iceland map. In all three maps, pictorial objects are relatively small. Regarding nonpictorial maps (Figure 3.6), a fantasy indoor map, a Rome city map, and a papercraft world map were often wrongly classified. While the first example may be a borderline case of our definition, it is not clear which activation may have triggered the misclassification of the second and third examples.

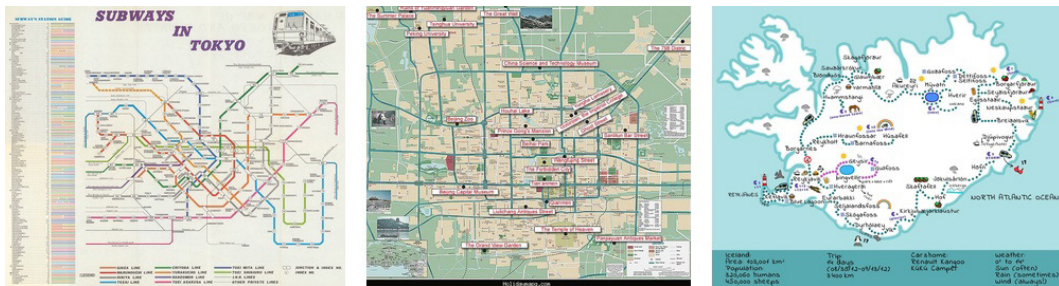


Figure 3.5: Selection of three frequently misclassified pictorial maps by both CNN models for resized and average over grid image evaluation options (image sources: Pinterest^{14, 15, 16})

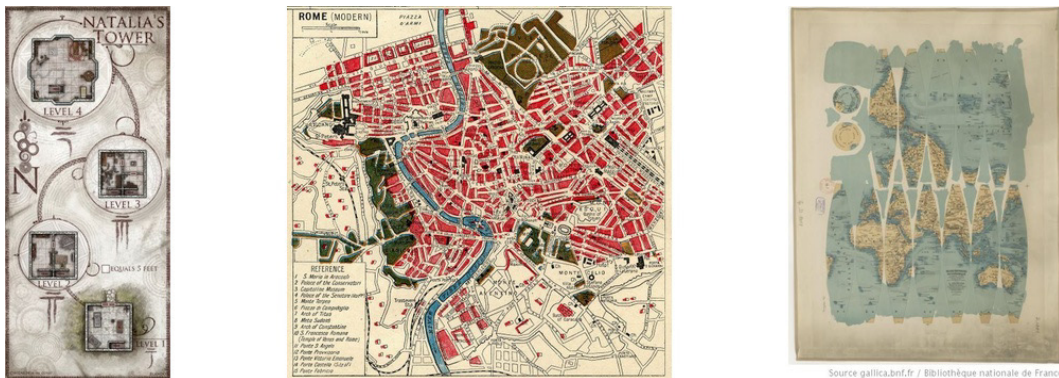


Figure 3.6: Selection of three frequently misclassified non-pictorial maps by both CNN models for resized and average over grid image evaluation options (image sources: Pinterest^{17, 18, 19})

¹⁴ <https://i.pinimg.com/736x/b6/d2/d1/b6d2d1375ac9808cc2998c814862e5d8.jpg>

¹⁵ <https://i.pinimg.com/736x/51/d7/8b/51d78ba2f5a0f8217a20dbb7fe8ca883--nice-map-beijing.jpg>

¹⁶ <https://i.pinimg.com/736x/50/a8/5a/50a85a6ea7ee74f36b5141138d47a281.jpg>

¹⁷ <https://i.pinimg.com/736x/5c/0a/a1/5c0aa1e7bfd262b50d2b6e43c237b833.jpg>

¹⁸ <https://i.pinimg.com/736x/6f/34/73/6f34737d2cc282e23c5668217ddf3544--printable-maps-vintage-printable.jpg>

¹⁹ <https://i.pinimg.com/736x/5f/f7/a2/5ff7a22d4cdf35580c12a654ac208ca5--map-mind-illustrated-maps.jpg>

3.3.3. Detection of sailing ships on maps

Definitions

Ships seem to be one of the most frequently appearing pictorial objects on maps from the late Middle Ages and Renaissance. At those times, ships were mainly used for exploration (e.g. Columbus discovering America), fishing (e.g. with harpoons and nets), trade (e.g. Italian merchants with the Far East), and battles (e.g. in the Anglo-Spanish War 1585–1604). Maps, such as portolan charts, were essential for navigation in all of these nautical endeavours. Map-makers seem to have placed ships on these maps as a symbol for the above uses, to attract attention, and to simply cover empty areas (Reinhartz, 2012). One of the earliest maps which features ships was Cresques’s (1375) Catalan Atlas. A junk and a boat with pearl fishers in the Indian Ocean, a galley near the Canaries, and another ship in the Caspian Sea are depicted on this world map (Unger, 2010). Ships mostly depict common types of the era and rarely represent specific instances – such as Magellan’s Victoria on Ortelius’s (1589) map ‘Maris Pacifici’. As copyright laws were introduced not before the eighteenth century, cartographers often reused images of ships (Reinhartz, 2012), for example, from Bruegel the Elder’s (1565) template showing 16 different ship types.

The word ship, which is a part of our everyday language, is defined as a “large sea-going vessel” and as “a vessel having a bowsprit and three masts” (OED Online, 2022a). For our purposes, we like to define ships as large sea-going vessels with at least one mast but not necessarily a bowsprit. Sails on the masts may be hoisted or lowered, and paddles and flags may be present. Barques, brigs, carracks, clippers, galleys, galleons, or junks would be exemplary ship types which we like to detect with CNNs. We like to differentiate ships from boats which are “small, typically open vessel[s] for travelling over water” (OED Online, 2022a). Commonly, it is distinguished that a ship can carry boats but a boat cannot carry ships. We also like to exclude submarines, which are able to travel underwater, and modern ships. To the latter would count nineteenth-century steam ships (e.g. paddle steamers), for instance, as well as twentieth-century passenger ships (e.g. cruise ships), cargo ships (e.g. container ships), fishing ships (e.g. trawlers), utility ships (e.g. icebreakers), and warships (e.g. aircraft carriers).

Data

We obtained 525 maps and illustrations with 3200 ships from 11 digital map libraries. Most of them were listed and described on the website ‘Map History/History of Cartography’ (Campbell, 2019). A complete list of libraries where we collected the maps is given in Table 3.7 in the Appendix. As only a few libraries offered APIs to search and download maps programmatically, we retrieved the maps mainly by crawling the websites and parsing their HTML content with Python scripts. For smaller collections, we obtained maps manually via the graphical user interface of the websites. If possible, we restricted our search to maps from the fifteenth century to the eighteenth century. The maps have an average width of 1116px and an average height of 907px. We split the maps with a ratio of 60:40 into training and validation sets for the CNNs. There are 294 maps with 1918 ships in the training set and 231 maps with 1283 ships in the validation set. Maps originating from the same digital map library are either in the training or in the validation set, but not in both; 41 maps of the validation set do not contain any ships.

Procedure

We compare two popular CNNs, Faster R-CNN and RetinaNet, to detect bounding boxes of ships in historic maps. Faster R-CNN extracts features²⁰ and proposes regions of interest in the first stage, and predicts bounding box coordinates in the second stage. In RetinaNet, these tasks are performed in one stage using a pyramid of feature maps²¹ with multiple scales. In our experiments, we use TensorFlow implementations of Faster R-CNN and RetinaNet (Gaiser, 2018). We chose ResNet50 as a sub-CNN for feature extraction since models pre-trained on natural images of ResNet architectures with more layers were not available for RetinaNet. When detecting objects in the COCO (2015) natural images dataset with ResNet50, Faster R-CNN achieves an average precision of 30% (J. Huang et al., 2019) and RetinaNet of 35% (Gaiser, 2018). In this metric, a detection is marked as correct when the intersection over the union of a ground-truth bounding box and a predicted bounding box (i.e. area of overlap/area of union) lies above a certain threshold. The average precision, which is the primary COCO metric used for comparisons, is the arithmetic mean for 10 different thresholds ranging from 50% to 95% in steps of 5%. Other COCO metrics consider only a certain threshold (i.e. 50% or 75%) or are applied only to bounding boxes of a certain size (small < 32²px; large > 96²px; medium in between). While COCO metrics were available for the Faster R-CNN implementation when training on custom datasets, we had to calculate them separately for RetinaNet with the Python COCO API. For both CNNs, we included bounding boxes with confidence scores > 0 in the calculations.

For our ship dataset, we trained Faster R-CNN with a learning rate of 10^{-4} for 50 epochs and RetinaNet with a learning rate of 10^{-5} for 30 epochs. Both networks received images in batches of 1 (i.e. single images), and images were flipped randomly along their horizontal axes as the only augmentation technique. Objects within an image are linked to anchors, which are rectangles with different ratios and scales. The anchor centres are distributed in equal intervals (= strides) over the image. We did not modify the predefined anchor ratios (i.e. 2:1, 1:1 and 1:2); however, we tested different anchor scales (see Table 3.3-Table 3.6). The existing Faster R-CNN model was trained on scales of 0.25, 0.5, 1.0 and 2.0, whereas the pre-trained RetinaNet model was set to scales of 2^0 , $2^{1/3}$ and $2^{2/3}$ ($\approx 1.0, 1.26, 1.59$). As ships usually cover only a small area of the image, we optimised the CNNs accordingly:

- Faster R-CNN configuration for small objects: We set the first stage features stride as well as the height and width stride of anchors to 8 instead of 16. The modification of the first stage features stride increases the size of the output feature map of ResNet50 so that more details of smaller objects will be preserved. The change of height and width stride results in smaller differences between anchors centres, thus leading to a finer virtual grid on the images where the anchors will be attached. In total, the number of trainable parameters stays the same.
- RetinaNet configuration for small objects: We used the first four out of five outputs of intermediate layers of ResNet50 instead of the last four. Anchor strides and anchor sizes are halved and feature pyramid levels 2-6 are used

²⁰ Features characterise objects in CNNs; they should not be confused with cartographic or geographic features.

²¹ Feature maps are outputs of intermediate or final layers in CNNs.

instead of 3-7. By this, images are less down-sampled so that details of smaller objects can be better preserved. As a positive side effect, the number of trainable parameters is halved.

On an NVIDIA GTX 1080, one epoch of training our ship dataset took both Faster R-CNN and RetinaNet about 1min30s for the normal configuration. For the configuration for small objects, the training time needed for one epoch increased to 2min20s for Faster R-CNN and to 2min10s for RetinaNet.

Results

We calculated the mean of the highest validation average precisions of three different training runs for Faster R-CNN (Table 3.3) and RetinaNet (Table 3.4) with different anchor scales. The predefined scales of Faster R-CNN reached the third-highest average precision, while the predefined scales of RetinaNet were the second highest. For both CNNs, a reduction of predefined scale values led to the best result for our dataset. Scales with other values resulted in lower average precisions. Here the predefined RetinaNet values scored astonishingly poor for Faster R-CNN. In general, the average precisions of RetinaNet were 8-10% higher than those of Faster R-CNN. We observed an overall increase in average precisions for the Faster R-CNN configuration for small objects (Table 3.5) and diverging results for RetinaNet (Table 3.6). Besides the preset scales of RetinaNet, we note a 7% increase in average precisions for Faster R-CNN and a 1% increase for RetinaNet of the two-scale combinations which achieved also the highest scores with the standard configuration. Three scale combinations for RetinaNet resulted in lower precisions while one remained on about the same level. Still, the top two results of RetinaNet are about 4% higher than the two highest average precisions of Faster R-CNN. Qualitative results show that larger freestanding ships are recognised well (Figure 3.7). With smaller sizes and more occlusions between the ships, however, the detection accuracy drops (Figure 3.8).

Scales	AP	AP ₅₀	AP ₇₅	AP _{small}	AP _{medium}	AP _{large}
1.0, 1.26, 1.59	5.4%	15.35%	2.01%	0.83%	7.75%	12.95%
0.5, 1.0, 1.5	20.24%	48.15%	12.66%	9.34%	26.3%	34.36%
0.25, 0.5, 1.0, 2.0 *	23.4%	55.05%	14.99%	12.82%	30.17%	34.37%
0.25, 0.5, 1.0	23.6%	54.67%	14.91%	12.76%	30.52%	35.06%
0.125, 0.25, 0.5, 1.0	24.93%	56.86%	16.91%	15.15%	31.46%	35.29%
0.0625, 0.125, 0.25, 0.5, 1.0	23.55%	54.91%	15.85%	12.77%	30.35%	34.71%

Table 3.3: Average COCO metrics of the best Faster R-CNN models of three runs for different scales (* = preset)

Scales	AP	AP ₅₀	AP ₇₅	AP _{small}	AP _{medium}	AP _{large}
1.0, 1.26, 1.59 *	34.82%	58.56%	36.99%	20.72%	44.84%	45.25%
0.5, 1.0, 1.5	35.37%	59.74%	37.69%	22.65%	44.35%	44.72%
0.25, 0.5, 1.0, 2.0	33.10%	58.67%	33.44%	21.64%	41.36%	42.99%
0.25, 0.5, 1.0	32.18%	57.62%	32.94%	21.78%	39.34%	41.52%
0.125, 0.25, 0.5, 1.0	32.62%	59.20%	32.39%	21.82%	40.39%	41.04%
0.0625, 0.125, 0.25, 0.5, 1.0	32.95%	59.03%	33.80%	22.35%	40.47%	40.96%

Table 3.4: Average COCO metrics of the best RetinaNet models of three runs for different scales (* = preset)

Scales	AP	AP ₅₀	AP ₇₅	AP _{small}	AP _{medium}	AP _{large}
1.0, 1.26, 1.59	6.7%	15.66%	4.4%	1.6%	9.18%	16.44%
0.5, 1.0, 1.5	27.61%	53.12%	26.2%	14.75%	36.28%	38.38%
0.25, 0.5, 1.0, 2.0 *	31.48%	61.04%	28.91%	18.75%	40.04%	42.68%
0.25, 0.5, 1.0	30.58%	60.57%	26.58%	17.78%	38.76%	42.54%
0.125, 0.25, 0.5, 1.0	32.26%	62.92%	28.97%	19.39%	40.66%	43.44%
0.0625, 0.125, 0.25, 0.5, 1.0	31.77%	61.06%	29.78%	18.39%	40.64%	42.64%

Table 3.5: Average COCO metrics of the best Faster R-CNN models for small objects of three runs configuration and different scales (* = preset)

Scales	AP	AP ₅₀	AP ₇₅	AP _{small}	AP _{medium}	AP _{large}
1.0, 1.26, 1.59 *	35.99%	63.02%	37.04%	25.91%	43.91%	41.10%
0.5, 1.0, 1.5	36.24%	63.35%	38.57%	26.25%	43.65%	42.10%
0.25, 0.5, 1.0, 2.0	28.45%	56.63%	24.45%	19.12%	35.79%	33.48%
0.25, 0.5, 1.0	32.37%	60.19%	31.73%	23.06%	39.87%	35.76%
0.125, 0.25, 0.5, 1.0	30.04%	59.04%	27.65%	20.20%	37.70%	34.11%
0.0625, 0.125, 0.25, 0.5, 1.0	29.68%	58.80%	26.73%	20.70%	36.92%	32.80%

Table 3.6: Average COCO metrics of the best RetinaNet models for small objects of three runs configuration and different scales (* = preset)



Figure 3.7: Ground truth (left) and detected bounding boxes (right) with the best trained Faster R-CNN model (AP: 32.8%) for large, freestanding ships (original image source: Sammlung Ryhiner²²)

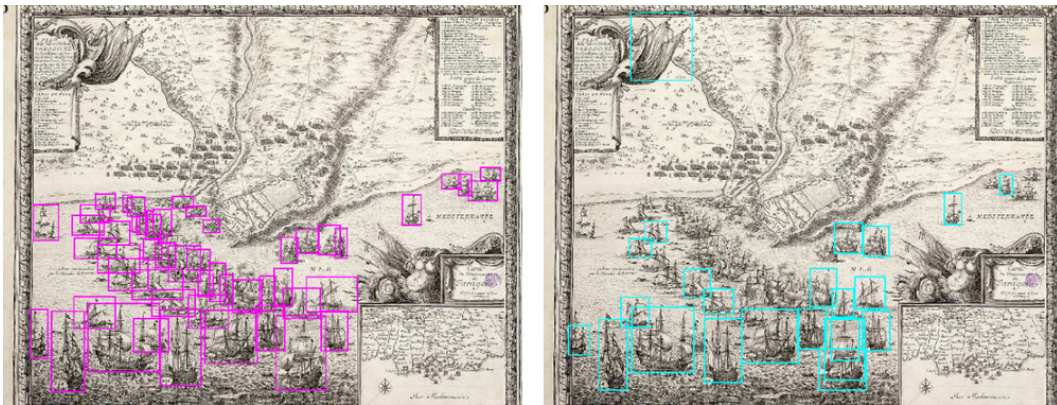


Figure 3.8: Ground truth (left) and detected bounding boxes (right) with the best trained RetinaNet model (AP: 36.8%) for occluded and small ships (original image source: Biblioteca Digital Hispánica²³)

²² https://biblio.unibe.ch/web-apps/maps/zoomify.php?col=ryh&pic=Ryh_3106_1

²³ <http://bdh-rd.bne.es/viewer.vm?id=0000022147>

3.4. Discussion

For our experiments, we prepared datasets for training and evaluating CNNs to detect pictorial map objects. Maps and images in the first dataset originate from Pinterest. We tried to limit similarly styled maps and similar motifs of images to two to three examples since some user-pinned map series and a large number of similar object types (e.g. cars, stitchery). To increase the difficulty, we added other artificial images than maps, such as sketches or invitation cards. The proportion of maps covering Europe and North America is higher than that of other continents; similar seems to be the case for the mapmakers, who are largely unknown on Pinterest. Thus, applying the retrained CNNs to maps of other cultural backgrounds and of other data sources would probably lead to decreased accuracies. The geographic coverage of the second dataset with its ancient sailing ships is similar to the first dataset, but here maps originate from different digital map libraries. The amount of training data is not as high as in other training datasets; however, the categorisation of maps and pictorial maps as well as the annotation of bounding boxes should be consistent as it has been curated by one person. Used definitions are not meant to be carved in stone; they can still be altered and networks fed with other training data accordingly. Although our map definition mentions interactive or animated maps, only static maps were used in our experiments, but theoretically screenshots from 2D or 3D map applications, web map services, or single frames from videos could be also taken as inputs for the CNNs. Concerning map applications and services, we would suggest that rather machine-readable interfaces should be provided to verify their identity and access their source data, which is already partly specified by OGC and ISO standards, than having to parse the map content with CNNs.

Overall, the accuracy of correctly distinguishing maps from non-maps as well as of pictorial maps from nonpictorial maps is quite high with more than 91%. The CNNs show a better performance than other machine learning methods like support vector machines or k-nearest neighbour, which achieved an F1 score (i.e. another metric for accuracy) of 74% for a similar map classification task (A. Goel et al., 2011). With image augmentation techniques - like translation, rotation, and scaling - or model ensembles - where different CNNs models are combined, our obtained accuracy may be increased even more. We did not apply these techniques as we would primarily investigate differences between Xception and InceptionResNetV2. Although newer CNN models exist with better accuracies, we chose those models as they share the same input size of images and they are available in the same library and programming language. Similar applies to compare Faster R-CNN and RetinaNet as both have the same backbone (i.e. ResNet50) and backend (i.e. TensorFlow) but different APIs. Faster R-CNN is harder to modify because it uses a lower-level API, that is why only configuration parameters were changed and not the network architecture as done for RetinaNet for small objects. Quantitative results from the object detection task cannot be directly compared with those of the classification tasks as we reported the values in the common COCO format. Qualitative results are convincing when ships are not too crowded, too small, or too blurry. How these difficulties may be solved is addressed in the future work section.

Our classification experiments to identify maps and pictorial maps resulted in a speed-accuracy trade-off: While feeding resized images into the CNNs is faster than splitting the images into grid cells and inputting those for validation, averaging the predicted classification scores of the cells is more accurate. The input of 299px images for

classification tasks is justified by the CNN architectures. Enlarging the input size would increase the number of learnable parameters, which may lead to memory shortages on the graphic board. Current attempts like EfficientNet (Tan & Le, 2019) optimise the number of parameters, but they rather target improving the performance than feeding in high-resolution images. We would argue that it is not necessary to input maps at high resolutions because humans can also recognise maps at some distance without inspecting all details. Only if details were important, higher resolutions would be advantageous, for instance, to identify small pictorial objects. This may explain why the accuracy of cropping images was higher than the resizing strategy in our second experiment, whereas the tested input strategies had a similar accuracy in the first experiment.

Only the resizing option was available in the object detection libraries; however, other hyperparameters like anchor scales and strides could be tuned. It cannot be excluded that other anchor scale values yield better results since ours were manually determined, partly with the help of a debugging tool for RetinaNet. Reducing the anchor strides increased the detection accuracy of Faster R-CNN; however, it was not possible to reduce the strides further due to constraints in the network configuration. The adaptation to detect smaller objects with RetinaNet also increased the accuracy and reduced the number of trainable parameters. Limiting the number of ResNet50 levels to three would have caused a reduction of levels in the feature pyramid network, which we assume is not desirable since the factor between the smallest resized (2×2 px) and largest resized ship (298×345 px) is about 2^7 to 2^8 . The size of the smallest resized ship also demonstrates that it is only barely detectable for the CNNs. It is not clear at this point if both configurations for small objects can be combined in one or both CNNs to increase the accuracy even more.

3.5. Summary and future work

In this paper, we examined identifying pictorial objects in historic and contemporary maps with CNNs. We reached an accuracy of about 97% to classify maps and non-maps with Xception and InceptionResNetV2. With about 92%, the accuracy was lower to distinguish between pictorial and non-pictorial maps. For the first task, the accuracy of Xception and InceptionResNetV2 was about the same, for the second task Xception was slightly more accurate than InceptionResNetV2. From the examined input options, calculating the average over regular image grid cells achieved the highest accuracy; however, this method is more computationally intensive than resizing the images. An average precision of about 32% could be obtained with Faster R-CNN and of about 36% with RetinaNet, both having ResNet50 as a backbone, to recognise sailing ships in maps. Configuring the networks to detect small objects increased the accuracy for both CNNs, in the case of Faster R-CNN more than that of RetinaNet. Reducing the anchor scale values from the original setup led also to higher accuracies. With our modifications, the average precision is slightly higher than the baselines for detecting objects in natural images with these networks.

Future work may extend our datasets with additional training data, for example by harvesting images from other websites or by synthetically creating special cases with maps in real-world images or zoomed-in maps. Also, other map types than pictorial maps could be classified, for instance, based on the visualisation type (e.g. chart maps),

the dimensionality (2D/3D) or the level of representation (abstract-realistic). Map types requiring a semantic understanding of the content, like usage (e.g. hiking) or theme (e.g. weather), would go beyond the visual recognition capabilities of CNNs though. For object detection, datasets with other types of pictorial objects could be prepared, such as persons, animals, or sea monsters. Eventually, these objects could be detected class-agnostically with weakly supervised methods (Gonthier et al., 2018). Another ability of CNNs is to detect visually salient objects (Borji et al., 2015), which could help cartographers to quantify A-level pictorial objects according to Roman's (2015) ABC rule.

Our detection accuracy may be improved by special CNN architectures for small objects (Eggert et al., 2017) as well as for crowded and occluded objects (Wang et al. 2018). Also, new trends like dilated (Hamaguchi et al., 2018) and deformable convolutions (Dai et al., 2017) may further increase the accuracy. Even hyperparameters may be optimised, and CNNs architectures may be created automatically (Zoph & Le, 2017). Novel CNN architectures like Mask R-CNN (He et al., 2017) and DeepLab (L. Chen et al., 2018) would be able to extract not only bounding boxes but also silhouettes of objects. Similarity metrics (Krizhevsky et al., 2012) could enable finding map series with a certain style of an author (e.g. artist, map agency) or maps produced with the same software. Calculating similarity metrics for single map objects (e.g. ships) would facilitate detecting duplicates, which could reveal hidden relationships between ancient cartographers. In combination with a metric on map readability (i.e. how accurately can map features be extracted), map producers could develop a style which is well-readable, yet distinguishable from others.

The overarching goals for cartographic research on CNNs would be to identify maps, to vectorise, georeference, and attribute them in a first step, and extract metadata in a second step. Similarity metrics to other maps or map objects could be derived from the CNN outputs in a third step. This would allow creating a global search engine, which indexes maps on the internet. Next to a simple text-based search, more sophisticated search filters could be provided, for example for map features (e.g. rivers and place names) or metadata (e.g. coordinate reference system and map style). Tools like an inverse map search or recommendations of similar maps are also thinkable due to the similarity metrics. With our three experiments, we contributed to finding maps on the internet as well as to extracting data (i.e. certain map objects) and metadata (i.e. a certain map type).

Appendix

Library	Website	Maps used
Beinecke Rare Book & Manuscript Library	https://brbl-dl.library.yale.edu/	159
Biblioteca Digital Hispánica	http://www.bne.es/	35
Bibliothèque Nationale de France	https://gallica.bnf.fr/	63
Bodleian Library	https://digital.bodleian.ox.ac.uk/	16
Norman B. Leventhal Map & Education Center	https://collections.leventhalmap.org/	17
David Rumsey Map Collection	https://www.davidrumsey.com/	33
John Carter Brown Library	https://jcb.lunaimaging.com/	33
Library of Congress	https://www.loc.gov/	6
New York Public Library	https://www.nypl.org/	7
Royal Museum Greenwich	https://pro.europeana.eu/	7
Sammlung Ryhiner	https://www.unibe.ch/universitaet/dienstleistungen/universitaetsbibliothek/recherche/sondersammlungen/kartensammlungen/index_ger.html	149

Table 3.7: Digital libraries from which historic maps with sailing ships were retrieved for training Faster R-CNN and RetinaNet

References

- Agarwal, A. (2019). *Top 40 Cartography Blogs & Websites for Cartographers To Follow in 2019*. Feedspot Blog. http://blog.feedspot.com/cartography_blogs/
- Alloa, E. (2016). Iconic Turn: A Plea for Three Turns of the Screw. *Culture, Theory and Critique*, 57(2), 228-250. <https://doi.org/10.1080/14735784.2015.1068127>
- Antoniou, A., & Kotmair, A. A. (2015). *Mind the map: Illustrated Maps and Cartography*. Gestalten.
- Bandrova, T. (2003). Atlas Rodinoznanie. *International Research in Geographical and Environmental Education*, 12(4), 354-358. <https://doi.org/10.1080/10382040308667547>
- Barron, R. (1990). *Decorative Maps*. Crescent Books.
- Baumgärtner, I., Debby, N. B.-A., & Kogman-Appel, K. (2019). *Maps and Travel in the Middle Ages and the Early Modern Period: Knowledge, Imagination, and Visual Culture*. De Gruyter.
- Bengio, Y., & LeCun, Y. (2007). Scaling learning algorithms towards AI. *Large-Scale Kernel Machines*, 34(5), 1-41.
- Berann, H. (1989). *Panoramic drawing of the Yosemite National Park* [Map]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Heinrich_Berann_NPS_Yosemite.jpg
- Block, A. (1614). *The figurative map of Adriaen Block* [Map]. The New York Public Library. <http://digitalcollections.nypl.org/items/510d47d9-7bf7-a3d9-e040-e00a18064a99>
- Bodum, L. (2005). Modelling Virtual Environments for Geovisualization: A Focus on Representation. In J. Dykes, A. M. MacEachren, & M. J. Kraak (Eds.), *Exploring Geovisualization* (pp. 389-402). Elsevier. <https://doi.org/10.1016/B978-008044531-1/50437-1>
- Borji, A., Cheng, M.-M., Jiang, H., & Li, J. (2015). Salient object detection: A benchmark. *IEEE Transactions on Image Processing*, 24(12), 5706-5722. <https://doi.org/10.1109/TIP.2015.2487833>
- Brueghel the Elder, P. (1565). *Sixteen Boats of Different Structure* [Map]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Pieter_Brueghel_I_-_Sixteen_Boats_of_Different_Structure,_c._1565_RP-P-1997-159.jpg
- Campbell, T. (2019). *Map History / History of Cartography*. <https://www.maphistory.info/>
- Caquard, S., & Cartwright, W. (2014). Narrative Cartography: From Mapping Stories to the Narrative of Maps and Mapping. *The Cartographic Journal*, 51(2), 101-106. <https://doi.org/10.1179/0008704114Z.000000000130>
- Cartwright, W. (2014). Rethinking the definition of the word 'map': An evaluation of Beck's representation of the London Underground through a qualitative expert survey. *International Journal of Digital Earth*, 8(7), 522-537. <https://doi.org/10.1080/17538947.2014.923942>
- Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4), 834-848. <https://doi.org/10.1109/TPAMI.2017.2699184>
- Child, H. (1956). *Decorative Maps* (1st edition). Studio Publications.
- Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1800-1807. <https://doi.org/10.1109/CVPR.2017.195>
- Clarke, V. (2015). *Map: Exploring the World*. Phaidon Press Limited. <https://www.goodreads.com/book/show/25208271-map>
- COCO Consortium. (2015). *Common Objects in Context (COCO)*. <http://cocodataset.org>

- Coronelli, V. M. (1690). *Map of Ethiopia, Abyssinia, and the Source of the Blue Nile* [Map]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:1690_Coronelli_Map_of_Ethiopia,_Abyssinia,_and_the_Source_of_the_Blue_Nile_-_Geographicus_-_Abissinia-coronelli-1690.jpg
- Cresques, A. (1375). *Catalan Atlas* [Map]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:1375_Atlas_Catalan_Abraham_Cresques.jpg
- Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., & Wei, Y. (2017). Deformable Convolutional Networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, 764-773. <https://doi.org/10.1109/ICCV.2017.89>
- Dodge, S., Xu, J., & Stenger, B. (2017). Parsing floor plan images. *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, 358-361. <https://doi.org/10.23919/MVA.2017.7986875>
- Duan, W., Chiang, Y.-Y., Knoblock, C. A., Uhl, J. H., & Leyk, S. (2018). Automatic Generation of Precisely Delineated Geographic Features from Georeferenced Historical Maps Using Deep Learning. *AutoCarto 2018*. <https://cartgis.org/autocarto/autocarto-2018/>
- Duzer, C. V. (2014). *Sea Monsters on Medieval and Renaissance Maps* (Reprint edition). British Library.
- Eggert, C., Zecha, D., Brehm, S., & Lienhart, R. (2017). Improving Small Object Proposals for Company Logo Detection. *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, 167-174. <https://doi.org/10.1145/3078971.3078990>
- Eytzinger, M., & Hogenberg, F. (1583). *Leo Belgicus* [Map]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:1583_Leo_Belgicus_Hogenberg.jpg
- Feng, Y., Thiemann, F., & Sester, M. (2019). Learning Cartographic Building Generalization with Deep Convolutional Neural Networks. *ISPRS International Journal of Geo-Information*, 8(6), 258. <https://doi.org/10.3390/ijgi8060258>
- Fuchs, R., Verburg, P. H., Clevers, J. G. P. W., & Herold, M. (2015). The potential of old maps and encyclopaedias for reconstructing historic European land cover/use change. *Applied Geography*, 59, 43-55. <https://doi.org/10.1016/j.apgeog.2015.02.013>
- Gaiser, H. (2018). *Keras RetinaNet*. <https://github.com/fizyr/keras-retinanet>
- Girshick, R. (2015). Fast R-CNN. *2015 IEEE International Conference on Computer Vision (ICCV)*, 1440-1448. <https://doi.org/10.1109/ICCV.2015.169>
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 580-587. <https://doi.org/10.1109/CVPR.2014.81>
- Goel, A., Michelson, M., & Knoblock, C. A. (2011). Harvesting maps on the web. *International Journal on Document Analysis and Recognition (IJ DAR)*, 14(4), 349-372. <https://doi.org/10.1007/s10032-010-0136-2>
- Gonthier, N., Gousseau, Y., Ladjal, S., & Bonfait, O. (2019). Weakly Supervised Object Detection in Artworks. In L. Leal-Taixé & S. Roth (Eds.), *Computer Vision - ECCV 2018 Workshops* (pp. 692-709). Springer International Publishing. https://doi.org/10.1007/978-3-030-11012-3_53
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Autoencoders. In *Deep Learning*. MIT Press.
- Goodman, M. B., & Neumaus, E. (1930). *A map of Berkeley, Oakland & Alameda* [Map]. David Rumsey Map Collection. <https://www.davidrumsey.com/luna/servlet/detail/RUMSEY~8~1~268595~90042837:A-map-of-Berkeley,-Oakland-&-Alamed>
- Google Developers. (2019). *Custom Search JSON API*. Google. <https://developers.google.com/custom-search/v1/overview>

- Graça, A. J. S., & Fiori, S. R. (2015). Proposal for a Tourist Web Map of the South Area of Rio: Cartographic Communication and the Act of Representing the Landscape in Different Scales and Levels of Abstraction. *Revista Brasileira de Cartografia*, 67(5), 1079-1090. <https://doi.org/10.14393/rbcv67n5-44629>
- Hamaguchi, R., Fujita, A., Nemoto, K., Imaizumi, T., & Hikosaka, S. (2018). Effective Use of Dilated Convolutions for Segmenting Small Object Instances in Remote Sensing Imagery. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1442-1450. <https://doi.org/10.1109/WACV.2018.00162>
- Harley, J. B. (1989). Deconstructing the map. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 26(2), 1-20. <https://doi.org/10.3138/E635-7827-1757-9T53>
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2980-2988. <https://doi.org/10.1109/ICCV.2017.322>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778. <https://doi.org/10.1109/CVPR.2016.90>
- Holmes, N. (1991). *Pictorial Maps: 'History, Design, Ideas, Sources'*. Watson-Guptill.
- Hornsby, S. J. (2017). *Picturing America: The Golden Age of Pictorial Maps* (1st edition). University of Chicago Press.
- Huang, J., Rathod, V., Votel, R., Chow, D., Sun, C., Zhu, M., Fathi, A., & Lu, Z. (2019). *Tensorflow Object Detection API*. <https://github.com/tensorflow/models>
- Jordan, P., Bergmann, H., Cheetham, C., & Hausner, I. (2009). *Geographical Names as a Part of the Cultural Heritage* (Vol. 18). Institut für Geographie und Regionalforschung der Universität Wien.
- Kang, Y., Gao, S., & Roth, R. E. (2019). Transferring multiscale map styles using generative adversarial networks. *International Journal of Cartography*, 5(2-3), 115-141. <https://doi.org/10.1080/23729333.2019.1615729>
- Kent, A. J. (2012). From a Dry Statement of Facts to a Thing of Beauty: Understanding Aesthetics in the Mapping and Counter-Mapping of Place. *Cartographic Perspectives*, 73, 39-60. <https://doi.org/10.14714/CP73.592>
- Kraak, M.-J., & Fabrikant, S. I. (2017). Of maps, cartography and the geography of the International Cartographic Association. *International Journal of Cartography*, 3(sup1), 9-31. <https://doi.org/10.1080/23729333.2017.1288535>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90. <https://doi.org/10.1145/3065386>
- Lamb, A., & Johnson, L. (2014). Middle Earth to Panem: Maps of Imaginary Places as Invitations to Reading. *Teacher Librarian*, 42(1), 60-63. <https://link.gale.com/apps/doc/A387953050/AONE>
- Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2020). Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2), 318-327. <https://doi.org/10.1109/TPAMI.2018.2858826>
- Liu, C., Wu, J., Kohli, P., & Furukawa, Y. (2017). Raster-to-Vector: Revisiting Floorplan Transformation. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2214-2222. <https://doi.org/10.1109/ICCV.2017.241>
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer Vision - ECCV 2016* (pp. 21-37). Springer International Publishing. https://doi.org/10.1007/978-3-319-46448-0_2
- Mason, B. (2016). *These Colorful Propaganda Maps Fueled 20th-Century Wars*. National Geographic News. <https://news.nationalgeographic.com/2016/10/propaganda-war-maps-gallery/>

- Merwin, D., Cromley, R., & Civco, D. (2009). A Neural Network-based Method for Solving “Nested Hierarchy” Areal Interpolation Problems. *Cartography and Geographic Information Science*, 36(4), 347–365. <https://doi.org/10.1559/152304009789786335>
- Minard, C. (1869). *Carte Figurative des pertes successives en hommes de l’armée française dans la campagne de Russie 1812-1813* [Map]. Wikimedia Commons. <https://commons.wikimedia.org/wiki/File:Minard.png>
- Nguyen, H. T. H., Wistuba, M., & Schmidt-Thieme, L. (2017). Personalized Tag Recommendation for Images Using Deep Transfer Learning. In M. Ceci, J. Hollmén, L. Todorovski, C. Vens, & S. Džeroski (Eds.), *Machine Learning and Knowledge Discovery in Databases* (pp. 705–720). Springer International Publishing.
- OED Online. (2022a). *Boat, n.; ship, n.* Oxford University Press. <https://www.oed.com/>
- Olszewski, R., Gnat, M., & Fiedukowicz, A. (2018). Artificial neural networks and fuzzy inference systems for line simplification with extended WEA metric. *Geodesy and Cartography*, 67(2), 255–269. <https://doi.org/10.24425/118708>
- Ortelius, A. (1585). *Islandia* [Map]. Wikimedia Commons. [https://commons.wikimedia.org/wiki/File:Islandia_\(Abraham_Ortelius\).jpg](https://commons.wikimedia.org/wiki/File:Islandia_(Abraham_Ortelius).jpg)
- Ortelius, A. (1589). *Maris Pacifici* [Map]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Ortelius_-_Maris_Pacifici_1589.jpg
- O’Shea, K., & Nash, R. (2015). *An Introduction to Convolutional Neural Networks*. arXiv. <https://doi.org/10.48550/arXiv.1511.08458>
- #pictorialmaps. (n.d.). Instagram. <https://www.instagram.com/explore/tags/pictorialmaps/>
- Pinterest. (n.d.). Pinterest. <https://www.pinterest.com/>
- Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv. <https://doi.org/10.48550/arXiv.1804.02767>
- Reinhartz, D. (2012). *The art of the map: An illustrated history of map elements and embellishments*. Sterling.
- Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
- Roman, J. (2015). *The Art of Illustrated Maps: A Complete Guide to Creative Mapmaking’s History, Process and Inspiration* (1st edition). HOW Books.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *Medical Image Computing and Computer-Assisted Intervention* (pp. 234–241). Springer International Publishing. https://doi.org/10.1007/978-3-319-24574-4_28
- Sarjakoski, L. T., Sarjakoski, T., Koskinen, I., & Ylirisku, S. (2009). The Role of Augmented Elements to Support Aesthetic and Entertaining Aspects of Interactive Maps on the Web and Mobile Phones. In W. Cartwright, G. Gartner, & A. Lehn (Eds.), *Cartography and Art* (pp. 107–122). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-68569-2_10
- Scherer, D., Schulz, H., & Behnke, S. (2010). Accelerating Large-Scale Convolutional Neural Networks with Parallel Graphics Multiprocessors. In K. Diamantaras, W. Duch, & L. S. Iliadis (Eds.), *Artificial Neural Networks - ICANN 2010* (pp. 82–91). Springer. https://doi.org/10.1007/978-3-642-15825-4_9
- Sen, A., Gokgoz, T., & Sester, M. (2014). Model generalization of two different drainage patterns by self-organizing maps. *Cartography and Geographic Information Science*, 41(2), 151–165. <https://doi.org/10.1080/15230406.2013.877231>
- Siam, M., Elkerdawy, S., Jagersand, M., & Yogamani, S. (2017). Deep semantic segmentation for automated driving: Taxonomy, roadmap and challenges. *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 1–8. <https://doi.org/10.1109/ITSC.2017.8317714>

- Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *ICLR 2015*. <https://iclr.cc/archive/www/doku.php%3Fid=iclr2015:main.html>
- Skelton, R. A. (1966). *Decorative Printed Maps of the 15th to 18th Centuries* (Second impression). Spring Books.
- Stanford Vision Lab. (2016). *ImageNet*. <http://www.image-net.org/>
- Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. (2017). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), Article 1. <https://doi.org/10.1609/aaai.v31i1.11231>
- Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-9. <https://doi.org/10.1109/CVPR.2015.7298594>
- Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *Proceedings of the 36th International Conference on Machine Learning (PMLR)*, 97, 6105-6114. <https://proceedings.mlr.press/v97/>
- TensorFlow Developers. (n.d.). Keras. Google. <https://www.tensorflow.org/guide/keras>
- Turconi, C. (1997). The map of Bedolina, Valcamonica Rock Art. *TRACCE Online Rock Art Bulletin*, 9. <http://www.rupestre.net/tracce/?p=2422>
- Uijlings, J. R. R., van de Sande, K. E. A., Gevers, T., & Smeulders, A. W. M. (2013). Selective Search for Object Recognition. *International Journal of Computer Vision*, 104(2), 154-171. <https://doi.org/10.1007/s11263-013-0620-5>
- Unger, R. W. (2010). *Ships on Maps: Pictures of Power in Renaissance Europe*. Palgrave Macmillan.
- Van Bleyswyck, D. (n.d.). *Kaart Figuratief* [Map]. Essential Vermeer 3.0. <http://www.essentialvermeer.com/maps/delft/kaart.html#.XNAEA1VMRaR>
- Virrantaus, K., Fairbairn, D., & Kraak, M.-J. (2009). ICA Research Agenda on Cartography and GIScience. *Cartography and Geographic Information Science*, 36(2), 209-222. <https://doi.org/10.1559/152304009788188772>
- Wallis, H., & Robinson, A. H. (Eds.). (1987). *Cartographical Innovations: An International Handbook of Mapping Terms to 1900* (New edition). Map Collector Publications Ltd.
- Wang, X., Xiao, T., Jiang, Y., Shao, S., Sun, J., & Shen, C. (2018). Repulsion Loss: Detecting Pedestrians in a Crowd. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7774-7783. <https://doi.org/10.1109/CVPR.2018.00811>
- Wang, Y., Lv, H., Chen, X., & Du, Q. (2015). A PSO-Neural Network-Based Feature Matching Approach in Data Integration. In C. R. Sluter, C. B. M. Cruz, & P. M. L. de Menezes (Eds.), *Cartography—Maps Connecting the World* (pp. 189-219). Springer International Publishing. https://doi.org/10.1007/978-3-319-17738-0_14
- Yanagisawa, H., Yamashita, T., & Watanabe, H. (2018). A study on object detection method from manga images using CNN. *2018 International Workshop on Advanced Image Technology (IWAIT)*, 1-4. <https://doi.org/10.1109/IWAIT.2018.8369633>
- Zhu, X. X., Tuia, D., Mou, L., Xia, G.-S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4), 8-36. <https://doi.org/10.1109/MGRS.2017.2762307>
- Ziran, Z., & Marinai, S. (2018). Object Detection in Floor Plan Images. In L. Pancioni, F. Schwenker, & E. Trentin (Eds.), *Artificial Neural Networks in Pattern Recognition* (Vol. 11081, pp. 383-394). Springer International Publishing. https://doi.org/10.1007/978-3-319-99978-4_30
- Zoph, B., & Le, Q. V. (2017). Neural Architecture Search with Reinforcement Learning. *ICLR 2017*. <https://iclr.cc/archive/www/doku.php%3Fid=iclr2017:schedule.html>

4. Instance Segmentation, Body Part Parsing, and Pose Estimation of Human Figures in Pictorial Maps

Raimund Schnürer¹, A. Cengiz Öztireli², Magnus Heitzler³, René Sieber⁴, Lorenz Hurni⁵

^{1,3,4,5} Institute of Cartography and Geoinformation, ETH Zurich, Zurich, Switzerland

² Department of Computer Science and Technology, University of Cambridge, Cambridge, United Kingdom

Peer-reviewed journal article

Published online: 10 August 2021

International Journal of Cartography

<https://doi.org/10.1080/23729333.2021.1949087>

Key findings

- Silhouettes of human figures on pictorial maps can be segmented individually by Convolutional Neural Networks at satisfying accuracy rates, though the delimitation to other pictorial objects remains challenging.
- Convolutional Neural Networks trained with a mixture of synthetic and realistic entities are more effective in segmenting instances of human figures on maps compared to entities which originate from a single level of representation.
- Body parts and pose points of human figures can be detected simultaneously by Convolutional Neural Networks at satisfying accuracy rates, though the handling of unusual poses remains challenging.
- Convolutional Neural Networks with fewer skip connections are more effective in detecting simultaneously body parts and pose points of human figures compared to networks with more skip connections.

Author contributions

Conceptualisation^{1,4,5}, Methodology¹, Software¹, Investigation¹, Data curation¹, Writing - Original draft¹, Writing - Review & Editing^{1,2,3,4,5}, Visualisation¹, Supervision^{2,4,5}, Project administration^{1,5}, Funding acquisition^{1,5}

Modifications to the original article

Typographic corrections, Citation updates of preprinted to published versions

Abstract

In recent years, convolutional neural networks (CNNs) have been applied successfully to recognise persons, their body parts and pose keypoints in photos and videos. The transfer of these techniques to artificially created images is rather unexplored, though challenging since these images are drawn in different styles, body proportions, and levels of abstraction. In this work, we study these problems on the basis of pictorial maps where we identify included human figures with two consecutive CNNs: We first segment individual figures with Mask R-CNN, and then parse their body parts and estimate their poses simultaneously with four different UNet++ versions. We train the CNNs with a mixture of real persons and synthetic figures and compare the results with manually annotated test datasets consisting of pictorial figures. By varying the training datasets and the CNN configurations, we were able to improve the original Mask R-CNN model and we achieved moderately satisfying results with the UNet++ versions. The extracted figures may be used for animation and storytelling and may be relevant for the analysis of historic and contemporary maps.

4.1. Introduction

Digital 'view-only' maps, such as scanned historical maps or modern maps made with graphic editors, include a multitude of information. To process this information in a machine-readable manner, the content of these maps needs to be extracted - in the best case fully automatically. At present, however, maps stored in a raster image format have been mostly manually annotated with metadata on social media websites and additionally georeferenced in digital map libraries, but information about their actual content is largely missing. In this work, we have a closer look at one particular content element of maps, namely human figures (Figure 4.1). This object type frequently occurs as a decoration in pictorial maps (Child, 1956). After successful detection, the following two use cases are thinkable: Firstly, historians (e.g. Davies, 2016) may be interested in the ethnos and clothing of figures, as well as in certain rites or common activities. Offering an additional search filter option in digital map catalogues for human figures in maps would therefore be highly beneficial. Secondly, figures could be animated to act as storytellers or guides in maps or paintings, for instance for museum visitors (e.g. D. S.-M. Liu et al., 2020). For the latter usage scenario, it is additionally required to identify their body parts and pose keypoints.

A promising technology to tackle the above-mentioned tasks are convolutional neural networks (CNNs). Recent experiments have shown that it is feasible to extract labels (Weinman et al., 2019), road intersection points (Saeedimoghaddam & Stepinski, 2020), or building footprints (Heitzler & Hurni, 2020) from maps with CNNs. By conducting our research, we like to extend this list by human figures. To our knowledge, this object class has not been retrieved with CNNs in maps yet, only in natural images such as photos. Here, architectures like *Mask R-CNN* (He et al., 2017) or *PANet* (S. Liu et al., 2018) were developed to segment individual objects. The task of parsing object parts, such as body parts, was approached with configurations like an adapted fully convolutional network (FCN) (Oliveira et al., 2016) or *DeepLab* (L. Chen et al., 2018). *Convolutional Pose Machines* (Wei et al., 2016) and *Stacked Hourglass* (Newell et al., 2016) enable to detect pose keypoints of single humans, whereas newer networks (e.g. Cao et al., 2017; J. Wang et al., 2021) are capable of registering multiple persons. Body part segmentation and keypoint detection were also combined, for example, in an FCN including a conditional random field (Xia et al., 2017) or *JPPNet* (Liang et al., 2019).

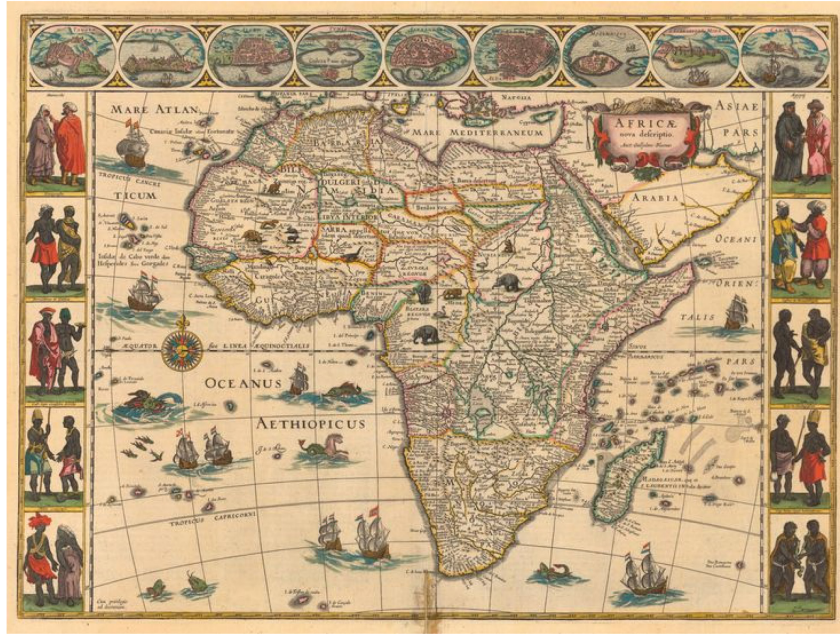


Figure 4.1: Exemplary pictorial map with human figures on the sides (source: Pinterest²⁴)

4.2. Data

Compared to real persons in photos, annotated datasets of human figures in maps are non-existing. Therefore, we created the data for training and testing our CNNs by ourselves. The data partly originates from a *Pictorial Map Classification Dataset*²⁵. This dataset consists of 3100 maps harvested from Pinterest accordingly to Art. 24d of the Swiss Copyright Act²⁶, among 1500 are pictorial maps. A manual classification by ourselves revealed that roughly half of the pictorial maps include human figures. We included humanoid representations (e.g. Statue of Liberty, Christ the Redeemer; snowmen, robots) but we excluded humanoid animals (e.g. King Kong) in our classification. Fifty-two of those maps, which include 387 larger figures, were manually annotated using a web application (Figure 4.2). The average size of maps is 577×581px and of figures is 44×78px. The low resolution of the map images to be fed into CNNs is due to memory limitations of current graphic boards. Annotating one figure took about nine minutes (incl. corrections), resulting in a total working time of about 60 h. Six body parts (head, torso, two arms, and two legs) were masked and skeletons consisting of 16 keypoints (head, neck, thorax, two shoulders, two elbows, two wrists, pelvis, two hip joints, two knees, and two ankles) were created. Since the number of annotated maps would be too little to train CNNs, these 52 maps will serve as testing data for the CNNs.

²⁴ <https://i.pinimg.com/736x/16/36/0b/16360bbad0f6ed13e56ff7215e7590dc.jpg>

²⁵ <http://narrat3d.ethz.ch/detection-of-pictorial-map-objects-with-cnns/>

²⁶ https://www.fedlex.admin.ch/eli/cc/1993/1798_1798_1798/en#art_24_d



Figure 4.2: Manually annotated body parts (left) and skeletons (right) of pictorial figures in the web application Supervisely on a touristic map (source: Pinterest²⁷). Images and annotations serve as test data.

For training the CNNs, we produced a synthetic dataset, which is a common approach in machine learning (e.g. Varol et al., 2017). It has the advantage that larger amounts of annotated data can be created in less time compared to manually annotating the data. However, the variability of data may not be as large (i.e. the styles of the figures in our case) and training times may need to be reduced so that the CNN does not specialise in the synthetic data. Our synthetic data consists of background maps, on which persons and objects from photos, generated figures, and icon objects are placed (Figure 4.3). By this mixture of real and abstract representations of humans, we hope that the CNN interpolates between them to recognise pictorial figures. Since not only human figures but also other objects (e.g. means of transportation, animals) are an integral part of pictorial maps, we included them as well so that the network learns to distinguish between figures and objects.

The background maps were also derived from the *Pictorial Map Classification Dataset* by selecting only those without any human figures. Thirty-three anthropomorphic maps (e.g. Europa Prima Pars Terrae In Forma Virginis, The Avenger - An Allegorical War Map) were excluded because their content is interwoven with humanoid figures. One map of them was added due to balancing reasons. The selection resulted in 2306 background maps, among which about one third are pictorial maps without any persons and two thirds are non-pictorial maps. To enrich these maps, 4558 real persons (Figure 4.4a), including skeletons and body part masks, and 5169 real objects (Figure 4.4b) from 19 different categories were firstly taken from the *PASCAL-Part Dataset*²⁸, which contains

²⁷ <https://i.pinimg.com/736x/19/3e/7c/193e7c6023c8ee543aae2e78c30e674d--travel-england-england-uk.jpg>

²⁸ <http://roozbehm.info/pascal-parts/pascal-parts.html>

annotations for photos. Secondly, 4558 synthetic human figures (Figure 4.4c) were generated in a custom web application, initially as scalable vector graphics (SVG) which are finally rasterised to images. The selection of real persons and the generation of synthetic figures are based on the frequency of occurrence of certain body part configurations in the test dataset (Table 4.1). The postures of the synthetic figures are derived from skeleton annotations of the *MPII Human Pose Dataset*²⁹. At the corresponding joints of each skeleton, ellipses were drawn for the head, whereas polygons - partly with rounded corners - were drawn for the torso, arms, and legs. Shapes for hats, hair, glasses, eyebrows, eyes, noses, mouths, hands, and shoes were additionally attached to the synthetic figures. Shapes, colours, fill patterns, body part sizes, and stroke widths were randomly varied. Pose keypoints, originating from the skeletons, and body part masks, derived from the overlays, were generated aside from the synthetic figure images. Thirdly, 4759 medium-sized, non-circular icon objects (Figure 4.4d) from 44 categories were retrieved from *Iconfinder*³⁰. In the last automated step, a random number of zero to 15 real and synthetic persons as well as objects are scaled randomly between 20 and 120px and placed randomly on the background maps. In case of overlaps, person masks covering an area of less than 50px, and corresponding keypoints, were excluded.



Figure 4.3: Real and synthetic entities randomly scaled and placed on a map (source: Pinterest³¹). Map images (source data, left) and silhouettes (target data, right) of figures serve as training data for instance segmentation.

²⁹ <http://human-pose.mpi-inf.mpg.de/>

³⁰ <https://www.iconfinder.com/>

³¹ <https://i.pinimg.com/736x/11/f1/eb/11f1ebb9bcf3d20690cb0f3d8fcf5119--city-maps-terra.jpg>



Figure 4.4: A real person, its body part mask and skeleton (a) as well as a real object from the PASCAL-Part dataset (b); a synthetic person, its body part mask and skeleton from our SVG figure generator (c) as well as an icon object from Iconfinder (d). Figures like (a) and (c) serve as training data for body part parsing and pose estimation.

We tested how the following training datasets, varying in real and synthetic entities (= persons and objects), affect the accuracy of the CNN targeted at instance segmentation:

- Real: 2304 maps with real entities
- Synthetic: 2304 maps with synthetic entities
- Separated: 1152 maps with real entities and 1152 maps with synthetic entities
- Mixed: 2304 maps with real and synthetic entities
- Separated-Mixed: 768 maps with real entities, 768 maps with synthetic entities, and 768 maps with real and synthetic entities

For training the CNN targeted at body part detection and pose estimation, 4558 real persons, 4558 synthetic figures, and a combination of 2279 real persons and 2279 synthetic figures (i.e. the Separated dataset) are taken into account. A larger selection of pictorial maps and figures, real persons and objects, synthetic figures and icon objects, and background and training maps can be found in Figure 4.9-Figure 4.16 in the Appendix.

Configuration	Frequency
Full body	49.61%
Both legs missing	13.18%
One arm missing	12.66%
One leg missing	4.65%
Both legs and both arms missing	5.43%
Single heads	4.13%
One leg and one arm missing	3.62%
Both legs and one arm missing	3.36%
Both arms missing	1.81%
Others	1.55%

Table 4.1: Frequency of occurrence of body part configurations in our test dataset, which consists of 387 pictorial figures

4.3. Methods

We follow a top-down approach (e.g. K. Lin et al., 2020) by first segmenting instances of human figures on maps and then body parts and pose keypoints. In the first step, we try to identify silhouettes of individual characters on pictorial maps with the established Mask R-CNN (He et al., 2017) architecture. This CNN is targeted at recognising objects, such as persons, from photos at a pixel level. Mask R-CNN is an extension of Faster R-CNN (Ren et al., 2017), a network, which is able to detect bounding boxes of objects. Similar to Faster R-CNN, a series of convolution and downscaling operations are initially applied to extract specific image features by a backbone network. The output, so-called feature maps, are processed next in two stages: Firstly, objectness scores, denoting the likelihood that a region contains an object, and offsets for anchors, which are rectangles differing in size and aspect ratio, and which are distributed equally in a grid covering the feature maps, are predicted in a region proposal network. Secondly, the regions of interest are further refined and a score for the potential object class is predicted. As an addition to Faster R-CNN, Mask R-CNN predicts a binary mask in this second stage, where each pixel of the mask corresponds to a probability. Only one channel is required for the mask since the separation of a potentially contained object from the background is determined by a threshold.

Mask R-CNN is included in the TensorFlow Model Garden³² where different CNN architectures are pre-implemented and accessible via a Python API. In our experiment, we retrained the model based on the COCO dataset (T.-Y. Lin et al., 2014), which comprises segmentations for 500,000 masked objects on photos, such as persons. For transfer learning, we take the five training datasets (i.e. Real, Synthetic, Separated, Mixed, Separated-Mixed) described in the previous chapter. As a backbone network for feature detection, we use ResNet with 101 layers (He et al., 2016) and atrous convolutions, which has a good accuracy-speed balance compared to the other three available TensorFlow models. Atrous (aka dilated) convolutions “enlarge the field of view of filters to incorporate larger context, which [has been] shown to be beneficial” (L. Chen et al., 2018, p. 4). We set the anchor stride to eight, which has been favourable to detect smaller objects (e.g. Schnürer et al., 2021). We vary the four sizes for the anchors (minimum: 0.0625, maximum: 2.0) and retain their three aspect ratios (i.e. 2:1, 1:1, and 1:2). Since the architecture requires much graphics memory, images could be fed in batches of one (i.e. single images) on an NVIDIA GeForce GTX 1080 Ti. On this graphics board, one epoch of learning (i.e. 2304 steps) takes about 30 minutes. Other parameters have not been modified from the given Mask R-CNN configuration file. For evaluation, we calculated the average precision (AP) for masks³³ with the COCO API. As given in the configuration for the original model, we set the threshold of confidence scores to larger than 0.3, which means that detections below the threshold will be discarded.

After having recognised human silhouettes, we parse body parts and detect keypoints simultaneously in our own network consisting of four different versions (Figure 4.5). Other networks have been proposed, that perform both tasks at the same time, but those do not have any code available (e.g. Xia et al., 2017) or classify different types of body parts (e.g. Liang et al., 2019). Therefore, we cannot report any baselines for pre-trained models on real persons. Mask R-CNN would be also able to indicate keypoints;

³² <https://github.com/tensorflow/models>

³³ <https://cocodataset.org/#detection-eval>

however, this functionality was not part of the code. Therefore, we decided to implement and test our own network configurations, inspired by a simple deconvolution head network (Xiao et al., 2018) and UNet++ (Z. Zhou et al., 2020). In contrast to the simple deconvolution head network (to be called ‘Simple Deconv’ in the following) but similarly to UNet++, we used a decreasing number of filters for the deconvolution operations. Diverging from UNet++ but similarly to Simple Deconv, we do not include the loss of intermediate layers and we do not perform any convolution operations after having upsampled the feature maps. Opposed to both networks but similar to other networks like Stacked Hourglass (Newell et al., 2016), we perform an ‘Add’ operation to merge layers instead of concatenating them and we do not upsample the image to the full resolution. More details of the architectural decisions and their alternatives are given in the Discussion chapter.

We implemented our body part parsing and pose estimation networks with TensorFlow’s high-level keras API³⁴. We use ResNet with 50 layers (He et al., 2016) pre-trained on ImageNet weights as a backbone network. We feed square RGB images encoded in the JPEG format with a size of 128²px into ResNet, which is smaller than the default image input size of 224²px, but the reduced size better conforms to our data. An additional one-strided convolution is performed to adjust the number of channels to 128 of the output layer after the first two-strided convolution in ResNet. This step is not necessary for the ResNet output feature maps of the second to the fourth stage. We omit the lowest stage of ResNet because figures on maps do not have as many details as persons in photos. The results of the backbone network are passed to a head network, where we test four different versions (Figure 4.5):

- Simple Deconv: The output feature map of the fourth ResNet stage ($X^{3,0}$) is upsampled three times ($X^{2,1}$, $X^{1,2}$, and $X^{0,3}$) by two-strided convolutions.
- Simple UNet: Supplementary to Simple Deconv, outputs of the third ($X^{2,0}$), second ($X^{1,0}$), and first ($X^{0,0}$) ResNet stage are added to the upsampled feature maps - one at a time.
- Simple UNet+: The output feature maps of the third and second ResNet stages are upsampled and added to the outputs of the second and first stage. The first result ($X^{1,1}$) from the previous addition is upsampled and added to the second result ($X^{0,1}$). The output of the third ResNet stage, the first and the third result ($X^{1,2}$) are added to the upsampled feature maps of Simple Deconv.
- Simple UNet++: Supplementary to Simple UNet+, the following skip connections are inserted: $X^{1,0}$ before $X^{1,2}$, $X^{0,0}$ before $X^{0,2}$, $X^{0,0}$ before $X^{0,3}$, and $X^{0,1}$ before $X^{0,3}$.

³⁴ https://www.tensorflow.org/api_docs/python/tf/keras

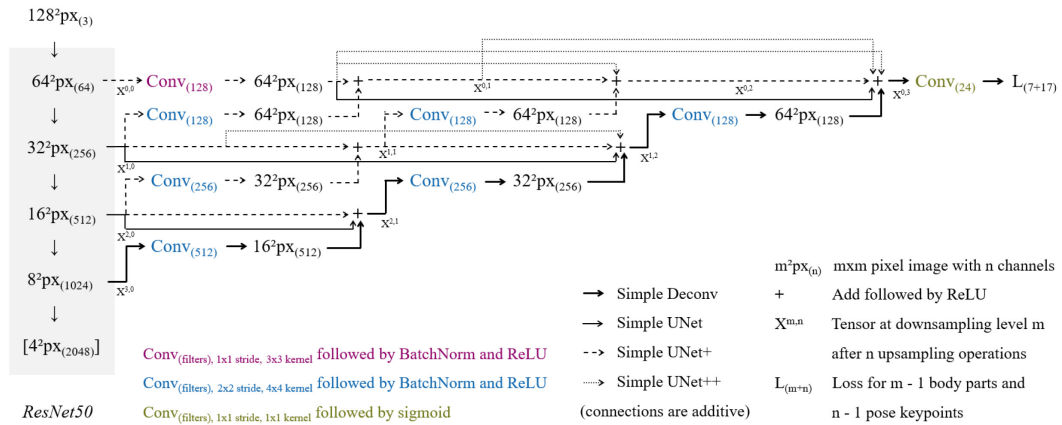


Figure 4.5: Our custom CNN architectures (see arrows) to parse body parts and estimate poses simultaneously

$X^{m,n}$ is the notation from UNet++: X symbolises the tensor, m corresponds to the downsampling level, and n denotes how many upsampling (i.e. deconvolution) operations have been performed. In our head networks, each deconvolution and adding operation is followed by a batch normalisation operation and a Rectified Linear Unit activation function. Kernels of convolutional layers are initialised by a truncated normal distribution centred at zero (i.e. 'he_normal'). The final convolution operation is followed by a sigmoid activation function so that we have a 64^2px image with 24 channels in the end. The channels correspond to six body parts and one channel for the background, and to 16 keypoints and one channel containing the inverted image of the summed keypoints. In the ground truth data, body part pixels have a value of one and other pixels have a value of zero. The keypoints are represented by a 2D Gaussian kernel, similar to Convolutional Pose Machines (Wei et al., 2016), having a probability of one in the centre and gradually decreasing values around the centre (Figure 4.6). The loss is split between body parts and keypoints, and reduced in both cases using the categorical cross entropy function and the RMSprop optimiser during training. The background channel for body parts is ignored in the loss function so that the network is not biased towards the white background of the images with human figures. We fed images in batches of 15 into the network and trained for 15 epochs, which took about 10 minutes in total.

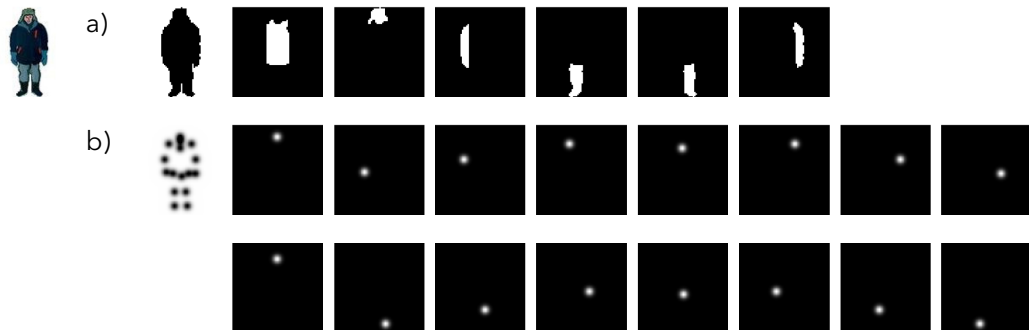


Figure 4.6: One-hot encoded masks for body parts (a) and keypoints (b) of a human figure from the test dataset. The first mask is the difference of the summed other masks. The other masks represent single body parts or pose keypoints.

4.4. Results

We trained each Mask R-CNN configuration (i.e. same training data and same anchor scales) five times, disregarded the highest and lowest score, and averaged the remaining three scores to reduce the variability in the comparison. Our evaluation procedure is similar to Zhang et al. (2019), who additionally calculated the standard deviation but did not discard the extreme scores. Quantitative results (Table 4.2) show that the highest AP on average was obtained when training Mask R-CNN with the Separated dataset, where real and synthetic entities are placed on different maps. The AP is higher than the original model trained with the COCO dataset. Qualitative results (Figure 4.7) illustrate that more human figures could be identified on maps with the retrained model. However, the number of false positives also increased as seen in the exemplary visual results. Yet, not all figures - especially smaller ones - could be identified.

Data	Anchor scales	AP	AP ₅₀	AP ₇₅	AP _{small}	AP _{medium}	AP _{large}
COCO	1/4, 1/2, 1/1, 2/1	15.57%	32.18%	14.08%	6.97%	21.54%	16.37%
Synthetic	1/4, 1/2, 1/1, 2/1	2.85%	7.67%	1.48%	0.54%	4.86%	0.00%
	1/8, 1/4, 1/2, 1/1	2.97%	8.55%	1.37%	0.78%	5.21%	2.23%
	1/16, 1/8, 1/4, 1/2	2.64%	6.83%	1.95%	0.65%	4.51%	0.00%
Real	1/4, 1/2, 1/1, 2/1	4.87%	12.04%	3.23%	2.12%	6.95%	8.42%
	1/8, 1/4, 1/2, 1/1	3.63%	9.12%	2.66%	1.81%	5.25%	2.97%
	1/16, 1/8, 1/4, 1/2	6.07%	15.49%	3.28%	2.10%	8.74%	9.90%
Separated	1/4, 1/2, 1/1, 2/1	19.11%	44.39%	11.70%	6.73%	27.23%	17.11%
	1/8, 1/4, 1/2, 1/1	18.18%	43.10%	9.56%	5.32%	26.61%	19.46%
	1/16, 1/8, 1/4, 1/2	17.54%	41.28%	10.61%	5.43%	25.69%	19.88%
Mixed	1/4, 1/2, 1/1, 2/1	10.40%	27.76%	4.90%	3.03%	15.93%	11.12%
	1/8, 1/4, 1/2, 1/1	9.73%	25.62%	4.45%	3.29%	14.47%	9.90%
	1/16, 1/8, 1/4, 1/2	11.07%	27.24%	5.82%	3.49%	16.73%	10.40%
Separated-Mixed	1/4, 1/2, 1/1, 2/1	14.29%	34.01%	7.53%	4.56%	20.79%	17.39%
	1/8, 1/4, 1/2, 1/1	13.32%	33.25%	6.30%	3.68%	19.86%	18.47%
	1/16, 1/8, 1/4, 1/2	14.21%	34.42%	8.04%	3.76%	21.49%	17.79%
<i>Best run: Separated</i>	1/4, 1/2, 1/1, 2/1	19.38%	46.43%	10.95%	6.26%	28.12%	18.71%

Table 4.2: Averaged COCO metrics of retrained Mask R-CNN models (backbone: ResNet101 with atrous convolutions) for different datasets and scales. The best and the worst result of five runs have been excluded from the calculation. The first row contains the baseline metrics for the original Mask R-CNN model trained on the COCO dataset. The highest average scores are marked in bold. The last row contains the highest overall achieved result for a retrained model.

APs were lower when training Mask R-CNN with synthetic or real entities only, likewise where those entities were mixed on maps. The Separated-Mixed dataset resulted in an AP ranging in the middle between the standalone datasets. The results vary slightly for different anchor scales; however, no clear trend could be observed whether smaller or larger values are favourable. For example, the best overall AP was achieved for the largest tested anchor scale values with the Separated dataset, whereas the highest AP on average was measured for the smallest anchor scale values when training with the Mixed dataset. For the other datasets, sometimes larger, intermediate, and smaller anchor scales led to higher accuracies on average.

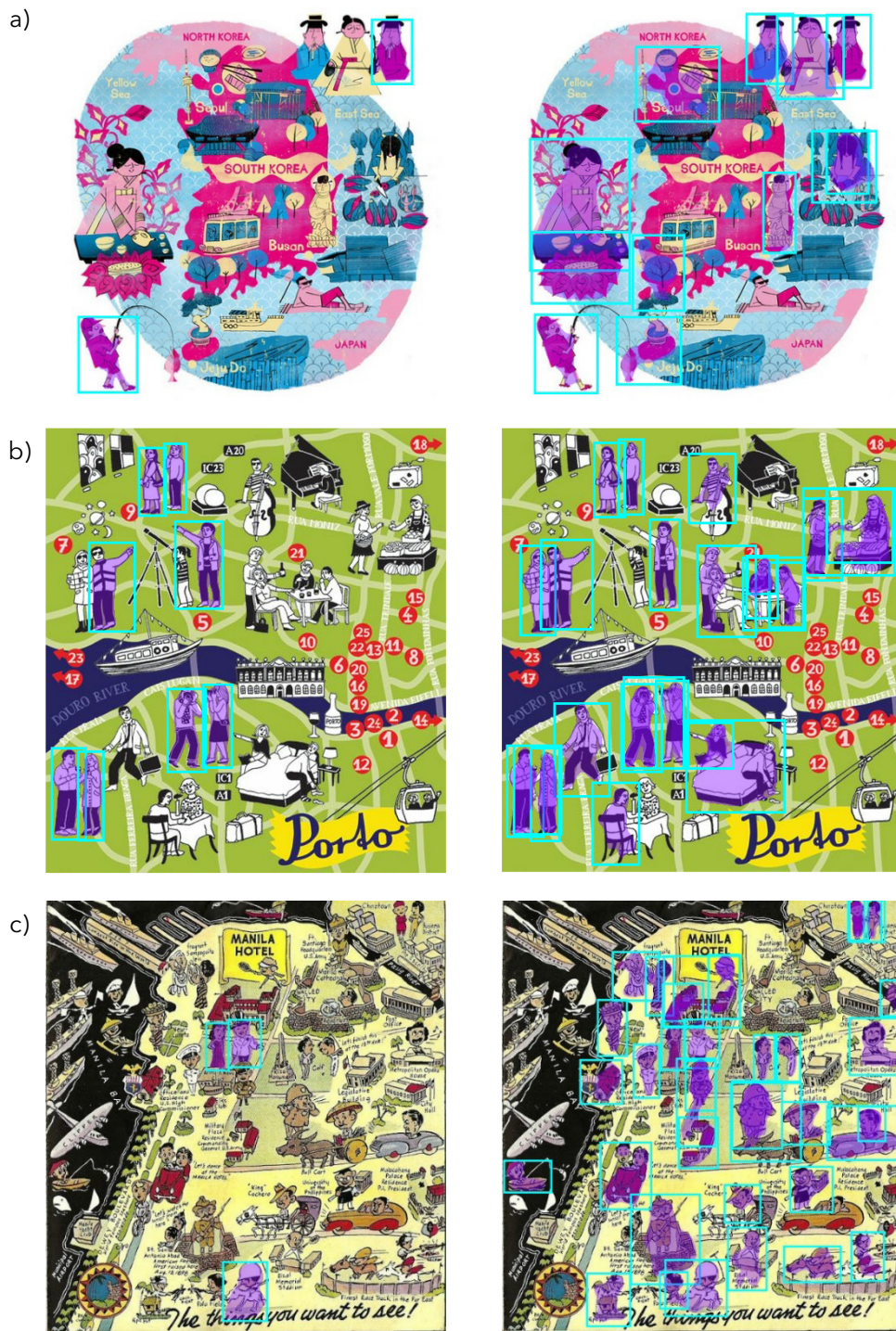


Figure 4.7: Comparison of Mask R-CNN results for few (a), several (b), and many (c) human figures on maps (sources: Pinterest^{35, 36, 37}), between the original COCO model (left) with an AP of 15.57% and the best retrained model (right) with an AP of 19.38%. In both cases, objects are highlighted with confidence scores > 0.3 .

³⁵ <https://i.pinimg.com/736x/1a/dc/c2/1adcc2066b649919f3d28afb4f22985b.jpg>

³⁶ <https://i.pinimg.com/736x/00/7a/a0/007aa06e4b34aac357df25b239781ce3.jpg>

³⁷ <https://i.pinimg.com/736x/14/d1/12/14d11277df1e2f9e66fc0d6723106c4e-illustrated-maps-manila.jpg>

As training times for our CNNs on body parsing and pose estimation were lower, we run each configuration (i.e. same dataset, same architecture) 20 times, disregarded the four highest and lowest scores, and averaged the remaining 12 scores. Quantitative results (Table 4.3) indicate that the highest AP on average for both tasks and the best overall accuracy was achieved for Simple UNet trained with real and synthetic persons. Simple UNet+ and Simple UNet++, which are architectures with more connections, performed slightly worse, whereas the results of Simple Deconv are comparable to Simple UNet. Training the CNNs with real data only yielded similar results for parsing body parts, whereas the addition of synthetic data led to higher average accuracies for detecting pose keypoints. Training with synthetic figures only resulted in clearly lower accuracies.

Qualitative results (Figure 4.8) demonstrate that common poses and some of the more difficult ones (e.g. side view, overlapping or missing body parts) could be identified satisfactorily; however, sometimes more challenging cases (e.g. persons viewed from behind, similar body parts) caused classification errors. Unusual poses (e.g. during sports activities) or too small figures were not recognised very well.

Data	Architecture	AP	AP ₅₀	AP ₇₅	AP _{small}	AP _{medium}	AP _{large}	
Synthetic	Simple Deconv	2.27%	5.94%	1.46%	2.38%	2.77%	6.59%	
		0.36%	2.03%	0.04%	0.36%	0.12%	2.89%	
	Simple UNet	1.94%	5.14%	1.30%	2.14%	2.01%	8.02%	
		0.41%	2.28%	0.05%	0.38%	1.22%	3.00%	
	Simple UNet+	1.99%	5.37%	1.24%	2.25%	2.11%	10.91%	
		0.46%	2.59%	0.05%	0.43%	0.45%	3.16%	
	Simple UNet++	1.66%	4.64%	0.95%	1.92%	2.04%	8.67%	
		0.22%	1.27%	0.02%	0.17%	0.48%	1.89%	
	Real	Simple Deconv	10.03%	19.23%	9.18%	10.65%	7.27%	3.29%
			6.38%	21.49%	2.45%	8.98%	4.48%	18.10%
		Simple UNet	10.25%	19.56%	9.38%	11.05%	6.83%	8.82%
			7.77%	25.33%	3.30%	10.92%	5.70%	20.69%
Simple UNet+		9.61%	18.53%	8.76%	10.48%	5.91%	3.63%	
		7.78%	25.04%	3.21%	10.96%	4.35%	20.68%	
Simple UNet++		8.70%	16.90%	7.83%	9.68%	5.10%	2.90%	
		6.05%	20.68%	1.90%	8.50%	4.27%	17.42%	
Separated		Simple Deconv	10.80%	21.21%	9.69%	11.17%	8.77%	27.12%
			8.91%	30.14%	3.24%	11.64%	5.05%	21.94%
		Simple UNet	10.76%	20.52%	9.80%	11.36%	7.74%	29.62%
			10.13%	33.89%	3.53%	13.49%	3.35%	23.74%
	Simple UNet+	10.33%	19.99%	9.25%	10.85%	7.76%	26.28%	
		9.81%	32.23%	3.35%	13.33%	3.70%	23.33%	
	Simple UNet++	9.84%	19.21%	8.85%	10.41%	7.39%	23.39%	
		8.98%	29.76%	3.07%	12.02%	4.19%	21.98%	
	<i>Best run: Separated</i>	<i>Simple UNet</i>	12.46%	23.78%	11.23%	13.18%	7.96%	43.49%
			13.12%	42.75%	4.25%	16.95%	3.55%	27.60%

Table 4.3: Averaged COCO metrics for body parts (first row) and pose keypoints (second row) for different datasets and architectures. The four best and worst results of twenty runs have been removed from the calculation. The highest average scores for body parts and pose keypoints are marked in bold. The last row contains the highest achieved accuracy of all runs.

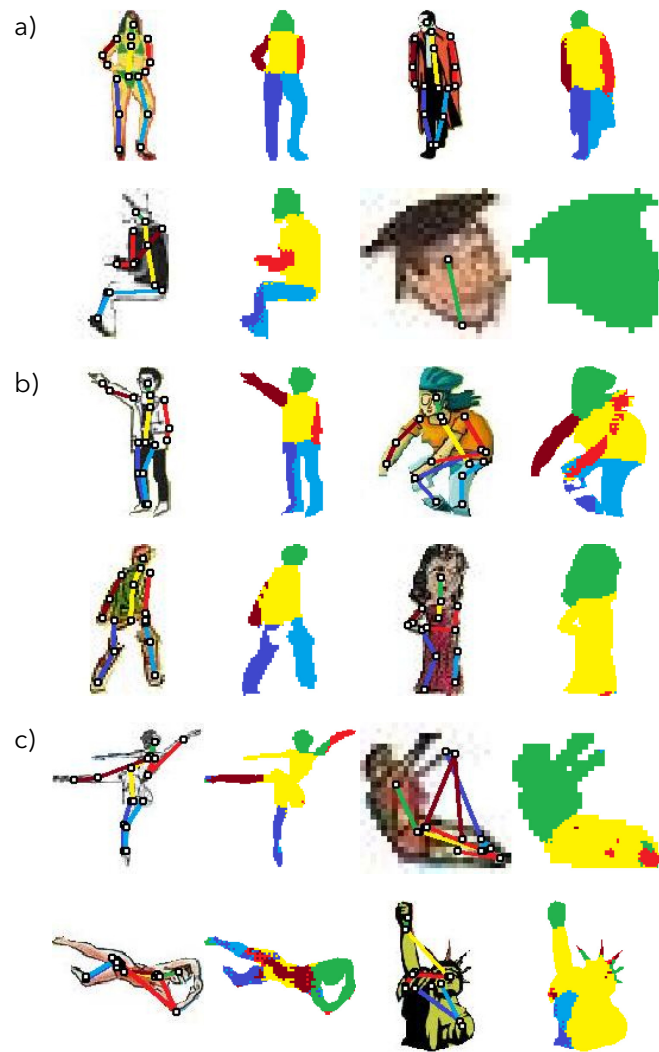


Figure 4.8: Selection of success cases (a), moderate failure cases (b), and severe failure cases (c) for simultaneous body parsing and pose estimation for the best retrained model on our Simple UNet

4.5. Discussion

While there were several training datasets available for human parsing in photos, no datasets existed for pictorial maps to our knowledge. As supervised learning with CNNs requires many training samples though, we decided to create a mixture of real persons extracted from photos and automatically generated abstract figures. Training a CNN with data from a different domain is usually less promising; however, it is not clear at this point whether a fully manually annotated dataset in our target domain would have led to better results due to the different drawing styles. In our case, the combination of real and synthetic data paid off for segmenting instances of human figures, their body parts, and estimating their poses. However, training the CNNs with synthetic data only would be not sufficient as the result metrics show. The accuracy may be higher when more variations of shapes, body features, and clothes of synthetic figures would be included. We tried to minimise the manual effort, for example, by generating a fill pattern with random polylines, but still, it is quite different from the original folds and shadows of clothes and hair. Originally, 10 body parts (incl. lower/upper arms/legs) were classified in our training datasets, but as CNNs have already struggled to segment 6 body parts in moderately complicated poses, we trained them with this number of categories. Therefore, a post-processing step (e.g. Voronoi diagram) would be required to distinguish the upper and lower parts of the limbs.

The achieved scores between 10% and 20% sound low, but qualitative results look already reasonable. The original Mask R-CNN model had an AP of 33%³⁸ for instance segmentation of different object categories on photos, but it is not clear where the score for persons would range. Our low scores for body parts and pose keypoints may be justified by the small image sizes so that already minor deviations have a large impact. Furthermore, the COCO metrics for keypoints³⁹ are not directly comparable to those of real images since they contain pre-defined standard deviations for every pose keypoint. Body proportions of pictorial figures, however, could be largely distorted. Internally, we have calculated a simpler error metric but as these results correlate with the COCO scores, we reported only the standard metric. For a comparison of the different CNN configurations, however, we relied on average results instead of giving only the best result. This methodology may be more robust concerning outliers since a configuration can be tested only a couple of times due to the long training times. When training more often, slightly higher APs than the reported ones can be achieved.

The developed CNN configurations were already the result of many trials, but a detailed evaluation of every architectural decision would go beyond the scope of this article. Instead, we like to give a brief overview of the different factors to consider: For both CNNs, it would be possible to use a different backbone network, image input sizes, learning rates, or number of learning steps and epochs. Besides those factors and the compared anchor scales, we relied on the default hyperparameters for Mask R-CNN because they already have been fine-tuned by the authors. For our CNN versions on body part parsing and pose estimation, we listed the tested possibilities and our decisions in Table 4.4. The alternatives led to worse or only marginally different outcomes compared to our parameters. We reported the scores of Simple UNet+ and

³⁸ https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf1_detection_zoo.md

³⁹ <https://cocodataset.org/#keypoints-eval>

Simple UNet++ to demonstrate that increasing the architectural complexity did not help to solve our problem. However, our results suggest that we have not found an optimal solution yet, only that we reached a local maximum. Further investigations, as proposed in the next chapter, will be needed to improve the quality of the outcomes.

	Possibility	Our decision
Training data	Different ratios of body part configurations	Proportionally to test data
	Different standard deviation of 2D Gaussian for representing keypoints	2
	Ratio between image input and output size	1:0.5
	Image format	JPEG
	Data augmentation (e.g. mirroring, noise)	No augmentation
	Initialisation	Using no pre-trained model for ResNet50 (i.e. learning from scratch) Different kernel initialiser of convolutional layers
Architecture: Encoder	Different conversion of the first stage of ResNet50	One additional convolutional layer
	Using all stages of ResNet50	Excluding the last stage of ResNet50
Architecture: Decoder	Different upsampling method (e.g. bilinear interpolation)	Two-strided convolution
	Different filter numbers (e.g. fixed) and kernel sizes (e.g. 3x3) for two-strided convolutions	Filters correspond to input channels, 4x4 kernel
	Additional convolutional layers after upsampling	No convolutional layers after upsampling
	Different layer combination operation (e.g. Concatenate)	'Add' operation
Training	Freezing parameters of ResNet50	Parameters of ResNet50 are adjusted during training
	Different loss function (e.g. mean squared error)	Categorical cross entropy
	Combined loss function for body parts and keypoints	Separate loss functions for body parts and keypoints
	Include intermediate layers in the loss function	Consider only the final layer for the loss function
	Different optimisers (e.g. Adam)	RMSprop

Table 4.4: Possibilities to consider and decisions made for our CNN versions on parsing body parts and estimating poses

4.6. Conclusion and future work

We have made the following contributions within the scope of this article:

- Creation of publicly available training datasets including annotated body parts and skeletons of real persons and synthetic figures on maps and single images
- Creation of publicly available test datasets including annotated body parts and pose keypoints of human figures on pictorial maps and single images
- Application of a CNN for instance segmentation of human figures on pictorial maps
- Development of CNN architectures for simultaneous prediction of body parts and pose keypoints of human figures on single images
- Qualitative and quantitative evaluation of the results of both CNNs

We measured an increased accuracy compared to the baseline model for identifying silhouettes of human figures on pictorial maps when training the CNN with real persons and synthetic figures on separate maps. The accuracy of CNNs detecting body parts and joints of human figures simultaneously was slightly higher for the simpler architectures. Here, the combination of real and synthetic data only led to a small gain in accuracy. Qualitative results showed that many figures on maps can be found, but not all. Body parts and keypoints were satisfactorily recognised for common poses by our developed CNN architectures, however, not for special cases.

Our work offers various potentials for improvement. Eventually, other datasets than *PASCAL-Part* with real persons could be added; unfortunately, those of *DensePose* (Güler et al., 2018) and *Look Into Person* (Liang et al., 2019) were only partly compatible. The synthetic figure generator could be extended so that a larger range of appearances and clothes is supported. Adding a generative adversarial network (GAN) may help to reduce the domain gap between training and test data (e.g. Sankaranarayanan et al., 2018). The number of annotated maps in the test dataset could be also increased as well as a distinction could be made between visible and hidden keypoints. Ideally, only one CNN is needed to fulfil all three tasks (i.e. instance segmentation, body part parsing, and pose estimation). Newer feature extractors than *ResNet*, for instance, *HR-Net* (J. Wang et al., 2021), may lead to a higher accuracy. Body part predictions may be smoothed by the addition of conditional random fields (Arnab et al., 2018).

The results of our work may be transferable to recognise human figures in books, comics/manga, or paintings. Extracting additional properties of figures (e.g. age, gender, skin colour, pieces of clothing, performed activity, and spatial relation to other figures) would be helpful for an even fine-grained search. Beyond, it would be interesting to identify other pictorial entities like animals or means of transportation. When figures were displaced from their original position, for example when being animated, an empty background would remain. This is a semantic image inpainting task that could be tackled by GANs (e.g. Yeh et al., 2017).

Appendix



Figure 4.9: Selection of pictorial maps from Pinterest. Those serve as our test data for instance segmentation.

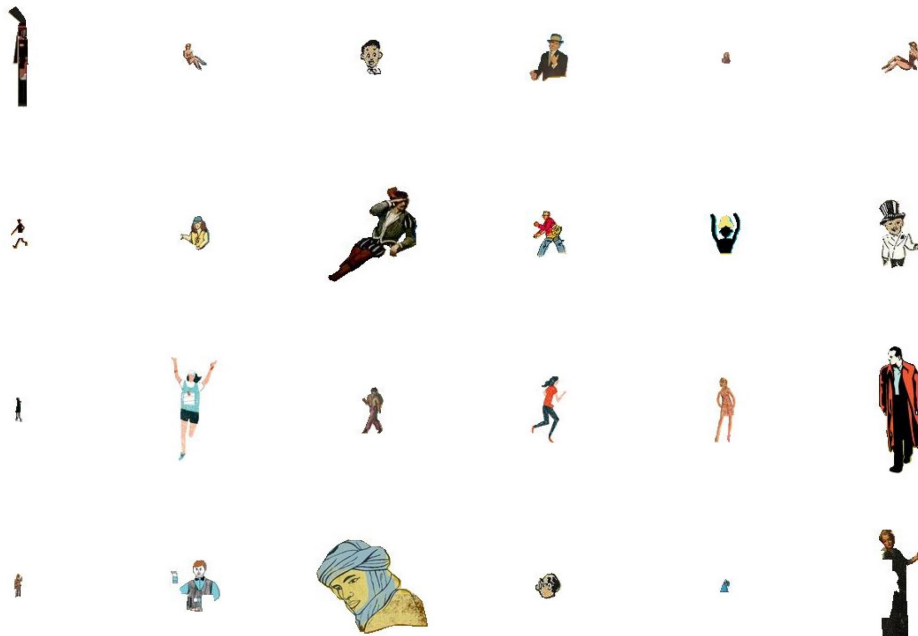


Figure 4.10: Selection of human figures extracted from pictorial maps from Pinterest. Those serve as our test data for body part parsing and pose estimation.



Figure 4.11: Selection of real persons extracted from photos from the PASCAL-part dataset. Those are a part of our training data for instance segmentation, body part parsing, and pose estimation.

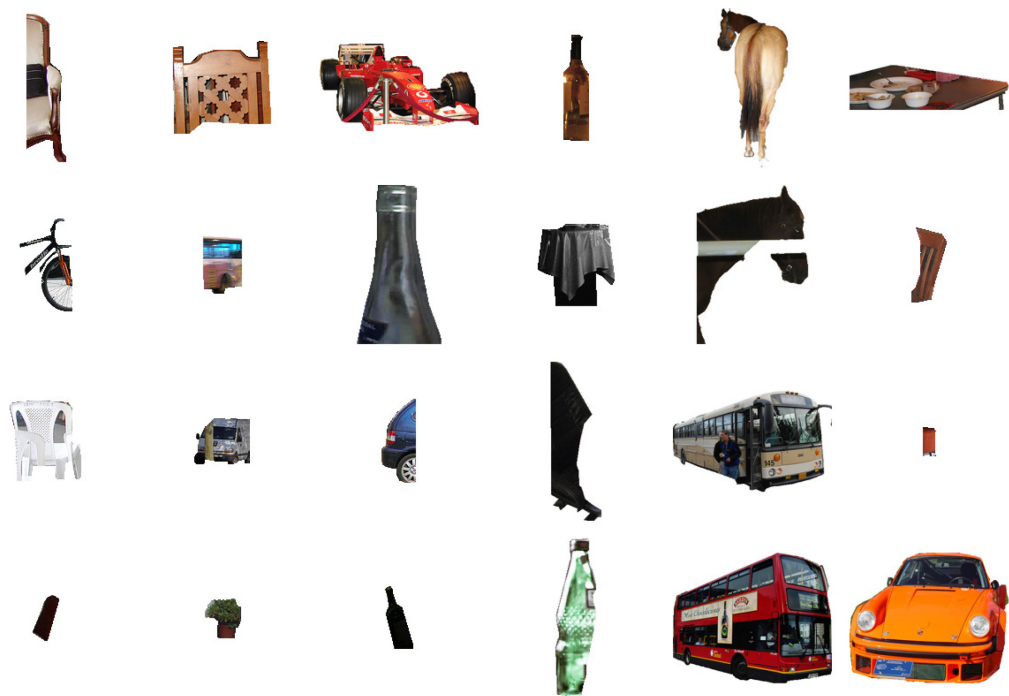


Figure 4.12: Selection of real objects extracted from photos from the PASCAL-part dataset. Those are a part of our training data for instance segmentation.



Figure 4.13: Selection of synthetically generated human figures by our custom web application with the MPII Human Pose dataset. Those are a part of our training data for instance segmentation, body part parsing, and pose estimation.

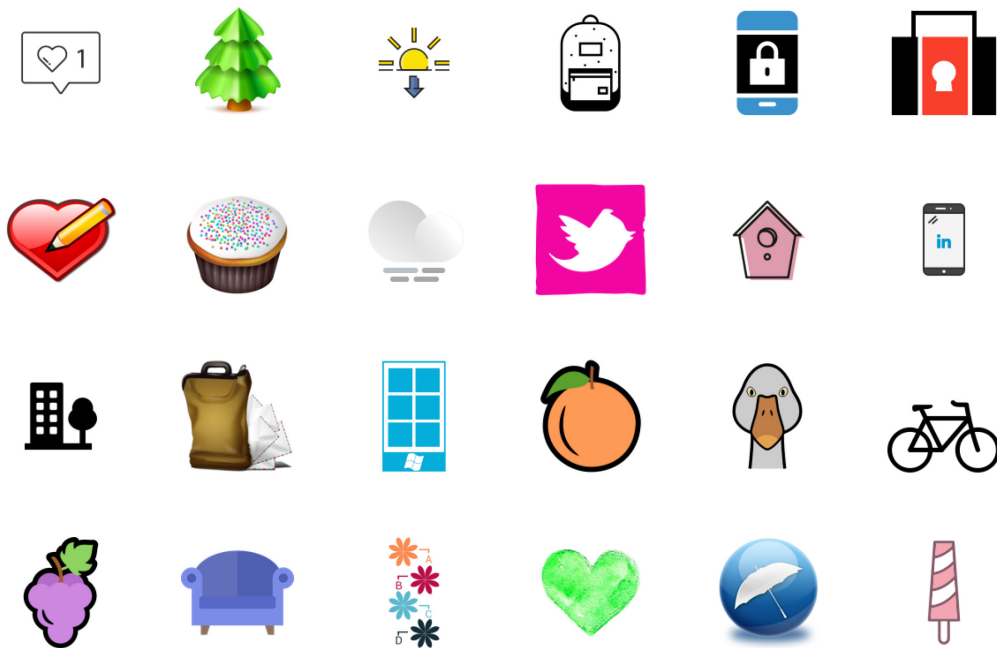


Figure 4.14: Selection of icon objects from Iconfinder. Those are a part of our training data for instance segmentation.

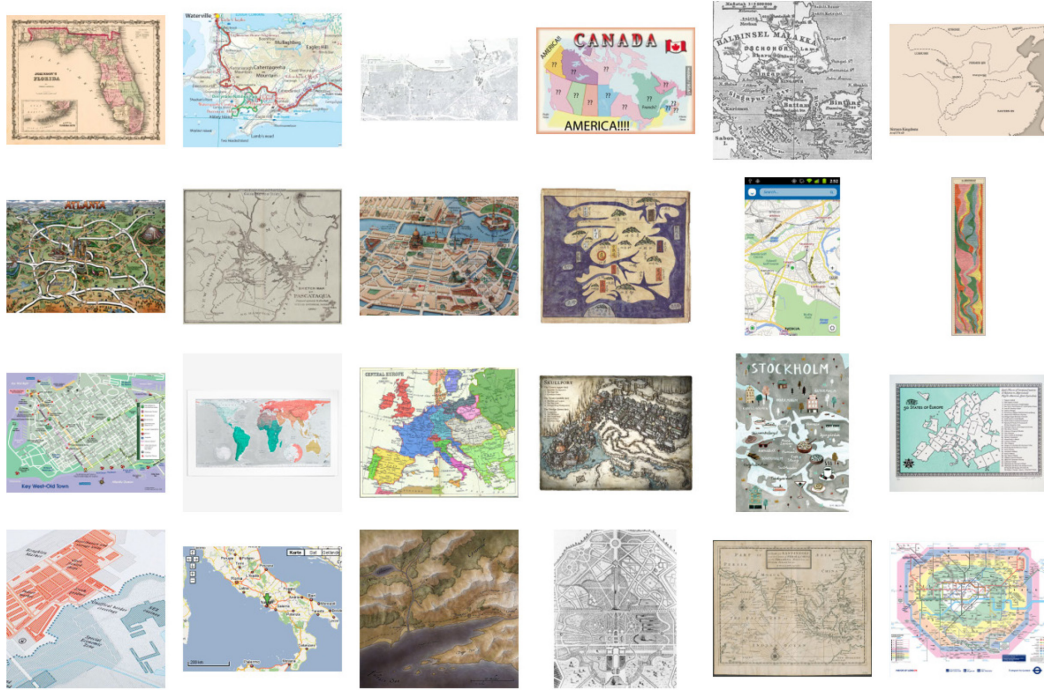


Figure 4.15: Selection of maps without any human figures from Pinterest. Those are a part of our training data for instance segmentation.



Figure 4.16: Selection of maps without any human figures, enriched with real or synthetic entities. Those are a part of our training data for instance segmentation.

References

- Arnab, A., Zheng, S., Jayasumana, S., Romera-Paredes, B., Larsson, M., Kirillov, A., Savchynskyy, B., Rother, C., Kahl, F., & Torr, P. H. S. (2018). Conditional Random Fields Meet Deep Neural Networks for Semantic Segmentation: Combining Probabilistic Graphical Models with Deep Learning for Structured Prediction. *IEEE Signal Processing Magazine*, 35(1), 37-52. <https://doi.org/10.1109/MSP.2017.2762355>
- Cao, Z., Simon, T., Wei, S., & Sheikh, Y. (2017). Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1302-1310. <https://doi.org/10.1109/CVPR.2017.143>
- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4), 834-848. <https://doi.org/10.1109/TPAMI.2017.2699184>
- Child, H. (1956). *Decorative Maps* (1st edition). Studio Publications.
- Davies, S. (2016). *Renaissance Ethnography and the Invention of the Human: New Worlds, Maps and Monsters*. Cambridge University Press. <https://doi.org/10.1017/CBO9781139568128>
- Güler, R. A., Neverova, N., & Kokkinos, I. (2018). DensePose: Dense Human Pose Estimation in the Wild. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7297-7306. <https://doi.org/10.1109/CVPR.2018.00762>
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2980-2988. <https://doi.org/10.1109/ICCV.2017.322>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778. <https://doi.org/10.1109/CVPR.2016.90>
- Heitzler, M., & Hurni, L. (2020). Cartographic reconstruction of building footprints from historical maps: A study on the Swiss Siegfried map. *Transactions in GIS*, 24(2), 442-461. <https://doi.org/10.1111/tgis.12610>
- Liang, X., Gong, K., Shen, X., & Lin, L. (2019). Look into Person: Joint Body Parsing Pose Estimation Network and a New Benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(4), 871-885. <https://doi.org/10.1109/TPAMI.2018.2820063>
- Lin, K., Wang, L., Luo, K., Chen, Y., Liu, Z., & Sun, M. (2020). Cross-Domain Complementary Learning Using Pose for Multi-Person Part Segmentation. 31(3), 1066-1078. <https://doi.org/10.1109/TCSVT.2020.2995122>
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer Vision - ECCV 2014* (pp. 740-755). Springer International Publishing. https://doi.org/10.1007/978-3-319-10602-1_48
- Liu, D. S.-M., Cheng, C.-I., & Liu, M.-L. (2020). Animating characters in Chinese painting using two-dimensional skeleton-based deformation. *Multimedia Tools and Applications*, 79(27), 20343-20371. <https://doi.org/10.1007/s11042-020-08842-5>
- Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path Aggregation Network for Instance Segmentation. 8759-8768. <https://doi.org/10.1109/CVPR.2018.00913>
- Newell, A., Yang, K., & Deng, J. (2016). Stacked Hourglass Networks for Human Pose Estimation. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer Vision - ECCV 2016* (pp. 483-499). Springer International Publishing. https://doi.org/10.1007/978-3-319-46484-8_29
- Oliveira, G. L., Valada, A., Bollen, C., Burgard, W., & Brox, T. (2016). Deep learning for human part discovery in images. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 1634-1641. <https://doi.org/10.1109/ICRA.2016.7487304>

- Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
- Saeedimoghaddam, M., & Stepinski, T. F. (2020). Automatic extraction of road intersection points from USGS historical map series using deep convolutional neural networks. *International Journal of Geographical Information Science*, 34(5), 947–968. <https://doi.org/10.1080/13658816.2019.1696968>
- Sankaranarayanan, S., Balaji, Y., Jain, A., Lim, S. N., & Chellappa, R. (2018). Learning from Synthetic Data: Addressing Domain Shift for Semantic Segmentation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3752–3761. <https://doi.org/10.1109/CVPR.2018.00395>
- Schnürer, R., Sieber, R., Schmid-Lanter, J., Öztireli, A. C., & Hurni, L. (2021). Detection of Pictorial Map Objects with Convolutional Neural Networks. *The Cartographic Journal*, 58(1), 50–68. <https://doi.org/10.1080/00087041.2020.1738112>
- Varol, G., Romero, J., Martin, X., Mahmood, N., Black, M. J., Laptev, I., & Schmid, C. (2017). Learning from Synthetic Humans. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4627–4635. <https://doi.org/10.1109/CVPR.2017.492>
- Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., & Xiao, B. (2021). Deep High-Resolution Representation Learning for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10), 3349–3364. <https://doi.org/10.1109/TPAMI.2020.2983686>
- Wei, S., Ramakrishna, V., Kanade, T., & Sheikh, Y. (2016). Convolutional Pose Machines. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4724–4732. <https://doi.org/10.1109/CVPR.2016.511>
- Weinman, J., Chen, Z., Gafford, B., Gifford, N., Lamsal, A., & Niehus-Staab, L. (2019). Deep Neural Networks for Text Detection and Recognition in Historical Maps. *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 902–909. <https://doi.org/10.1109/ICDAR.2019.00149>
- Xia, F., Wang, P., Chen, X., & Yuille, A. L. (2017). Joint Multi-person Pose Estimation and Semantic Part Segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6080–6089. <https://doi.org/10.1109/CVPR.2017.644>
- Xiao, B., Wu, H., & Wei, Y. (2018). Simple Baselines for Human Pose Estimation and Tracking. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), *Computer Vision - ECCV 2018* (pp. 472–487). Springer International Publishing. https://doi.org/10.1007/978-3-030-01231-1_29
- Yeh, R. A., Chen, C., Lim, T. Y., Schwing, A. G., Hasegawa-Johnson, M., & Do, M. N. (2017). Semantic Image Inpainting with Deep Generative Models. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6882–6890. <https://doi.org/10.1109/CVPR.2017.728>
- Zhang, J., Malmberg, F., & Sclaroff, S. (2019). Salient Object Subitizing. In J. Zhang, F. Malmberg, & S. Sclaroff (Eds.), *Visual Saliency: From Pixel-Level to Object-Level Analysis* (pp. 65–93). Springer International Publishing. https://doi.org/10.1007/978-3-030-04831-0_5
- Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2020). UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation. *IEEE Transactions on Medical Imaging*, 39(6), 1856–1867. <https://doi.org/10.1109/TMI.2019.2959609>

5. Inferring Implicit 3D Representations from Human Figures on Pictorial Maps

Raimund Schnürer¹, A. Cengiz Öztireli², Magnus Heitzler³, René Sieber⁴, Lorenz Hurni⁵

^{1,3,4,5} Institute of Cartography and Geoinformation, ETH Zurich, Zurich, Switzerland

² Department of Computer Science and Technology, University of Cambridge, Cambridge, United Kingdom

Peer-reviewed journal article

Published online: 26. June 2023

Cartography and Geographic Information Science

<https://doi.org/10.1080/15230406.2023.2224063>

Key findings

- Depth coordinates for 2D skeletons of human figures from pictorial maps can be estimated at very good accuracy rates by an Artificial Neural Network.
- 3D implicit surfaces for body parts of human figures can be inferred from binary masks and additional pose points at good accuracy rates by a Convolutional Neural Network, though the preservation of details remains challenging.
- UV coordinates of human figures can be predicted from depth images and body part masks at good accuracy rates by a Convolutional Neural Network, though the handling of imprecisely inferred body parts remains challenging.
- Textures of human figures can be inpainted from UV coordinates of a given image at satisfying accuracy rates by a Convolutional Neural Network, though the preservation of pictorial details remains challenging.
- Facial details of human figures can be enhanced at sufficient accuracy rates by a Convolutional Neural Network, though the generation of plausible results remains challenging.
- The assembled 3D models of human figures from the previous stages can be rendered in real-time using a ray tracing algorithm.

Author contributions

Conceptualisation^{1,4,5}, Methodology¹, Software¹, Investigation¹, Data curation¹, Writing - Original draft¹, Writing - Review & Editing^{1,2,3,4,5}, Visualisation¹, Supervision^{2,4,5}, Project administration^{1,5}, Funding acquisition^{2,5}

Modifications to the original article

Conversion of American English to British English, Typographic corrections

Abstract

In this work, we present an automated workflow to bring human figures, one of the most frequently appearing entities on pictorial maps, to the third dimension. Our workflow is based on training data and neural networks for single-view 3D reconstruction of real humans from photos. We first let a network consisting of fully connected layers estimate the depth coordinate of 2D pose points. The gained 3D pose points are inputted together with 2D masks of body parts into a deep implicit surface network to infer 3D signed distance fields (SDFs). By assembling all body parts, we derive 2D depth images and body part masks of the whole figure for different views, which are fed into a fully convolutional network to predict UV images. These UV images and the texture for the given perspective are inserted into a generative network to inpaint the textures for the other views. The textures are enhanced by a cartoonisation network and facial details are resynthesised by an autoencoder. Finally, the generated textures are assigned to the inferred body parts in a ray marcher. We test our workflow with 12 pictorial human figures after having validated several network configurations. The created 3D models look generally promising, especially when considering the challenges of silhouette-based 3D recovery and real-time rendering of the implicit SDFs. Further improvement is needed to reduce gaps between the body parts and to add pictorial details to the textures. Overall, the constructed figures may be used for animation and storytelling in digital 3D maps.

5.1. Introduction

Technology companies – such as Meta, Microsoft or Sony – invest heavily in the creation of the metaverse these days (Gilbert, 2021). In the visions of these companies (e.g. Meta, 2021), people equipped with head-mounted displays can immerse in virtual 3D environments for work or leisure activities. The virtual space may represent our physical world, be purely imaginary or be a mixture of both. Creating digital 3D representations of topographic elements and thematic content from the real world by abstraction and generalisation would be of interest from a cartographic perspective. Early works have focused on the rendering of sketched 3D buildings (Döllner & Walther, 2003) or the modelling of pictorial 3D mountains and sights (Naz, 2005). This is opposed to ‘mirror worlds’, for instance in Google Earth, where the real world is convincingly reflected (Park & Kim, 2022) and photo-realistically rendered. Cartographic 3D models and mirror worlds are closely related to ‘digital twins’, which are virtual representations of real-world entities for mainly simulation purposes (Park & Kim, 2022), for example, historical reconstructions (Herold & Hecht, 2018) or urban planning (Schrotter & Hürzeler, 2020).

Avatars are one key concept of the metaverse (Park & Kim, 2022). Those virtual 3D models of humans, animals, or other personifications embody real humans or computer-controlled entities, who can be interacted with. In a cartographic 3D environment, avatars may give background information to a topic, tell personal stories, or serve as tour guides. For example, 3D figures could illustrate daily life (e.g. a farmer on a field), act in special events (e.g. a priest at a coronation ceremony), or represent famous persons (e.g. Goethe in Weimar). Past research has examined the animation of 3D objects, such as cars and horses, on the terrain (Evangelidis et al., 2018) and the integration of 3D characters into cartographic virtual reality environments (Matthys et al., 2021).

The enrichment of cartographic digital twins with human figures would be an analogy to historic maps, where human figures have been inserted for ethnographic or humoristic purposes, amongst others (Child, 1956). Historic, but also contemporary pictorial maps would be valuable sources for creating 3D models of the depicted figures. Similarly to cartoons (X. Wang & Yu, 2020), pictorial figures on maps are usually composed of rather geometrically formed and possibly disproportionate shapes, whose low-detail textures are filled with flat colours and accentuated by sharp black edges. Nevertheless, the manual creation of 3D figures would be labour-intensive and cumbersome, and parametric models may not be detailed enough. Machine learning methods are a promising technique to reconstruct 3D models from humans and objects in photos (Fahim et al., 2021). In cartography, researchers rather focused on the detection of topographic elements on maps – such as buildings (Heitzler & Hurni, 2020) or water bodies (S. Wu, Heitzler, et al., 2022) – or pictorial figures (Schnürer et al., 2022) by convolutional neural networks (CNNs) so far, however not with their transfer into the 3D space yet.

In this work, we like to close this gap by applying a series of neural networks to infer 3D representations, encoded as signed distance fields (SDFs), from 2D figures on pictorial maps. Each point in an SDF holds a value denoting the distance to the nearest boundary of an object. As a difference to previous works, we do not recover the SDFs from textures but merely from silhouettes. Compared to meshes, point clouds, or voxels, implicit representations like SDFs have some desirable properties such as infinite geometric detail or easy blending capabilities. We use sphere tracing (Hart, 1996) to render the

figures in real-time, whereas other researchers mainly used marching cubes (Lorensen & Cline, 1987) to polygonise the SDFs. The sphere tracing algorithm is relatively well-established in contrast to newer methods like neural rendering (e.g. Eslami et al., 2018; Lassner & Zollhöfer, 2021). Our work has great potential for skeletal animation since we construct the figures by compositing 3D body parts according to 3D pose points. We see atlases, education, museums, tourism, or games in- and outside the metaverse as primary application areas.

5.2. Related work

In recent years, many advances have been made in reconstructing 3D persons and objects from single images using machine learning methods. For example, Omran et al. (2018) apply CNNs to predict parameters of the pose and shape of a 3D person model by taking advantage of segmented body parts as an intermediate representation. Saito et al. (2019) developed the 'PIFu' architecture, which produces a 3D occupancy field for the geometry of a person by a multilayer perceptron (MLP) and texture colours by a generative adversarial network (GAN). In 'ARCH', described by Huang et al. (2020), animation capabilities of human models are considered by including a semantic deformation field, amongst others. C.-H. Lin et al. (2020) encode images of objects in a hypernetwork, which is a network generating weights for another network. In the architecture of C.-H. Lin et al. (2020), the hypernetwork predicts parameters of implicit functions for an MLP, which converts encoded 3D coordinates into SDF and RGB values and which is updated by a recurrent neural network.

In a subset of single-view 3D reconstruction networks, coarse and detailed geometry are handled separately. In the deep implicit surface network (DISN), proposed by W. Wang et al. (2019), SDFs are predicted from local and global features, which are extracted from feature maps of an image encoder. Branches for coarse and fine-level geometry exist also in 'PIFuHD' (Saito et al., 2020). The successor of 'PIFu' contains two CNNs, three MLPs, and a conditional GAN predicting normal maps. Li and Zhang (2021) demonstrated with 'D²IM-Net' how to transfer surface details from displacement maps to coarse shapes by one image encoder and two decoder branches as well as including a Laplacian loss function.

While the above networks use 3D training data, another subgroup of object reconstruction networks, also known as off-the-shelf recognition systems, uses only the given 2D images for supervision. S. Liu et al. (2019) elaborated a ray-based field probing technique to correct errors of predicted 3D implicit surfaces. Lunz et al. (2020) added a proxy neural renderer to a GAN to render 2D images by the traditional non-differentiable rendering pipeline. In 'U-CMR', Goel et al. (2020) optimised possible camera views to render meshes and textures of objects and birds. Ye et al. (2021) render images from semi-implicit volumetric representations and only take approximate instance segmentation masks into account for supervision. In 'pixelNeRF' (A. Yu et al., 2021), the volumetric density and colour of objects are implicitly encoded along camera rays by a CNN.

A third subcategory of networks additionally outputs distinct parts or part memberships for the 3D reconstruction. In an early work, Agarwal and Triggs (2006) approximated body parts by cuboids using non-linear regression with Support Vector Machines. In a

more modern architecture, Niu et al. (2018) extracted object parts as cuboids by sequential CNNs recovering masks and hierarchies. Paschalidou et al. (2020) trained a partition network to split objects into two parts, a geometry network to find shape parameters of geometric primitives, and a structure network that links the partitions to the primitives.

Instead of reconstructing the object out of individual shapes, Varol et al. (2018) relied in 'BodyNet' on a voxel-based representation, which is predicted by CNNs together with 2D and 3D poses and 2D body parts. A more fine-grained part membership than a body part segmentation are UV coordinates, which link texture images to the surface of a 3D model. UV coordinates can be predicted from images and also be used for 3D reconstruction (e.g. Güler et al., 2017; Yao et al., 2019).

A last group of networks related to our research is concerned with the reconstruction of objects and figures based on silhouettes or sketches. Di and Yu (2017) propose a stacked hierarchical network consisting of 3D CNNs to create objects from black-and-white silhouette images. Delanoy et al. (2018) reconstruct voxelised objects from sketches with an image encoder-decoder CNN and an updater CNN. Brodt and Bessmeltsev (2022) recover 3D poses from sketched characters by training a 2D pose estimation network and applying an optimisation algorithm focusing on bone tangents, body part contacts, and bone foreshortening. No literature could be found to reconstruct 3D persons or objects from paintings, comics/manga, or maps by machine learning methods.

In this work, we address this shortage by following a bottom-up approach, distantly related to deep local shapes (Chabra et al., 2020), to build pictorial human figures from individual body parts. In a top-down approach, contrariwise, it may be more challenging to identify 3D body parts after having constructed a holistic 3D model. The enclosure of 3D body parts into bounding boxes may accelerate sphere tracing computations and may lead to less storage compared to covering the full 3D space. As the variance of SDF values within the bounding boxes is lower than the variance of the entire body, a more efficient training process and more fine-grained reconstruction results can be expected. For deriving 3D body parts from their 2D silhouettes, we use the DISN architecture (W. Wang et al., 2019) due to its simplicity and adaptability. 3D skeletal points, whose depth coordinates are predicted by another minimalistic network (Martinez et al., 2017), serve as anchor points for creating the 3D body parts. Textures based on the given view are generated by an inpainting network (Grigorev et al., 2019) using UV coordinates predicted by a U-Net (Ronneberger et al., 2015). Finally, the textures are enhanced by a cartoonisation network (X. Wang & Yu, 2020) and an autoencoder (Gondara, 2016). Overall, we aim at providing an easily understandable yet effective pipeline for inferring implicit 3D representations of pictorial figures.

5.3. Data

Generalised 3D body meshes of a female and male person from the SMPL-X dataset (Pavlakos et al., 2019) form the basis for our experiments. In the following, we process the meshes (Figure 5.1) with a Blender plugin provided for the SMPL-X dataset and automate the steps with the Blender scripting API. We assign about 3200 poses from the AGORA dataset (Patel et al., 2021) to the meshes, half to females and the other half to

males. Additionally, we vary height and weight parameters (i.e. 1.40 m & 60 kg, 1.80 m & 75 kg, 2.20 m & 90 kg) for the posed body meshes since pictorial humans may have distorted proportions. Next, we determine the 3D pose points of the mesh by retrieving the bone heads from the skeleton. In total, we extract 20 pose points (head, neck, thorax, pelvis, left/right [l/r] shoulder, l/r elbow, l/r wrist, l/r hand, l/r hip, l/r knee, l/r ankle, l/r foot) and take the midpoint of two other pose points (l/ r eye).

As a further processing step, we split the 3D body mesh into sub-meshes for different body parts (i.e. head, torso, upper arms, lower arms, hands, upper legs, lower legs, feet). For this, we first iterate through the mesh vertices and derive a body part index from the maximum weight associated to each vertex group. Secondly, we iterate through the mesh triangles and assign them the same body part index as the majority of vertices of a triangle. Triangles with the same body part index are then selected and separated from the mesh. To smoothen the spikes at the boundaries, we split the two edges of a boundary triangle at their centre points and assign the resulting smaller triangle to the other body part. Finally, we calculate centre points for the vertices lying at any boundaries and connect them to close any arisen holes. The individual body parts are exported in OBJ format and converted to SDFs by the mesh-to-sdf library (Kleineberg et al., 2021).

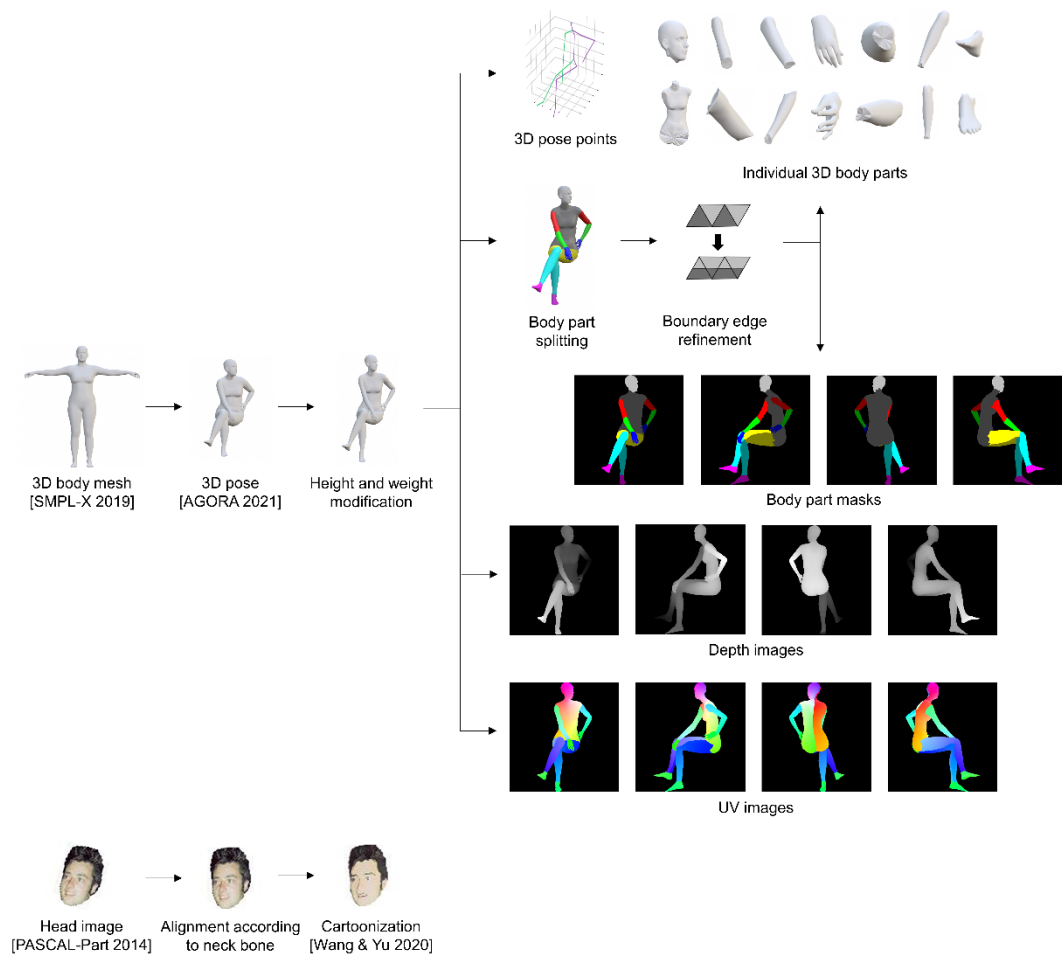


Figure 5.1: Our data processing workflow

After subdividing the body parts into watertight meshes, we create binary mask images for each body part and categorical mask images for all body parts using vertex colours and custom shaders in the rendering pipeline of Blender. Additionally, images with depth values and UV coordinates are generated using ray casting. By repositioning the orthographic camera, four views (i.e. front, left, back, right) are produced for each of the three types of 2D images.

2D body part masks, depth and UV images as well as 3D pose points and SDFs of individual body parts will serve as training and validation data for our networks. For enhancing the textures, we cartoonised the heads of about 3100 humans (X. Wang & Yu, 2020) from the PASCAL-Part dataset (X. Chen et al., 2014). All data items are normalised to equal sizes, but their original size is stored as metadata. As testing data, we annotated 2D skeletons and body part masks of 12 larger figures from historic and contemporary pictorial maps, which mainly originate from a pictorial map classification dataset (Schnürer et al., 2021). The selected test figures vary in poses, clothes, genders, skin colours, drawing styles, and viewing perspectives.

5.4. Methods

5.4.1. 3D pose estimation

We use a network proposed by Martinez et al. (2017) to predict depth coordinates for a 2D skeleton. The network consists only of two blocks of fully connected and dropout layers as well as a residual connection. In the original work, humans are captured by four cameras having a perspective projection. We adapt the network by introducing an orthographic projection, where the depth coordinate of the 3D skeletons is omitted to construct the projected 2D coordinates for our training data. Since pictorial figures are mostly hand-drawn in arbitrary projections and an additional network would have to be trained to estimate the parameters of a perspective camera (e.g. focal length, distortion coefficients), we apply only the orthographic projection to our test data. Nevertheless, we report the results of a hypothetical perspective camera for our validation data. Another minor modification of the original network is the addition of five skeleton keypoints - one at each hand and foot, and one at the head. Those will be helpful for the 3D body part inference in the next step.

We train each network configuration for 100 epochs using the given hyperparameters (i.e. batch size = 64, learning rate = 0.001, dropout = 0.5, batch normalisation) by Martinez et al. (2017) since the authors already tested those extensively in their work. All experiments in this article are conducted on an NVIDIA GeForce GTX 1080 and our custom architectures are implemented with TensorFlow (Google, 2022). We set the height of our figures uniformly to 1.79 meters since this is the average height of the two test subjects in the original dataset according to their body meshes. Quantitative results (Table 5.1) show that the root mean squared errors (RMSE) of the orthographic projection are only slightly higher compared to the perspective one, whereas the percentages of correct keypoints (PCK_{150mm}) are slightly lower. An extreme outlier occurs, especially for the orthographic projection, when our validation data is predicted by the network trained on their data, demonstrating that re-training is necessary. After doing so, a similar accuracy to their original training and validation data is reached. The error

slightly increases when adding the five pose points (i.e. l/r hand, l/r foot, midpoint between the eyes) in our skeletons. Qualitative results show that poses can be recovered well for pictorial figures (Figure 5.2). Only in one out of the 12 test figures, an arm was positioned in front of the body instead of behind it (Figure 5.4, lower row).

Training data	Validation data	Projection	Pose points	RMSE [mm]	PCK _{150mm} [%]
theirs	theirs	perspective	16	44.40	97.56
theirs	theirs	orthographic	16	45.68	96.59
theirs	ours	perspective	16	193.82	45.84
theirs	ours	orthographic	16	808.38	7.03
ours	ours	perspective	16	42.23	96.83
ours	ours	orthographic	16	46.68	95.10
ours	ours	perspective	21	48.82	95.63
ours	ours	orthographic	21	54.85	93.30

Table 5.1: Average root mean squared errors and percentages of correct keypoints for estimating depth coordinates of human poses using their (Martinez et al., 2017) and our data as well as different projections and number of pose points

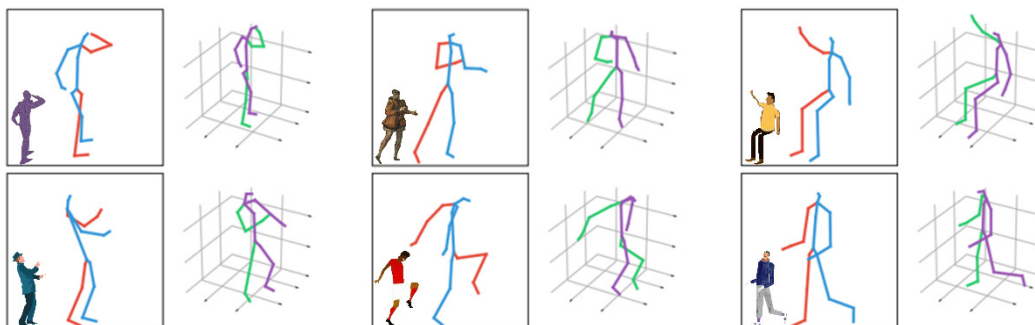


Figure 5.2: Estimated 3D poses (green/violet) from 2D poses (red/blue) of pictorial figures from our test data after training the network of Martinez et al. (2017) with our data (i.e. 21 pose points, orthographic projection)

5.4.2. 3D body part inference

We generate 3D body parts from their 2D masks by a network called DISN (W. Wang et al., 2019). Originally, the network predicts 3D SDFs of objects in 2D images, which are encoded in a series of convolution and downsampling operations to a final feature map (i.e. global features). Intermediate feature maps of the encoding process are up-sampled and concatenated so that local features can be retrieved point-wise. W. Wang et al. (2019) propose two variations for the decoding part: On the one hand, the encoded 3D query points, global and local features are combined and decoded by fully connected layers in DISN one-stream. On the other hand, 3D query points and global features as well as 3D query points and local features are first combined and decoded by fully connected layers in parallel, and finally added in DISN two-stream.

We extend the network by additionally concatenating 3D pose points, which have been estimated in the previous stage, to the global and local features together with the 3D query points. We use two 3D pose points as anchor points for each body part (e.g. elbow and wrist for a lower arm), except for the torso where four points are used (l/r hip, l/r

shoulder). The pose points embed information about the orientation and depth of the body parts. This has the advantage that an initial network proposed by W. Wang et al. (2019) can be omitted, which estimates translation and rotation parameters to transform points from world space into camera space. Since we feed only 64×64 px masks into the adapted network, we reduced its parameters (Text 5.1, Appendix). We sample the same amount of positive and negative SDF values (i.e. 2000 each) to get a distinct zero-iso-surface. The distance values are sampled randomly within the cubic grid to recover coarse structures and fine details of body parts, however leading to an accuracy trade-off in either granularity.

We report errors for inferring body parts with and without pose points for the one-stream and two-stream architecture for hands (Table 5.2). We selected a hand as an exemplary body part for our measurements (Figure 5.3) since fingers are the most difficult structure to recover (Table 5.3). Each configuration is trained five times for 200 epochs at a learning rate of 0.0001 and a batch size of four. Results show that errors are similar for the two architectures, and decrease with the additional pose points in both cases. To construct all body parts (Figure 5.4), we mirror symmetric body parts (e.g. right and left foot).

	DISN one-stream		DISN two-stream	
	without pose points	with pose points	without pose points	with pose points
RMSE	0.063	0.051	0.063	0.050
IoU [%]	47.80	52.63	47.79	53.31

Table 5.2: Average root mean squared errors and intersections over unions on our validation data for inferring 3D SDFs of hands from 2D masks

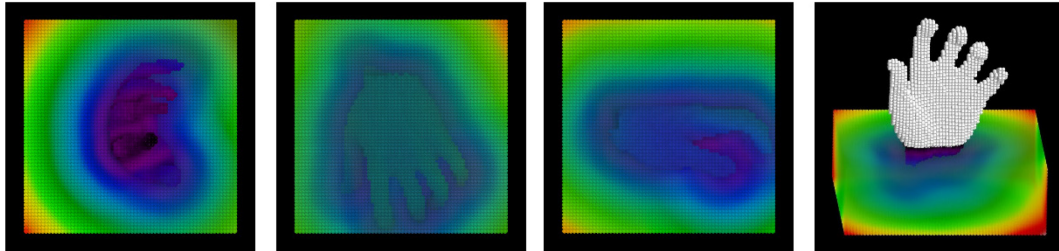


Figure 5.3: SDF around a hand viewed from the top, back and left, and in oblique perspective. Only positive distance values (blue = small, green = intermediate, red = large distances) are coloured.

	Torso	Head	Upper arm	Lower arm	Hand	Upper leg	Lower leg	Foot
RMSE	0.014	0.011	0.024	0.029	0.050	0.015	0.013	0.019
IoU [%]	87.01	93.01	80.42	74.23	53.31	87.65	83.66	81.19

Table 5.3: Average root mean squared errors and intersections over unions on our validation data for inferring 3D SDFs of different body parts from 2D masks using DISN two-stream with pose points

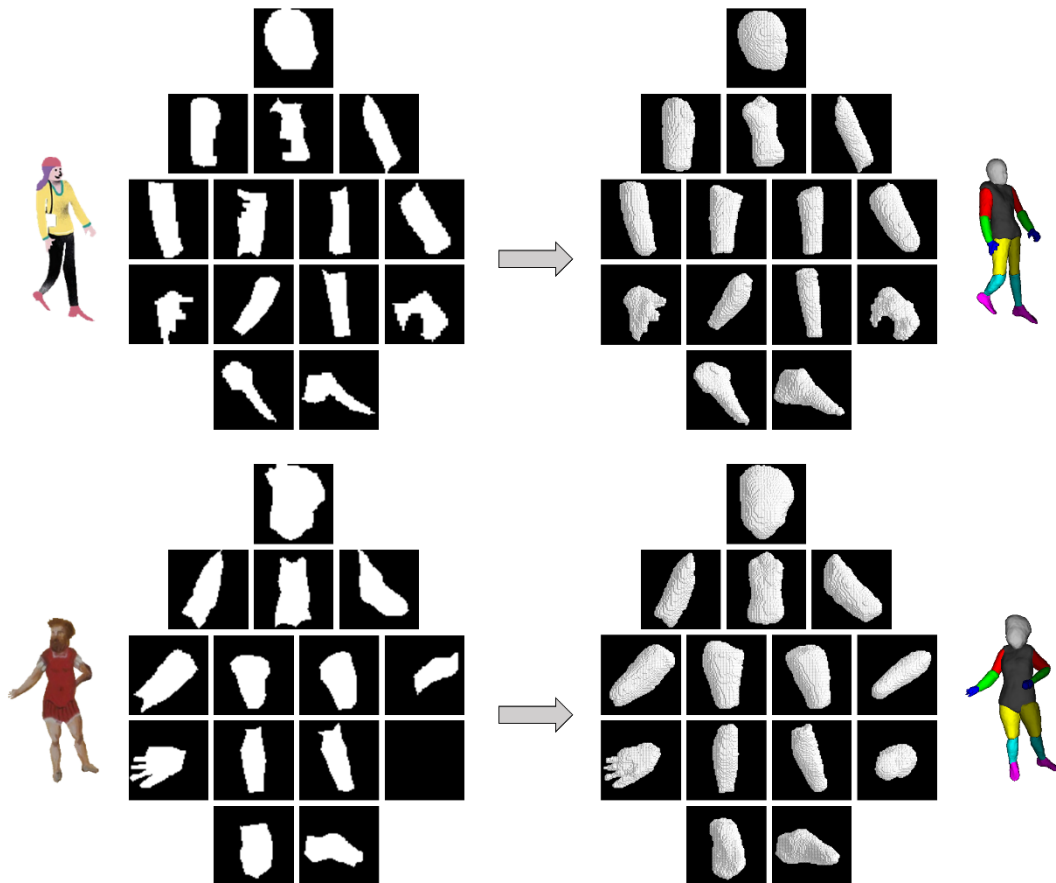


Figure 5.4: Inferred body part SDFs (distance < 0) from masks by DISN one-stream with concatenated pose points. Each network producing a 3D body part is trained individually. The mask of the left hand of the second figure is empty since it is hidden in the original image.

5.4.3. UV coordinates prediction

We predict UV coordinates, ranged zero to one, from a depth image and body part masks of the figures by designing a network similar to U-Net (Ronneberger et al., 2015). The input data is derived from the outputs of the previous two steps. Each 3D body part is positioned at the midpoints of the bones of the 3D skeleton to form the full body. The size of each body part is determined by the height and width of the 2D body part mask as well as enclosed 3D skeleton points. For the latter, a multi-layer perceptron – consisting of three layers with 20, 40, 20 neurons – predicts the size from the enclosed 3D skeleton coordinates to compensate for the lack of depth information of the 2D body mask.

By assembling the inferred 3D body parts (Text 5.2, Appendix), we derive depth images and body part masks for four camera views (i.e. front, left, back, right). The additional front view will be helpful to generate textures for overlapping parts later on. Since the projection of the drawn figures may vary, we simply assume an orthographic projection. We feed depth images and body part masks in batches of eight and resized to 256×256 px into our U-Net-like network (Figure 5.5). Our network consists of 1- and 2-strided convolutions, which are used for down- and upsampling, as well as skip connections. The network is trained for 50 epochs at a learning rate of 0.001 and for

another 50 epochs at a learning rate of 0.0001 with the Adam optimiser. Since the loss converged at similar values, we report the results of one single run. It turned out that the additional body part masks, which are multiplied with the depth image, lead to a lower error compared to training with depth images only for our validation data (Table 5.4). Smooth UV images were produced for our test data (Figure 5.6).

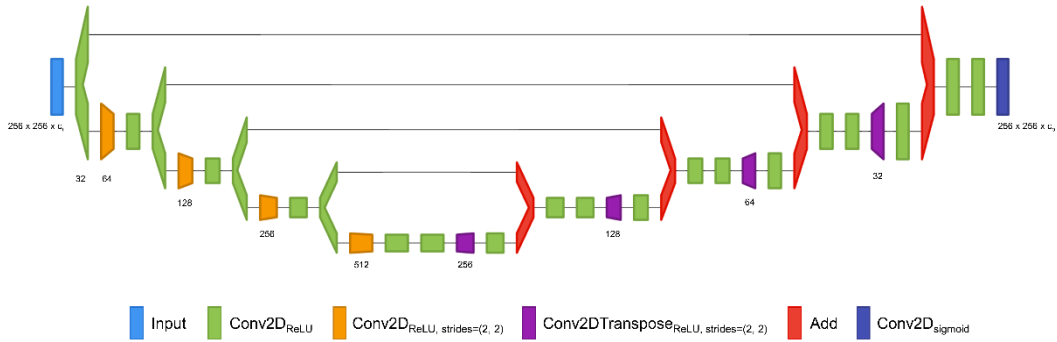


Figure 5.5: Network architecture for predicting UV coordinates from a depth map. $c_i = 1$ for inputting a depth image; $c_i = 14$ for inputting a depth image multiplied by body part masks; $c_o = 3$ for outputting the two UV coordinate channels and a body mask channel used in the loss calculation. Numbers below each layer denote the channel dimension. Figure created by Net2Vis (Bäuerle et al., 2021)

Input data	UV coordinates (MAE)	UV coordinates (RMSE)
Depth image	0.014	0.042
Depth image × Body part masks	0.011	0.031

Table 5.4: Mean absolute errors (MAE) and root mean squared errors (RMSE) for predicting UV coordinates of pictorial human figures from different types of input data

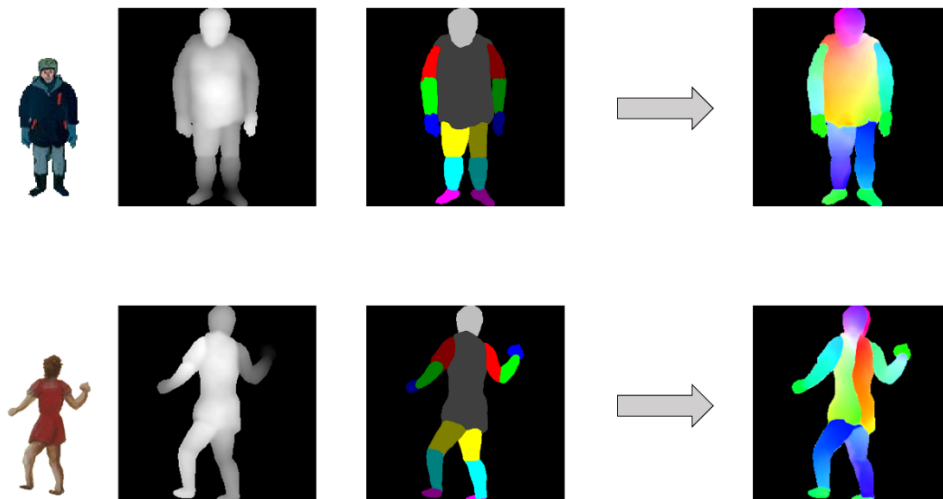


Figure 5.6: Predicted UV coordinates of pictorial human figures from a depth image and body part masks by our fully convolutional network. The body part masks are one-hot encoded in 14 channels for the neural network. The UV coordinates are stored in two channels and have been mapped to a squared colour circle (Figure 5.8) for visualisation purposes.

5.4.4. Texture inpainting and enhancement

We create 256×256 px textures when viewing the figure from behind, the left, and the right by a generative network (Grigorev et al., 2019). Due to minor mismatches of the shape between the predicted UV coordinates in the previous step and the given texture, we input the intersection of both into the network. We add a grey rectangle to the background since the network was trained on human models standing in front of a white wall, which appears greyish due to lighting and shadows. As a post-processing step, we crop the output images to the given body masks.

Since the authors did not publish the code for training their generative network, we could only use a pre-trained version and thus report qualitative results only (Figure 5.7). We apply texture maps, whose colour values were retrieved from the inpainted textures using the predicted UV images (Figure 5.8), to the body parts to render the final images (Text 5.3, Appendix). In general, colours of clothes, skin, and hair were mostly plausibly generated. Artefacts appear for uncommon poses (e.g. sitting, playing football) and at the shoes/feet. Coarse texture structures could be created; however, pictorial black strokes representing folds in the textures were not transferred from the source image. To take these pictorial characteristics into account, a cartoonisation network (X. Wang & Yu, 2020) can be applied to the inpainted textures.

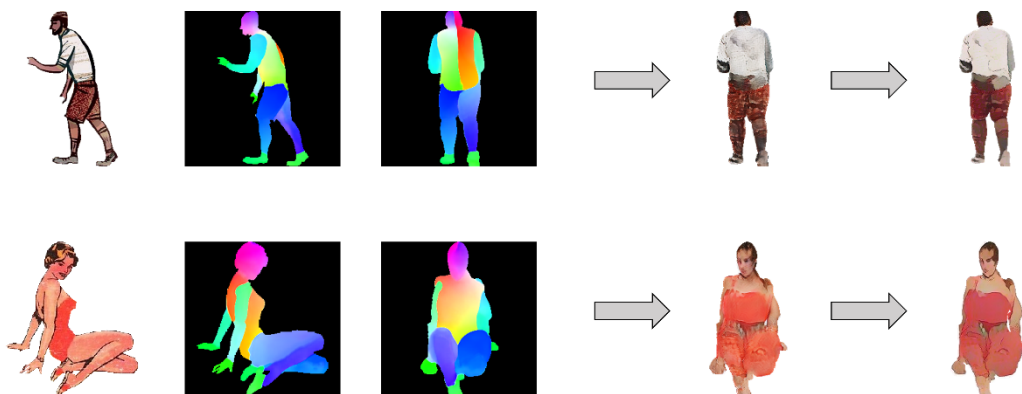


Figure 5.7: Generated textures (already cropped to the input mask) of two pictorial figures from a given image as well as source and target UV maps by the inpainting network. The inpainted image is cartoonised to mimic a more pictorial style.



Figure 5.8: UV coordinates in the inpainted texture (left) and in the colour wheel (right) at the position of the left eye of a pictorial figure (Figure 7, upper row)

Since some of the inpainted faces clearly originate from real humans, we train an autoencoder to map them to a more pictorial style and to possibly recover missing facial details. We establish a shallow branch (Gondara, 2016) for denoising the hue and saturation channel, and a deeper branch including a bottleneck for learning structures and shadings in the value channel (Figure 5.9). Colours (i.e. hue and saturation) and shadings (i.e. value) are weighted equally in the loss function. We augment the realistic input images by blurring and oil painting, by adding noise (i.e. gaussian, salt and pepper) and by varying the hue. The target images have been converted from the input images by the above cartoonisation network. The autoencoder is trained for 200 epochs at a batch size of 32 and a learning rate of 0.001 with the Adam optimiser. During training, head images with varying looks can be obtained (Figure 5.10). Convincingly painted results, however, are rather the exception (i.e. roughly 5% of the generated images).

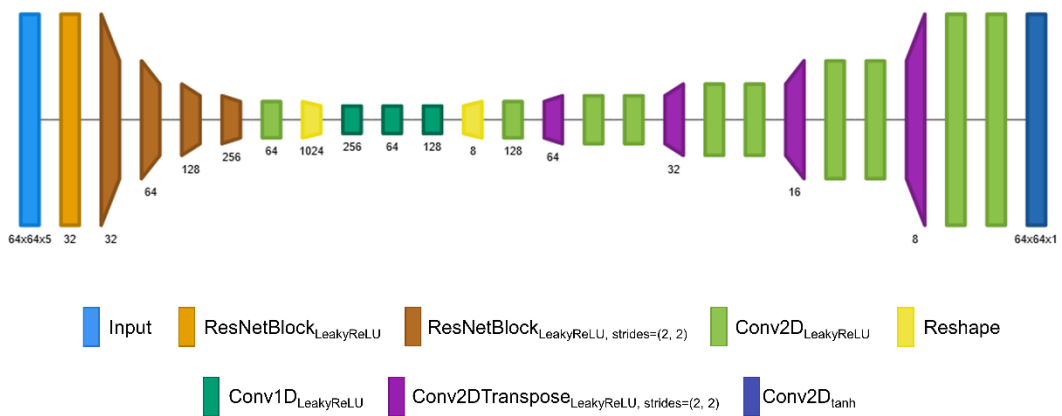


Figure 5.9: Network architecture for enhancing textures of inpainted heads. We input HSV images together with predicted UV coordinates to account for the different orientations of the heads. This branch outputs the value of the colour, whereas hue and saturation are outputted by another branch. Figure created by Net2Vis (Bäuerle et al., 2021)

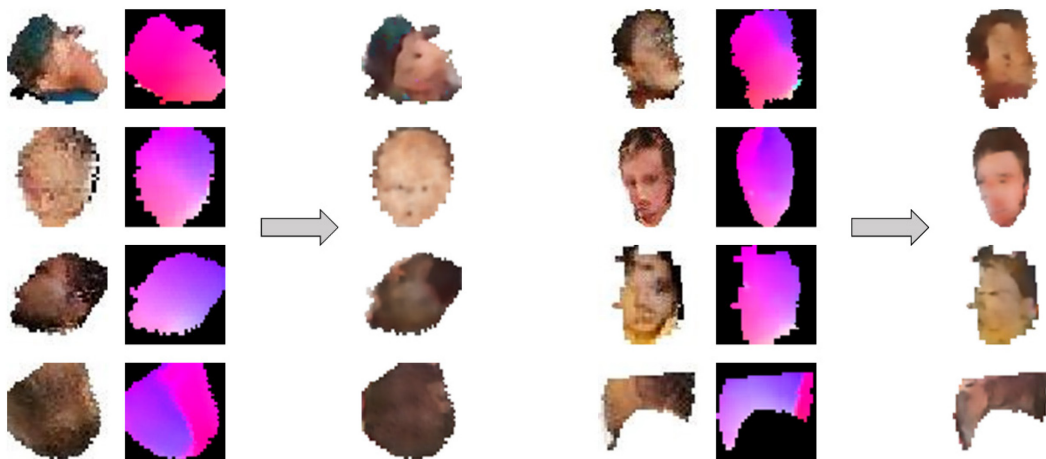


Figure 5.10: Denoised and resynthesised textures of inpainted heads from different views (i.e. right, front, left, back)

5.4.5. Real-time rendering

The inferred figures can be rendered in real-time using the sphere tracing algorithm (Hart, 1996). A 256×256px image is rendered at 25 frames per second (FPS) and a 512×512px image at 11 FPS with an NVIDIA GeForce 1080 GTX even when enabling computationally intensive features (i.e. trilinear interpolation, normal calculation, texture blending). Optionally, we can enable a perspective projection by sending rays from a point location; however, differences to the orthographic projection are only marginal. Also, diffuse lighting can be added by calculating the angle between a virtual light source and the surface normals.

To integrate the figures into existing 3D map environments such as virtual globes, which are mainly based on the traditional rendering pipeline, they can be rendered with transparent background in billboards, while the sphere tracing algorithm is implemented in the fragment shader (Schnürer et al., 2017). Another option is to export a point cloud by returning the 3D coordinates of surface intersections in the ray marcher. The point cloud can be further turned into a triangle mesh by the ball-pivoting algorithm (Bernardini et al., 1999). We exemplarily illustrated the outcome of the latter two conversion steps by placing the 3D figures on the original map in a 3D modelling software (Figure 5.11) and a virtual globe toolkit (Figure 5.12).



Figure 5.11: Remeshed pictorial 3D figure placed on the original map (Owen, 2015) in Blender (Blender Foundation, 2022)

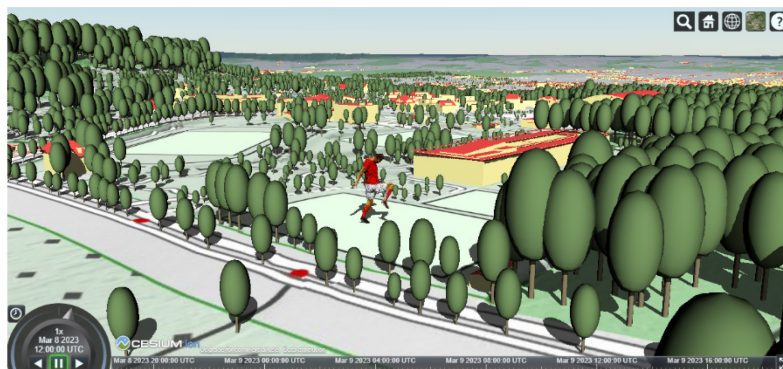


Figure 5.12: Remeshed pictorial 3D figure in the virtual globe CesiumJS (Cesium GS, 2022). The figure is placed in front of the FIFA headquarters, similar to the original map (Flynn, n.d.). 2D base map, 3D buildings and trees originate from the GeoAdmin API (swisstopo, 2022).

5.5. Use case

We see a large potential in adding the constructed figures as protagonists or secondary characters to story maps. Particularly, 3D maps convey the topography vividly and allow channelling of the depicted topic by occlusions (e.g. mountains, trees, fog). Instead of presenting multimedia content in overlays (e.g. Matt, 2019), we suggest placing essential figures or other objects directly and in a consistent style on the map to support the story.

In the following, we outline how a story map including characters, animals, and additional 3D objects may be designed for children (Figure 5.13). We take Charles Darwin's (1839) round-the-world journey as an example, specifically a stop in Port St. Julian, Patagonia, in January 1834. After a short introduction to this setting, the user can visit different places in any order. We provide different incentives to follow the story and to interact with characters and the environment (Table 5.5). The story ends after having explored all places.

Pedagogically, our proposed story map may improve map and visualisation literacy (e.g. route planning, interpretation of climate diagrams), and may help to correct misconceptions (e.g. Patagonians perceived as giants). It would connect interdisciplinary fields, such as biology (e.g. penguins), history (e.g. early explorers), and ethnology (e.g. indigenous people). Although some gamification elements are included, it is intended to put the focus on the scientific aspects. Our presented pipeline of different machine learning models will help map creators in constructing and potentially animating 3D human figures based on a given 2D template. With some artistic skills, the 2D figures may be drawn by the map creator instead of reusing the works of others.

The sketched story map may be experienced on desktop and tablet computers, but also with head-mounted displays. Users of the latter devices perceive the 3D environment in virtual reality, what allows introducing metaverse concepts, such as avatars. For instance, the quizzes may be solved together with other students or a teacher may give hints or explanations. Their avatars may also be reconstructed from 2D figures and the user's movements in reality may be mirrored, possibly by additional tracking systems (e.g. camera, gesture controllers). In the future, we anticipate an increase in these kinds of virtual excursions and learning activities for geography classes since they are more affordable than visiting the location in the real world and more intriguing than reading a textbook or watching a video.

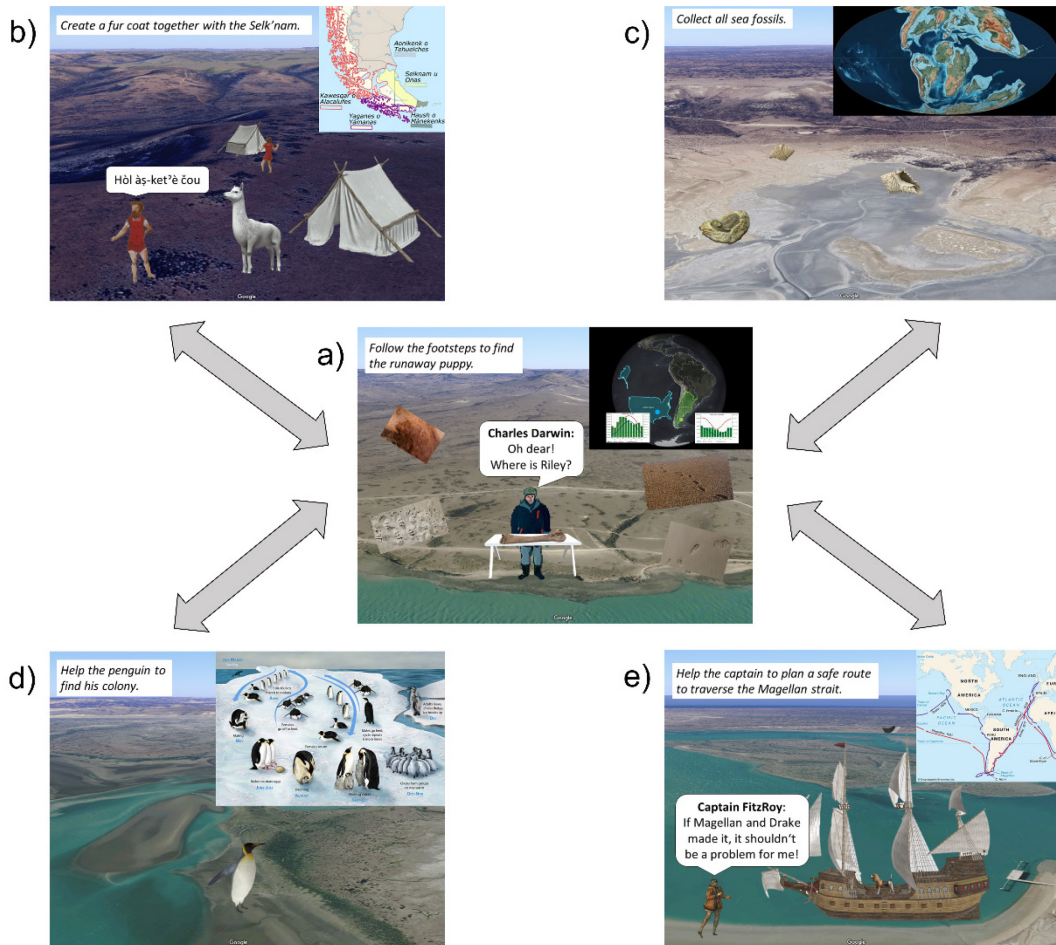


Figure 5.13: Sketched story map for children about Charles Darwin in Patagonia (sources: Table 5.6, Appendix). The start location is in the middle and possible places to explore are at the corners. At each place, a puzzle needs to be solved after being provided with some contextual information (top right graphic of each scene). The reconstructed figures will be part of the story.

	Possible storytelling means	Occurrence in Figure 5.13
Attraction	having an overarching goal (e.g. finding a runaway dog)	a)
	curiosity to explore the remote area (e.g. by following footsteps)	a)
Affection	smooth transitions between local and global scale (e.g. by giving background information to the current scene)	a), b), c), d), e)
	temporal animations (e.g. plate tectonics)	c), d), e)
Interaction	personal stories of characters (e.g. the captain)	b), e)
	small tasks to fulfil (e.g. assembling a piece of clothing)	b), c), d), e)
Comeback	variations in tasks and questions (e.g. by randomisation)	b), c), d), e)
	a different ending (e.g. the runaway dog reappears at different locations)	b), c), d), e)

Table 5.5: Application of storytelling concepts (Thöny et al., 2018) to our sketched story map

5.6. Discussion

Our work is a continuation of another computer vision experiment for pictorial maps (Schnürer et al., 2022), where silhouettes of figures, their body parts, and pose points have been segmented by two neural networks. Therefore, it can be assumed that these data items can be automatically extracted. Nevertheless, we annotated our test data manually to have a solid foundation for the current experiment. For our training data, we varied the sizes and weights of body meshes of real humans, which had a positive impact on reconstructing a test figure with thin long limbs and a small head. The consideration of clothes would definitely improve the quality of reconstruction, for instance, our current pipeline does not support hats. However, publicly available training datasets of clothed humans did not contain 3D body meshes or body part segmentations. Beyond, some clothes (e.g. long skirts) would behave differently than the underlying body parts, but we aimed primarily at skeletal animation as a follow-up use case.

Besides selecting and enhancing the training data, the structure of the different networks may be modified. We showed that increasing the number of pose points from 16 to 21 led to a one centimeter higher error for the 3D pose estimation network, which is still tolerable and provided a benefit to the reconstruction of hands, feet, and the head. We used captured poses of mainly standing persons as training data; alternatively, bones of a skeleton may be oriented according to a range of possible joint angles (Soucie et al., 2011) to better handle uncommon poses. We did not consider including pose points of fingers, which would have been available in the SMPL-X training dataset, since the hands of pictorial figures are usually small, their fingers are not easily distinguishable, and sometimes contain less than five fingers. Instead of estimating the 3D coordinates directly, relative rotation angles of joints or limbs may be predicted; however, this would require more complex and constrained network designs, as noted by Martinez et al. (2017). Estimating more camera parameters would probably increase the prediction accuracy, yet we achieved satisfactory results by simply assuming an orthographic projection, similar to other works (e.g. Z. Huang et al., 2020).

While no fine-tuning was necessary for the pose estimation network, we carefully configured DISN to infer body parts. We also examined normalizing the data (e.g. sqrt/log transform), using 3D deconvolutions (instead of query points), sampling points near the surface (instead of equally spaced grid points), predicting a top and side view (additionally to the front view), outputting a binary field (additionally to the SDF); but those did not significantly improve the reconstruction quality. Generally, silhouette-based 3D reconstruction is a more difficult task than a texture-based recovery since textures may contain depth information; thus, the qualitative results are only partly comparable to those of the original network. The addition of pose points helped to recover partly or totally hidden body parts, though not visible hands are approximated by ellipsoids, which is the average shape. Other issues concern the rather realistic forms of the body parts and the gaps between them (Figure 5.14).

Predicting the UV coordinates from the depth map was a straightforward task by using a fully convolutional network similar to U-Net (Ronneberger et al., 2015). Inputting depth maps only would have been already sufficient to get a smooth image for the validation data (i.e. real humans). However, on our test data (i.e. pictorial humans), where the depth map is derived from the inferred 3D body parts, stains appeared on the UV image, which could be remedied by additionally feeding the masks of the body parts into U-Net. We

refrained from predicting UV coordinates or even texture colours together with the body parts because “colour prediction is a non-trivial task as RGB colours are defined only on the surface while the [signed distance] field is defined over the entire 3D space” (Saito et al., 2019, p. 2308).

Although the inpainting network is biased toward generating textures of real humans, it produced adequate results on a coarse level for pictorial humans. For better matching the texture with the input, the potential of symmetries could be exploited (Y. Zhou et al., 2021). Furthermore, texture maps probably need to be deformed (Shu et al., 2018) in case the body part shapes deviate much. Since texture mappings for occluded body parts may be incorrect when viewing the figure from another than the four perspectives, one may increase the number of textures to prevent these artefacts. Due to the lack of adequate training data, we cartoonised photos of humans to train our autoencoder; however, the generated textures are still quite realistic. Our autoencoder is able to recover some facial details, yet a more expressive latent space may be learned by a variational autoencoder. Overall, an end-to-end network would be desirable instead of our four consecutive networks to benefit from synergy effects.

To realise our exemplary use case, existing story map editors (e.g. ESRI, 2022) would need to be extended to support different storylines, game templates, and textual options. The availability of a 3D model store would facilitate the reusability and the copyright management of the reconstructed figures and other objects. Positioning and viewing perspective, animation parameters and triggers, interactions with the map content and other characters, and possibly cartographic functions may be defined for the characters in the story map editor. In cognitive experiments, optimal parameters (e.g. animation speed) would need to be clarified in detail and whether the approach including figures in story maps generally offers an added value to the user.




Problem	Visual example	Possible solution
Gaps may appear between body parts		Refine transformation parameters (i.e. translation, rotation, scaling) of body parts by optimization
Non-visible hands are approximated by ellipsoids		Adapt a hand template based on the lower arm thickness or by parameters of the visible hand
More abstract geometric shapes are not supported		Add synthetic training data (e.g. cuboids, ellipsoids) and learn a 3D deformation field, which uses body parts of real humans as templates

Figure 5.14: Failure cases of inferred body parts

5.7. Summary and future work

In this work, we generated implicit 3D representations of human figures appearing on pictorial maps using machine learning methods (Figure 5.15). We showed that plausible poses and body parts can be inferred when training the networks with data of real humans. However, we see a need for improvement in refining shapes and textures, in supporting hair and clothes as well as in simplifying the workflow. Our automated workflow takes only a few minutes to complete, whereas manual sculpting and texturing of the 3D models would take several hours.

Next to human figures, also other pictorial entities like animals, sea monsters, or ships may be transferred to the third dimension. Moreover, 3D buildings or distinct landscape features may be derived from historic maps in oblique view (e.g. Murer, 1576). Cartography would be rather concerned with abstract and georeferenced representations while the reconstruction of detailed representations would rather fall into the domains of other fields (e.g. archaeology, anthropology, evolution biology).

The automatic 3D reconstruction of buildings and landscapes will accelerate the development of 'time travel' applications, where users can see the past and future of a geographic area (e.g. Stadt Zürich, 2022). When additionally viewing a 3D city in virtual reality, users will get a more vivid impression of its structure (e.g. the narrowness of alleys). As envisioned in the concept of the metaverse, the virtual space may be populated with avatars, where our animation-ready 3D pictorial human figures come into operation. We would propose the term 'cartoverse' for this kind of cartographic metaverse. Several challenges would need to be addressed in the future to provide a fully functional cartoverse, for instance, how to avoid motion sickness during spatial navigation or for different perspectives, how to interact with 3D objects via speech or gesture recognition, or how to present thematic information additionally to the topographic elements.

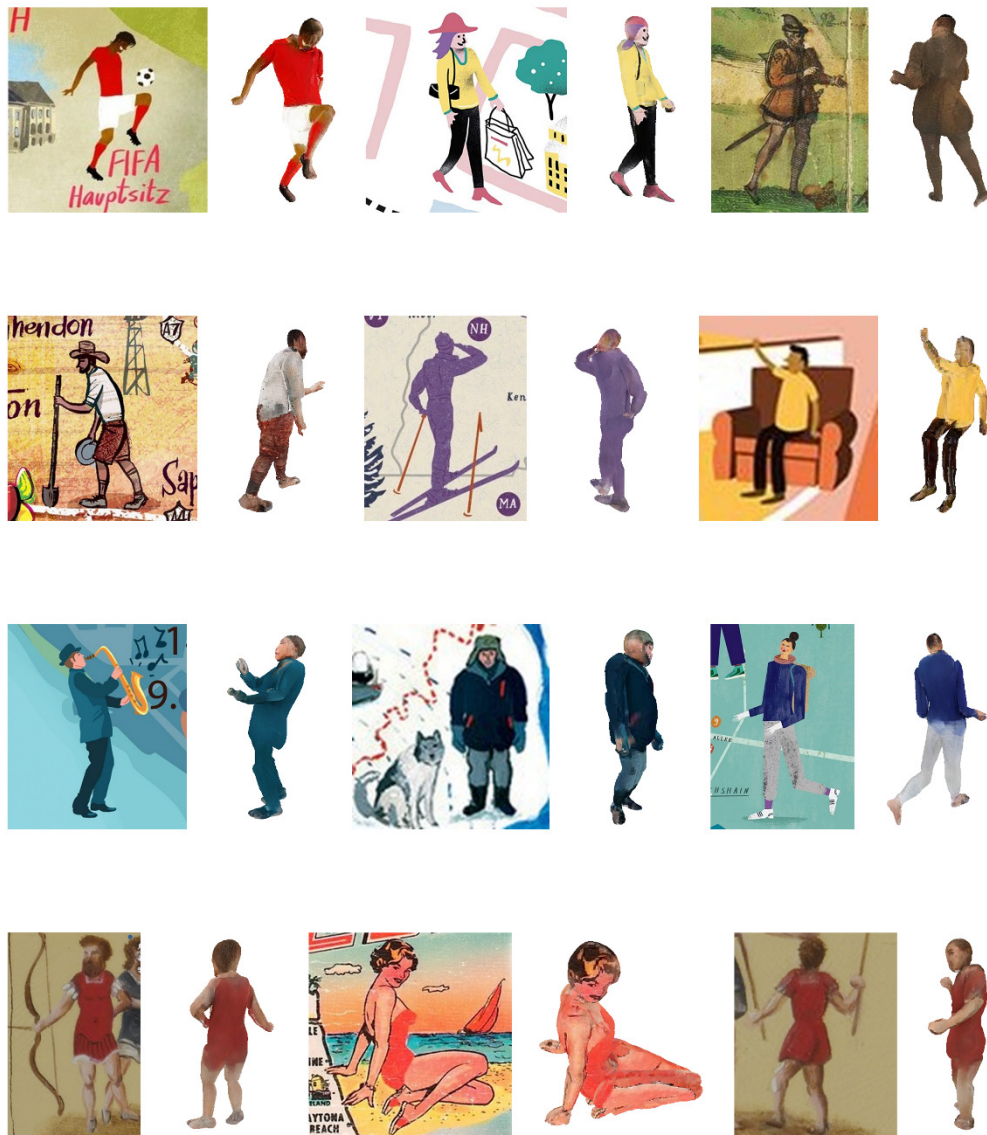


Figure 5.15: All pictorial human figures on maps from our test dataset and our inferred 3D models from different views

Appendix

Query points and additional pose points, which are inputted into DISN, are encoded by three 1D convolutions with four times smaller filter sizes (i.e. 16, 64, 128). In the image encoder, we reduced the number of filters for convolutions by four (i.e. 8, 16, 32, 64, 128) and use only one convolution layer after a strided convolution layer at each level. Similarly, we lowered the size of the global features by four (i.e. 256). The filter sizes for 1D convolutions in the decoder part remained unchanged (i.e. 512, 256, 1).

Text 5.1: Adaptations to the Deep Implicit Surface Network (DISN) for 3D body part inference

We implemented a custom ray marcher in Python using the library ‘Numba’, which enables executing parallel operations on the GPU, for rendering the inferred 3D objects represented by SDFs. An SDF value denotes the shortest distance (d) to object surfaces, while the sign indicates whether a point lies inside ($d < 0$), on ($d = 0$), or outside ($d > 0$) the object. We trilinearly interpolate the eight closest grid points of the evenly spaced SDF to get smoother surfaces. The SDFs of the different body parts can be combined by the union operation (i.e. $\min(d_1, d_2)$). Afterwards, a depth image can be obtained by iteratively cumulating the covered distances from the virtual camera to the body surface during each ray marching step. As an enhancement, we smooth the depth map with a 5×5 averaging filter. A body part mask is produced by returning the index of the first hit body part along the ray.

Text 5.2: Derivation of depth maps and body part masks for UV coordinates prediction

The ray marcher (Text 5.2) is extended by projecting the inpainted texture maps to the surfaces of the 3D body parts from four views (i.e. front, back, left, right). As an enhancement, we blend the obtained textures, that is, the steeper the angle of the surface normal to the texture, the more weight the colour value gains from this texture. The normals are approximated by nearby SDF values at the surface points (i.e. $n = [SDF(x+\epsilon, y, z) - SDF(x-\epsilon, y, z), SDF(x, y+\epsilon, z) - SDF(x, y-\epsilon, z), SDF(x, y, z+\epsilon) - SDF(x, y, z-\epsilon)]$). For an interactive view, we pass the rendered images to the canvas of the library ‘matplotlib’, where mouse events can be captured to zoom and rotate around the depicted figure.

Text 5.3: Application of textures to figures

Maps	
Background maps	https://www.google.com/maps
Globe for comparing sizes of countries	https://arnofiva.github.io/world-sizes/
Natives in Patagonia	https://commons.wikimedia.org/wiki/File:Pueblos_ind%C3%ADgenas_de_la_Patagonia_Austral.svg
Lifecycle of penguins	https://de.wikipedia.org/wiki/Datei:Penguin-lifecycle-de.svg#/media/Datei:PENGUIN_LIFECYCLE_H.JPG
Plate tectonics	(Scotese, 2016)
Seafarers' voyages	https://www.britannica.com/biography/Ferdinand-Magellan/Circumnavigation-of-the-globe
Images	
Persons	Sources are given in our test dataset
Penguin footprints	https://commons.wikimedia.org/wiki/File:Penguin_footprints_on_the_beach_%285565682274%29.jpg
Human footprints	https://commons.wikimedia.org/wiki/File:Punta_Prosciutto_footsteps.jpg
Large footprints	https://commons.wikimedia.org/wiki/File:Nodosaur_Footprint_Verified_-_Detail_of_Baby_Footprint_(7846740914).jpg
Small footprints	https://commons.wikimedia.org/wiki/File:Footsteps_on_the_beach,_Seaford_-_geograph.org.uk_-_2599216.jpg
Climate charts	https://www.climatestotravel.com/
3D models	
Bone	https://sketchfab.com/3d-models/horse-bone-d51e7216a5bb41bea3ee2576fa92eedc
Table	https://sketchfab.com/3d-models/table-for-building-91d5f0058b734eafad16a3e43070ebe6
Alpaca	https://sketchfab.com/3d-models/alpaca-non-commercial-5de8754563254e34837ec4aacac8632e
Tent	https://sketchfab.com/3d-models/tent-fa46028e8d3849399ba5271df07ed99c
Penguin	https://sketchfab.com/3d-models/emperor-penguin-310f1d21cf534fd0bcf073aa9b08a740
Fossil	https://sketchfab.com/3d-models/trilobite-fossil-b4c9b051a23445bfafcc9953deb54cc5
Shell	https://sketchfab.com/3d-models/seashell-fossil-bc4b85625dd045608b498c41f8b5c1a7
Sailing ship	https://sketchfab.com/3d-models/segelschiff-sailing-ship-updated-pbr-f349ecd6b81c4c25aa5d628e8913048e
Dog	https://sketchfab.com/3d-models/beagle-341cb7dd930a427eab3f8e925718e6c0
Miscellaneous	
Selk'nam language	https://ids.clld.org/contributions/311

Table 5.6: Sources of the 3D story map with pictorial figures

References

- Agarwal, A., & Triggs, B. (2006). Recovering 3D human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1), 44–58. <https://doi.org/10.1109/TPAMI.2006.21>
- Bäuerle, A., Van Onzenoodt, C., & Ropinski, T. (2021). Net2vis—a visual grammar for automatically generating publication-tailored cnn architecture visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 27(6), 2980–2991. <https://doi.org/10.1109/TVCG.2021.3057483>
- Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., & Taubin, G. (1999). The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5(4), 349–359. <https://doi.org/10.1109/2945.817351>
- Blender Foundation. (2022). *Blender*. <https://www.blender.org/>
- Brodthorn, K., & Bessmeltsev, M. (2022). Sketch2Pose: Estimating a 3D Character Pose from a Bitmap Sketch. *ACM Transactions on Graphics*, 41(4). <https://doi.org/10.1145/3528223.3530106>
- Cesium GS. (2022). *CesiumJS*. Cesium. <https://cesium.com/platform/cesiumjs/>
- Chabra, R., Lenssen, J. E., Ilg, E., Schmidt, T., Straub, J., Lovegrove, S., & Newcombe, R. (2020). Deep Local Shapes: Learning Local SDF Priors for Detailed 3D Reconstruction. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), *Computer Vision – ECCV 2020* (pp. 608–625). Springer International Publishing. https://doi.org/10.1007/978-3-030-58526-6_36
- Chen, X., Mottaghi, R., Liu, X., Fidler, S., Urtasun, R., & Yuille, A. (2014). Detect What You Can: Detecting and Representing Objects using Holistic Models and Body Parts. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1971–1978. <https://doi.org/10.1109/CVPR.2014.254>
- Child, H. (1956). *Decorative Maps* (1st edition). Studio Publications.
- Darwin, C. (1839). *The Voyage of the Beagle*. Project Gutenberg eBook. <https://www.gutenberg.org/files/944/944-h/944-h.htm>
- Delanoy, J., Aubry, M., Isola, P., Efros, A. A., & Bousseau, A. (2018). 3D Sketching using Multi-View Deep Volumetric Prediction. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 1(1), 21:1–21:22. <https://doi.org/10.1145/3203197>
- Di, X., & Yu, P. (2017). *3D Reconstruction of Simple Objects from A Single View Silhouette Image*. arXiv. <https://doi.org/10.48550/arXiv.1701.04752>
- Döllner, J., & Walther, M. (2003). Real-time Expressive Rendering of City Models. *Proceedings on Seventh International Conference on Information Visualization*, 4, 245–250. <https://doi.org/10.1109/IV.2003.1217986>
- Eslami, S. M. A., Jimenez Rezende, D., Besse, F., Viola, F., Morcos, A. S., Garnelo, M., Ruderman, A., Rusu, A. A., Danihelka, I., Gregor, K., Reichert, D. P., Buesing, L., Weber, T., Vinyals, O., Rosenbaum, D., Rabinowitz, N., King, H., Hillier, C., Botvinick, M., ... Hassabis, D. (2018). Neural scene representation and rendering. *Science*, 360(6394), 1204–1210. <https://doi.org/10.1126/science.aar6170>
- ESRI. (2022). *ArcGIS StoryMaps*. <https://storymaps.arcgis.com/>
- Evangelidis, K., Papadopoulos, T., Papatheodorou, K., Mastorokostas, P., & Hilaris, C. (2018). 3D geospatial visualizations: Animation and motion effects on spatial objects. *Computers & Geosciences*, 111, 200–212. <https://doi.org/10.1016/j.cageo.2017.11.007>
- Fahim, G., Amin, K., & Zarif, S. (2021). Single-View 3D reconstruction: A Survey of deep learning methods. *Computers & Graphics*, 94, 164–190. <https://doi.org/10.1016/j.cag.2020.12.004>
- Flynn, D. (n.d.). *Zurich City Map*. Pinterest. <https://www.pinterest.co.uk/pin/469922542351340326/>
- Gilbert, B. (2021). Zuckerberg is most worried about Apple, Google, Microsoft, Sony and others as the main competition for the ‘metaverse’. *Business Insider*. <https://www.businessinsider.com/facebook-says-apple-sony-microsoft-google-are-metaverse-competition-2021-11>
- Goel, S., Kanazawa, A., & Malik, J. (2020). Shape and Viewpoint Without Keypoints. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), *Computer Vision – ECCV 2020* (pp. 88–104). Springer International Publishing. https://doi.org/10.1007/978-3-030-58555-6_6

- Gondara, L. (2016). Medical Image Denoising Using Convolutional Denoising Autoencoders. *IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, 241-246. <https://doi.org/10.1109/ICDMW.2016.0041>
- Google. (2022). *TensorFlow*. <https://www.tensorflow.org/>
- Grigorev, A., Sevastopolsky, A., Vakhitov, A., & Lempitsky, V. (2019). Coordinate-Based Texture Inpainting for Pose-Guided Human Image Generation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12127-12136. <https://doi.org/10.1109/CVPR.2019.01241>
- Güler, R. A., Trigeorgis, G., Antonakos, E., Snape, P., Zafeiriou, S., & Kokkinos, I. (2017). DenseReg: Fully Convolutional Dense Shape Regression In-the-Wild. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2614-2623. <https://doi.org/10.1109/CVPR.2017.280>
- Hart, J. C. (1996). Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces. *The Visual Computer*, 12(10), 527-545. <https://doi.org/10.1007/s003710050084>
- Heitzler, M., & Hurni, L. (2020). Cartographic reconstruction of building footprints from historical maps: A study on the Swiss Siegfried map. *Transactions in GIS*, 24(2), 442-461. <https://doi.org/10.1111/tgis.12610>
- Herold, H., & Hecht, R. (2018). 3D Reconstruction of Urban History Based on Old Maps. In S. Münster, K. Friedrichs, F. Niebling, & A. Seidel-Grzesińska (Eds.), *Digital Research and Education in Architectural Heritage* (pp. 63-79). Springer International Publishing. https://doi.org/10.1007/978-3-319-76992-9_5
- Huang, Z., Xu, Y., Lassner, C., Li, H., & Tung, T. (2020). ARCH: Animatable Reconstruction of Clothed Humans. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3090-3099. <https://doi.org/10.1109/CVPR42600.2020.00316>
- Kleineberg, M., Sundt, P. B., & Davies, T. (2021). *Mesh-to-sdf*. https://github.com/marian42/mesh_to_sdf
- Lassner, C., & Zollhöfer, M. (2021). Pulsar: Efficient Sphere-based Neural Rendering. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1440-1449. <https://doi.org/10.1109/CVPR46437.2021.00149>
- Li, M., & Zhang, H. (2021). D²IM-Net: Learning Detail Disentangled Implicit Fields from Single Images. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10241-10250. <https://doi.org/10.1109/CVPR46437.2021.01011>
- Lin, C.-H., Wang, C., & Lucey, S. (2020). SDF-SRN: Learning Signed Distance 3D Object Reconstruction from Static Images. *NIPS'20: Proceedings of the 34th International Conference on Neural Information Processing Systems*, Article 961, 11453-11464. <https://dl.acm.org/doi/10.5555/3495724.3496685>
- Liu, S., Saito, S., Chen, W., & Li, H. (2019). Learning to infer implicit surfaces without 3D supervision. *NIPS'19: Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Article 745, 8295-8306. <https://dl.acm.org/doi/abs/10.5555/3454287.3455032>
- Lorensen, W. E., & Cline, H. E. (1987). Marching cubes: A high resolution 3D surface construction algorithm. *ACM SIGGRAPH Computer Graphics*, 21(4), 163-169. <https://doi.org/10.1145/37402.37422>
- Lunz, S., Li, Y., Fitzgibbon, A., & Kushman, N. (2020). *Inverse Graphics GAN: Learning to Generate 3D Shapes from Unstructured 2D Data*. arXiv. <https://doi.org/10.48550/arXiv.2002.12674>
- Martinez, J., Hossain, R., Romero, J., & Little, J. J. (2017). A Simple Yet Effective Baseline for 3d Human Pose Estimation. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2659-2668. <https://doi.org/10.1109/ICCV.2017.288>
- Matt, A. (2019). *Charles Darwin and the Voyage of the HMS Beagle*. https://scout.wisc.edu/archives/r50646/charles_darwin_and_the_voyage_of_the_hms_beagle
- Matthys, M., De Cock, L., Vermaut, J., Van de Weghe, N., & De Maeyer, P. (2021). An "Animated Spatial Time Machine" in Co-Creation: Reconstructing History Using Gamification Integrated into 3D City Modelling, 4D Web and Transmedia Storytelling. *ISPRS International Journal of Geo-Information*, 10(7), Article 460. <https://doi.org/10.3390/ijgi10070460>
- Meta. (2021). *The Metaverse and How We'll Build It Together*. Connect 2021. <https://www.youtube.com/watch?v=Uvufun6xer8>

- Murer, J. (1576). *Murerplan*. Wikipedia. <https://en.wikipedia.org/wiki/Murerplan>
- Naz, A. (2005). *3D interactive pictorial maps* [Master's thesis]. Texas A&M University.
- Niu, C., Li, J., & Xu, K. (2018). Im2Struct: Recovering 3D Shape Structure from a Single RGB Image. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4521–4529. <https://doi.org/10.1109/CVPR.2018.00475>
- Omran, M., Lassner, C., Pons-Moll, G., Gehler, P., & Schiele, B. (2018). Neural Body Fitting: Unifying Deep Learning and Model Based Human Pose and Shape Estimation. *International Conference on 3D Vision*, 484–494. <https://doi.org/10.1109/3DV.2018.00062>
- Owen, N. (2015). *Queensland: National Geographic's 'Traveller' Mag*. Behance. <https://www.behance.net/gallery/30454283/Queensland-National-Geographics-Traveller-Mag>
- Park, S.-M., & Kim, Y.-G. (2022). A Metaverse: Taxonomy, Components, Applications, and Open Challenges. *IEEE Access*, *10*, 4209–4251. <https://doi.org/10.1109/ACCESS.2021.3140175>
- Paschalidou, D., Van Gool, L., & Geiger, A. (2020). Learning Unsupervised Hierarchical Part Decomposition of 3D Objects From a Single RGB Image. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1057–1067. <https://doi.org/10.1109/CVPR42600.2020.00114>
- Patel, P., Huang, C.-H. P., Tesch, J., Hoffmann, D. T., Tripathi, S., & Black, M. J. (2021). AGORA: Avatars in Geography Optimized for Regression Analysis. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13463–13473. <https://doi.org/10.1109/CVPR46437.2021.01326>
- Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A. A., Tzionas, D., & Black, M. J. (2019). Expressive Body Capture: 3D Hands, Face, and Body From a Single Image. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10967–10977. <https://doi.org/10.1109/CVPR.2019.01123>
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *Medical Image Computing and Computer-Assisted Intervention* (pp. 234–241). Springer International Publishing. https://doi.org/10.1007/978-3-319-24574-4_28
- Saito, S., Huang, Z., Natsume, R., Morishima, S., Li, H., & Kanazawa, A. (2019). PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization. *2019 IEEE/CVF International Conference on Computer Vision (CVPR)*, 2304–2314. <https://doi.org/10.1109/ICCV.2019.00239>
- Saito, S., Simon, T., Saragih, J., & Joo, H. (2020). PIFuHD: Multi-Level Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 81–90. <https://doi.org/10.1109/CVPR42600.2020.00016>
- Schnürer, R., Eichenberger, R., Sieber, R., & Hurni, L. (2017). Animations for 3D Solid Charts in a Virtual Globe - Techniques, Use cases, and Implementation. *28th International Cartographic Conference (ICC)*. https://www.researchgate.net/publication/335014544_Animations_for_3D_Solid_Charts_in_a_Virtual_Globe_-_Techniques_Use_cases_and_Implementation
- Schnürer, R., Öztireli, A. C., Heitzler, M., Sieber, R., & Hurni, L. (2022). Instance Segmentation, Body Part Parsing, and Pose Estimation of Human Figures in Pictorial Maps. *International Journal of Cartography*, *8*(3), 291–307. <https://doi.org/10.1080/23729333.2021.1949087>
- Schnürer, R., Sieber, R., Schmid-Lanter, J., Öztireli, A. C., & Hurni, L. (2021). Detection of Pictorial Map Objects with Convolutional Neural Networks. *The Cartographic Journal*, *58*(1), 50–68. <https://doi.org/10.1080/00087041.2020.1738112>
- Schrotter, G., & Hürzeler, C. (2020). The Digital Twin of the City of Zurich for Urban Planning. *Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, *88*(1), 99–112. <https://doi.org/10.1007/s41064-020-00092-2>
- Scotese, C. (2016). *Plate Tectonics, Paleogeography, and Ice Ages, (Modern World–540Ma)*, YouTube Animation. https://youtu.be/g_iEWvtKcuQ
- Shu, Z., Sahasrabudhe, M., Alp Güler, R., Samaras, D., Paragios, N., & Kokkinos, I. (2018). Deforming Autoencoders: Unsupervised Disentangling of Shape and Appearance. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), *Computer Vision – ECCV 2018* (pp. 664–680). Springer International Publishing. https://doi.org/10.1007/978-3-030-01249-6_40

- Soucie, J. M., Wang, C., Forsyth, A., Funk, S., Denny, M., Roach, K. E., Boone, D., & The Hemophilia Treatment Center Network. (2011). Range of motion measurements: Reference values and a database for comparison studies. *Haemophilia*, 17(3), 500-507. <https://doi.org/10.1111/j.1365-2516.2010.02399.x>
- Stadt Zürich. (2022). *Zürich 4D*. <https://www.stadt-zuerich.ch/hbd/de/index/staedtebau/zuerich-4d.html>
- swisstopo. (2022). *GeoAdmin API*. <https://api3.geo.admin.ch/>
- Thöny, M., Schnürer, R., Sieber, R., Hurni, L., & Pajarola, R. (2018). Storytelling in Interactive 3D Geographic Visualization Systems. *ISPRS International Journal of Geo-Information*, 7(3), 123. <https://doi.org/10.3390/ijgi7030123>
- Varol, G., Ceylan, D., Russell, B., Yang, J., Yumer, E., Laptev, I., & Schmid, C. (2018). BodyNet: Volumetric Inference of 3D Human Body Shapes. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), *Computer Vision - ECCV 2018* (pp. 20-38). Springer International Publishing. https://doi.org/10.1007/978-3-030-01234-2_2
- Wang, W., Xu, Q., Ceylan, D., Mech, R., & Neumann, U. (2019). DISN: Deep Implicit Surface Network for High-Quality Single-View 3D Reconstruction. *NIPS'19: Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Article 45, 492-502. <https://dl.acm.org/doi/10.5555/3454287.3454332>
- Wang, X., & Yu, J. (2020). Learning to Cartoonize Using White-Box Cartoon Representations. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8087-8096. <https://doi.org/10.1109/CVPR42600.2020.00811>
- Wu, S., Heitzler, M., & Hurni, L. (2022). Leveraging uncertainty estimation and spatial pyramid pooling for extracting hydrological features from scanned historical topographic maps. *GIScience & Remote Sensing*, 59(1), 200-214. <https://doi.org/10.1080/15481603.2021.2023840>
- Yao, P., Fang, Z., Wu, F., Feng, Y., & Li, J. (2019). *DenseBody: Directly Regressing Dense 3D Human Pose and Shape From a Single Color Image*. arXiv. <https://doi.org/10.48550/arXiv.1903.10153>
- Ye, Y., Tulsiani, S., & Gupta, A. (2021). Shelf-Supervised Mesh Prediction in the Wild. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8839-8848. <https://doi.org/10.1109/CVPR46437.2021.00873>
- Yu, A., Ye, V., Tancik, M., & Kanazawa, A. (2021). pixelNeRF: Neural Radiance Fields from One or Few Images. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4576-4585. <https://doi.org/10.1109/CVPR46437.2021.00455>
- Zhou, Y., Liu, S., & Ma, Y. (2021). NeRD: Neural 3D Reflection Symmetry Detector. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 15935-15944. <https://doi.org/10.1109/CVPR46437.2021.01568>

6. Conclusion

In this dissertation, several artificial neural networks are applied to fulfil cartographic tasks, such as map identification, object detection, and construction of 3D objects. It is shown that these labour-intensive tasks for humans can be automatised while achieving considerable results and leaving room for further consolidation. A focus is put on transforming static 2D figures from pictorial maps into animatable 3D figures for interactive cartographic applications. By this, novel storytelling concepts can be introduced and existing approaches be extended to increase the attractivity and usability of digital 3D maps.

Three research questions, specified in Chapter 1.2, are examined in this dissertation.

RQ1a: Identification of maps and pictorial maps

More than 6000 images from a social media website are collected for the task of map recognition. Using two CNNs for classification, maps are distinguished from images at an accuracy of 95-97%, while pictorial maps are separated from other maps at an accuracy of 87-92%.

For these classification tasks, a definition of maps is established considering the latest developments in digital cartography and a definition of pictorial maps is formulated based on the level of representation of the depicted objects. The definitions contribute to a common understanding of maps and help to delimit them from similar visual artefacts. The collected training and test data required for this distinction can serve as a baseline for further improving the machine learning models. Since the input resolution of images is currently quite limited due to the high working memory demands of the models on the GPU, different input strategies (e.g. resizing, cropping) are examined in detail.

RQ1b: Detection of pictorial objects including parts and key points on maps

About 500 maps from digital libraries containing illustrations of sailing ships are annotated representatively for recognizing pictorial objects. Additionally, more than 4000 humans from real-world photos and synthetically generated human figures are taken as training data to extract their silhouettes, body parts, and pose points. Bounding boxes of sailing ships are detected in pictorial maps at an average precision of 32-36% by two CNNs for object detection. Silhouettes of human figures are identified in pictorial maps at an average precision of 17-19% by a CNN for instance segmentation, while body parts and pose points of the figures are recognised simultaneously at an average precision of 9-11% by an adapted CNN for semantic segmentation.

It is demonstrated that the choice of hyperparameters (e.g. anchor scales, feature map sizes) has a decisive impact on the accuracy of the networks for object detection. The average precision of networks detecting objects and segmenting instances was about 33-41% at that time (He et al., 2017; T.-Y. Lin et al., 2020). Since annotating new datasets for supervised learning is very tedious, especially for instance segmentation, similar object representations are considered for training the networks. Using merely

synthetically generated data yields to be less effective for segmentation tasks, possibly due to a too little variety of styles. In turn, real-world datasets are sometimes unbalanced, for instance, most of them include upright humans, thus uncommon poses are not handled well by the networks. A combination of abstract and realistic representations is beneficial for recognizing pictorial objects, nonetheless, the identification of crowded and very small objects needs to be further improved. The simultaneous prediction of pose points and body parts contributes towards designing networks which are capable of solving multiple tasks at the same time.

RQ2: Derivation of pictorial 3D objects from the segmented 2D objects

3D meshes of a male and female human are assigned more than 3000 poses and are varied in height and weight for generating a set of pictorial 3D figures. Their poses, shapes, and textures are plausibly inferred from 2D human figures by a series of ANNs. The ANNs are specialised in 3D pose point estimation, single-view 3D reconstruction, UV coordinate prediction, texture inpainting, and facial enhancement.

It is observed that assuming either an orthographic or perspective projection has only a minimal impact on creating 3D figures from their 2D complements. Coarse body part shapes are reconstructed well by a network, whereas fine-grained structures, such as fingers, need further attention in the future. The addition of pose points improves inferring 3D body parts from binary masks, likewise, body part masks help to predict UV maps from depth maps. Training an ANN with cartoonised head images amends noisy and too realistic outputs of the figure’s faces. A fully functional and understandable pipeline consisting of different models is established, though its maintainability may be challenging and synergy effects are not fully exploited.

RQ3: Rendering of the inferred pictorial 3D objects in real time

The 3D models of the pictorial figures are represented by implicit surfaces and are rendered with sphere tracing in 256x256px images at 25 frames per second, which enables basic interactivity and animations.

SDFs are chosen as a mathematical model for implicit surfaces, but not functional representations, so-called ‘F-Reps’ (Pasko et al., 2001), which are less frequently used. As outputting SDF values on-the-fly for given coordinates by an ANN does not result in a real-time performance, SDF values are pre-estimated and stored in a 3D grid. For rendering, the sphere tracing algorithm is implemented in the Python programming language, mainly to ease debugging. Features like a trilinear interpolation of SDF values or texture blending are included in the renderer to enhance the quality of the 3D models.

Research question	Article I	Article II	Article III
<i>RQ1a</i>	✓		
<i>RQ1b</i>	✓	✓	
<i>RQ2</i>			✓
<i>RQ3</i>			✓

Table 6.1: Correspondence of research questions and research articles

The correspondence of research questions and research articles is given in Table 6.1.

Individual training datasets are prepared for all research articles; however, some datasets are reused in subsequent articles. Maps without any pictorial figures from *Article I* serve as background maps, on which additional entities are inserted, for training a network in *Article II*. Photos of real persons' faces from *Article II* are cartoonised to resynthesise the faces of pictorial humans in *Article III*. Some maps including larger pictorial human figures from *Article I* are selected as test data for *Article III*.

ANNs, most of them CNNs, containing residual connections have been studied throughout the dissertation. In *Article I*, residual connections enclose the Inception or Xception modules of the classification networks. The backbone networks described in *Articles I* and *II* also contain these connections. Three of the four head networks examined in *Article II* for estimating poses and segmenting body parts include an ascending number of residual connections. In *Article III*, networks with these connections are used for predicting 3D and texture coordinates as well as inpainting textures.

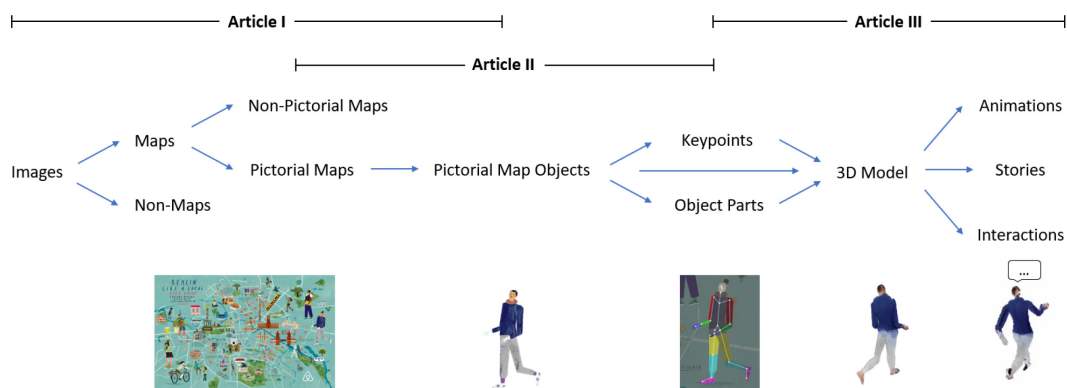


Figure 6.1: Workflow pursued in the research articles

By training the ANNs, animatable 3D figures are automatically derived from static 2D figures using the example of pictorial humans (Figure 6.1). Human figures appear often on pictorial maps; thus, these illustrative maps are distinguished from other maps and images in *Article I*. In *Article II*, silhouettes of the figures are identified as well as their joints and body parts, which is required for skeletal animation. The joints and body parts are transferred into the 3D space, and textures of the figures are completed for hidden views in *Article III*.

Finally, the constructed figures can be rendered in a real-time 3D environment like a virtual globe (Figure 6.2). The rendering area can be restricted to billboards to avoid performing the sphere tracing algorithm on the entire screen. Camera parameters, signed distance values, textures, model position and orientation are passed to the fragment shader of the billboard. Animations can be defined programmatically in the fragment shader or transformation parameters derived from motion-captured real humans can be passed additionally. In the latter case, disoriented body parts can occur since the internal rotation of the bones is not considered yet. The transformations to animate the figures need to be applied also to the normals for correct texturing. Further billboards positioned near the head of the figures can contain texts or input elements for interactive storytelling.

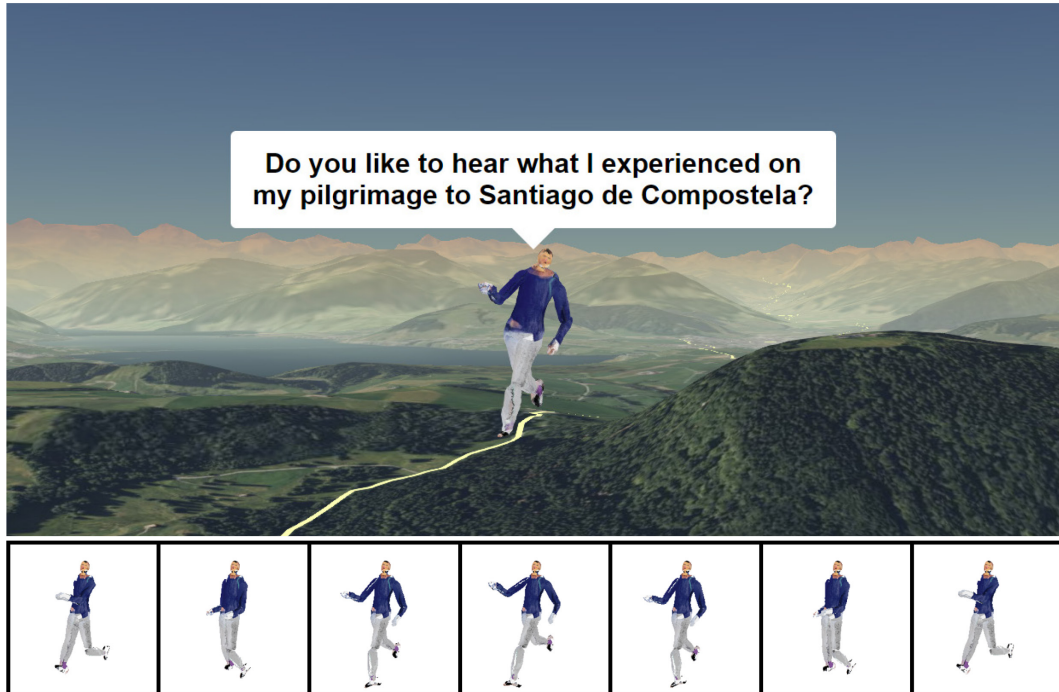


Figure 6.2: An inferred human figure rendered via sphere tracing in a virtual globe (here: CesiumJS). The figure is equipped with a speech bubble for storytelling and a walking animation based on the sine function. Selected key frames of the animation sequence are depicted below.

Note

Data, code, and models of all research articles are publicly available at <http://narrat3d.ethz.ch> so that other researchers can reproduce and improve the results.

7. Outlook

7.1. Technology

At present, cartographic storytelling is a type of visual storytelling, which is often combined with other visuals like images, videos, or charts. In the future, it is thinkable that physical *output devices* appeal also to other senses, such as acoustics, haptics, and olfactics (Figure 7.1). This trend can be observed in so-called 4DX or 5D cinemas, where various effects like motions, water, or scents are triggered to support actions happening in the movie. However, it is not known yet whether multisensory devices will be affordable or practicable for individual use. Nevertheless, it is already nowadays possible to emit sounds via speakers or let game controllers vibrate.

Haptic devices like mice, keyboards, and touchpads are currently prevalent *input devices* for digital cartographic applications. These peripherals allow users to interact with the storytellers by asking questions, giving answers, or issuing commands. Reactions of the storytellers may be derived automatically by scripts or by artificial intelligence (e.g. ChatGPT), or provided manually by other users. It can be assumed that some of the research prototypes examining different modalities like eye-tracking (e.g. Kwok et al., 2019), gesture recognition (e.g. Berger, 2021), or speech recognition (e.g. Shan & Sun, 2023) turn soon into marketable products. These devices are particularly interesting for the cartoverse, where avatars are controlled by users (see Chapter 5.7). In the far future, other haptic devices or even neural interfaces can be expected (T.-H. Yang et al., 2021).

Besides supporting to process inputs and to produce outputs of human-computer interfaces, *machine learning* will enhance discriminative and generative cartographic tasks on the methodological side. Ideally, the analysed content of digitised maps (e.g. pictorial maps) can be transferred into a data model, which can be spatially, temporally, and thematically queried. Knowledge graphs (Hogan et al., 2021) are a promising candidate for this kind of data structure, where machine learning additionally helps to deduce relationships of entities. Concerning generative tasks, an uprising concept is neural rendering (Tewari et al., 2022), where images based on given parameters are produced by machine learning models, which may improve the performance and quality of graphics and animations in maps. Beyond, machine learning may accelerate and enhance the extraction of thematic data (e.g. from websites), the production of maps (e.g. placement of labels), and the development of user interfaces (e.g. based on user analytics).

Traditional algorithms in computer science will be also continuously improved. Performance gains of sphere tracing can be expected with newer generations of GPUs (i.e. NVIDIA RTX), which are optimised for *ray tracing*. Next to pictorial human figures, also topographic objects may be represented by implicit surfaces in 3D. This may ease implementing interactions between these objects, for example showing the footsteps of the figures on the terrain. Beyond, light effects produced by ray tracing will contribute towards creating suitable atmospheres for cartographic storytelling.

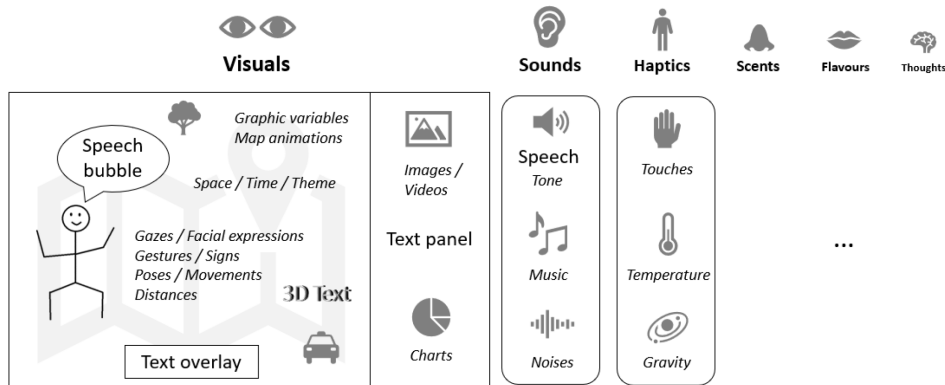


Figure 7.1: Cartographic storytelling as a multisensory experience (bold font) including different types of verbal (regular font) and non-verbal (italic font) communication

7.2. Concepts

Augmenting maps with pictorial objects will allow cartographers to introduce verbal and nonverbal means of visual communication (Figure 7.1). An example of *verbal communication* is to tell stories via text in speech bubbles. This will enable transferring different narrative perspectives from literature: Figures can be subjective first-person narrators like protagonists (e.g. famous personalities), who tell personal stories, or supportive characters (e.g. sewers in textile production), who give background knowledge to a topic. Otherwise, a rather objective third-person narrator (e.g. a domain expert) can highlight the peculiarities of a thematic map. By presenting information and stories explicitly, users do not need to deduce them merely from the map. Some meanings may still be disguised or hidden, which need to be implied by ‘reading between the lines’.

The map reader’s interpretations are an inherent part of *nonverbal communication*. For example, by placing a pictorial object into the map space, the underlying topographic object can be emphasised (e.g. a cow on a meadow). Moreover, nonverbal communication comprises movements, also known as kinesics, which can be used to guide the user through the map (e.g. by follow-me gestures) or to support a thematic map (e.g. complex movements of an athlete). The virtual object (e.g. a train) can reflect the movements of a real-world counterpart in real time. Alternatively, animation paths can be scripted by cartographers to indicate the likelihood that an entity (e.g. a bird) is present in an area, which can be possibly linked with uncertainty visualisations. Nonverbal communication can influence verbal communication. For instance, based on the distance to the pictorial object, also known as proxemics, and conditional on the time, also known as chronemics, different facts or secrets can be revealed (e.g. when a storyteller is within close distance and after some time passed).

The design of story maps including pictorial objects is connected to intrinsic and extrinsic storytelling approaches (see Chapter 2.1.4). Animated interactive objects will complement *intrinsic* cartographic storytelling (Figure 7.2a) because they are an integral part of the map and can be the target of events (e.g. clicks). Ideally, the style of the additional objects corresponds to the existing map elements since inconsistencies may be perceived as not visually pleasing. Individual objects will be already sufficiently

representative, but also crowd visualisations are thinkable (e.g. people dancing at a music festival). Concerning generalisation, details need to be possibly reduced when viewing the 3D objects from a larger distance.

In *extrinsic* cartographic storytelling (Figure 7.2b), events emitted in the user interface may affect animated interactive objects on the map. Objects or their properties may be varied depending on the storyline or time, spatial navigation tools, and thematic filters. In the user interface, metadata (e.g. name, age, gender), special functions and items, and animation controls (e.g. play, pause, fast-forward, loop) may be shown once an object is selected on the map. For an overview, a legend with all objects may be provided, possibly allowing users to focus on an object inside the map.

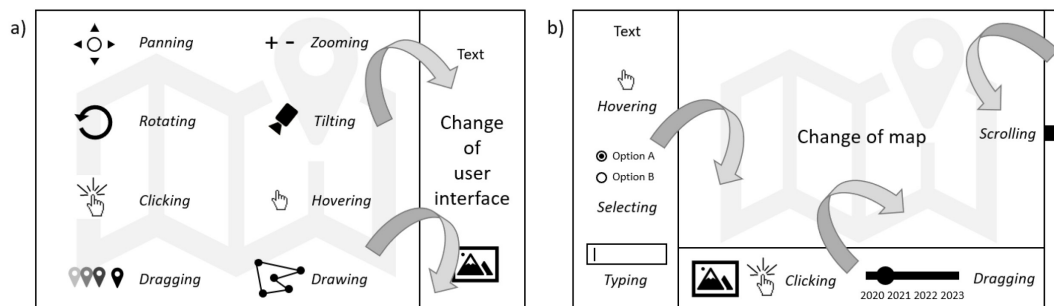


Figure 7.2: Comparison between intrinsic (a) and extrinsic (b) cartographic storytelling. The story unfolds by spatially navigating and interacting inside the map with intrinsic cartographic storytelling, while the story is driven by events dispatched in the user interface with extrinsic cartographic storytelling. Bi-directional story flows occur when both methods are combined (Sieber et al., 2021).

7.3. Usability

Maps, especially topographic maps, are mainly used for *orientation* and *navigation* purposes. Animated pictorial 3D objects do not seem to be very suited to assist fast travelling users, like drivers in cars or pilots in planes, who need to decide quickly and who need to be focused more on the environment than on the map. However, figures can be potentially added to cartographic AR applications for slowly travelling users, like hikers in the mountains or tourists in cities. For example, virtual animals may appear if the real ones are hidden or a virtual tour guide may give hints about special places. A positive side-effect triggered by these map supplements may be a higher activity rate of users, similar to the popular AR game 'Pokémon Go'. Nevertheless, there is a danger of accidents in outdoor environments due to distraction.

Another purpose of maps is *planning*, for example, constructing or restoring buildings or quarters. Viewers of these maps are already familiar with static 3D figures and objects included for decorative aspects. Nowadays, a trend can be observed towards digital twins of cities (Schrotter & Hürzeler, 2020), which are interactively experienceable, show the prospective changes in high fidelity, and allow performing complex analyses (e.g. shadow calculations, noise propagation). Animated figures may be inserted there for accessibility and risk simulations, which make sense rather on large-scale maps than on

small-scale maps. In terms of public participation, figures may represent involved actors expressing different opinions.

Maps are an integral part of *education* in geography. Since computers and tablets are increasingly available in schools, animated interactive objects may be added to digital maps to generate positive emotions for students so that anxieties about the complexity of data may become secondary. It is not clear yet whether maps with pictorial objects offer an added value compared to maps without them, thus certain metrics (e.g. effectiveness, efficiency, memorability) would need to be obtained while assessing the geographic competencies of students. Other use cases include virtual field trips or virtual teaching, bearing in mind that a full virtualisation of learning activities is not equivalent to on-site teaching, which became evident by the COVID-19 pandemic.

Maps are an effective *communication* means to present spatial patterns, flows, and relationships to the general public in atlases or museums. Animated interactive objects may raise the interest of people to explore the maps and may give a better understanding of the often abstractly encoded information in maps. Cartographic guidelines would need to be derived from usability studies, for example, how many objects should be animated at which speed to not impair the user's cognition. The additional objects seem to be suited for presenting environmental or societal problems (e.g. sustainable development goals formulated by the United Nations) but may be extraneous for exploration and analysis tasks.

Entertainment is an aspect of maps, which does not need to be neglected. Animated objects may illustrate stories about geographic and historic curiosities, potentially enhanced with effects from social media sites. As an interactive layer, gamification elements (e.g. quizzes, puzzles) may be included, which motivate and challenge the users. In particular, VR applications within the scope of the metaverse offer a large potential in a cartographic depiction of the real world, for example, live chats with other people represented by avatars (Park & Kim, 2022). In this context, the protection of children against common internet threats and the danger of reality loss deserves special attention, though the latter issue is perhaps less apparent in the cartoverse than in other digital twins due to the higher level of abstraction.

Note

This chapter outlined prospective technological, conceptual, and usability aspects, which may arise from adding animated interactive 3D figures to story maps. The structure of the chapter has been aligned to a medium-centred model of communication (Elleström, 2018) with a digital story map as the medium (Chapter 7.1), cartographers as producers (Chapter 7.2), and map users as perceivers (Chapter 7.3). Since all parts are interrelated, overlaps may occur.

Acknowledgements

Many people accompanied and supported me during this chapter in my life:

First and foremost, I would like to thank my supervisor, *Prof. Lorenz Hurni*, for giving me the opportunity to elaborate and conduct this doctoral project. The stable working conditions and pleasant atmosphere at our institute provided the basis to investigate the subject matter in-depth.

Secondly, my thank goes to all collaborators of this research project, starting with my advisor, *Dr. René Sieber*, who let me build the scientific foundations for this project during my time at the Atlas of Switzerland. Besides the doctoral project, we frequently bespoke other topics, may it be ICA commissions or SOLA competitions. Subsequently, I like to appreciate my co-supervisor, *Prof. Cengiz Öztireli*, for his encouragement, his technical guidance and his benevolent interest in this project. Similarly, my co-advisor, *Prof. Arzu Çöltekin*, is to be highlighted for her warm-hearted suggestions and her patience in finalizing an article on 3D charts together. *Dr. Jost Schmid-Lanter* contributed historical facts about maps to the first article of this dissertation and we exchanged ideas concerning the St. Gallen Globe. *Dr. Magnus Heitzler* devoted his knowledge in machine learning to the second and third research article. Apart from this, we explored the world of trading stocks and cryptocurrencies. For the background chapters of this dissertation, *Dr. Barbara Piatti* gave valuable inputs regarding storytelling and *Prof. Renato Pajarola* enhanced various properties related to computer graphics. Additional to my supervisors and collaborators, *Dr. Cristina Iosifescu Enescu*, *Dr. Olga Koblet*, and *Dr. Matthias Thöny* helped to revise the grant proposal for this project.

Next, I would like to express my gratitude to my colleagues from the Institute of Cartography and Geoinformation at ETH Zurich. This includes my fellow doctoral students, *Marianna Farmakis-Serebryakova*, *Chenjing Jiao*, *Sidi Wu*, *Xue Xia*, and *Katharina Henggeler*, for our doctoral seminars, scientific discussions and leisure activities. I'm thankful to all people from the Atlas of Switzerland, especially *Michael Schmuki* and *Raphael Vomsattel* for our work on the atlas and our awesome snowshoe hikes, and from the Swiss World Atlas, especially *Wenke Zimmermann* for mastering the COVID-19 pandemic and *Patrick Lehmann* for his design hints for this document. My neighbour two doors down, *Stefan Räber*, and I fathomed all kinds of linguistic and website-related problems. The IT teams from ETH, our department and our institute, represented by *Claudia Matthys*, provided superb infrastructure and technical support. Our secretary, *Natalie Ammann Baumgartner*, organised smoothly all conference trips and social events. Together with our teaching coordinator, *Dr. Christian Häberling*, and our manager of continuing education, *Sabine Wöhlbier Röthlisberger*, we handled successfully administrative issues during the pandemic. *Dr. Mattia Ryffel*, *Pascal Tschudi*, *Cédric Dind*, *Stefan Schalcher*, and I pursued a nice side project on augmented reality. A big thank you to all Geomatics students for being kind and motivated in our courses throughout the years. Greetings go to all my former colleagues, whom I met during this period, as well as the geoinformation engineering group chaired by *Prof. Martin Raubal*. Not to forget is all the administrative, facility management and gastronomical staff who makes ETH Zurich fully operational every day.

Acknowledgements

Furthermore, I met many interesting people during a summer school on 3D animation in Nottingham. I participated in a productive workshop on drafting an atlas proposal in Leipzig, hosted by *Prof. Francis Harvey* and *Eric Losang*. At conferences and other gatherings, I met members of the Swiss Society of Cartography and members of ICA commissions, like *Dr. Angeliki Tsorlini* and *Prof. Vít Voženílek*. To be mentioned is also *Prof. Dirk Burghardt*, who motivated me at the cartographic conferences all over the world.

Finally, I like to thank my parents, my brother, my grandparents and other close relatives for their continuous backing, grounding and making me feel at home every time I come to Berlin. During my visits there, I also frequently had fun with my friends from school, *Benjamin*, *Maria*, and *Christian*. Warm regards to my connections near Munich, like *Eric* for our birthday emails and *Lea* for visiting cities. Credited near Zurich are *Alžběta* for hiking, *Nadia* for quizzing, *Raluca* for recommending music, and *Roland* for sharing management course experiences. Thank you to my neighbour, *Ernst*, for the weekly supply with a news magazine, and to my other neighbour, *Katherine*, for taking care of my plants when I was at conferences or on holidays.

The cover image has been generated with the aid of the Stable Diffusion model *Artium*.

This work was supported by an *ETH Zurich Research Grant*.

Curriculum vitae

The doctoral student's curriculum vitae is included in the printed version only.

Publications

- Schnürer, R. (2013). Sensor Discovery in Virtual Globes for Citizen Scientists. *Geoinformatik 2013*.
<http://geoinformatik2013.de/index.php/en/papers/application-track>
- Schnürer, R., Dind, C., Schalcher, S., Tschudi, P., & Hurni, L. (2020). Augmenting Printed School Atlases with Thematic 3D Maps. *Multimodal Technologies and Interaction*, 4(2), Article 23.
<https://doi.org/10.3390/mti4020023>
- Schnürer, R., Eichenberger, R., & Sieber, R. (2014). Creating Styles, Legends, and Charts for 3D Maps. A Mashup of D3.js, osgEarth, and the Chromium Embedded Framework (Poster). *AutoCarto 2014*.
https://www.researchgate.net/publication/335014536_Creating_Styles_Legends_and_Charts_for_3D_Maps_A_Mashup_of_D3js_osgEarth_and_the_Chromium_Embedded_Framework
- Schnürer, R., Eichenberger, R., Sieber, R., & Hurni, L. (2015). 3D Charts – Taxonomy and Implementation in a Virtual Globe. *Revista Brasileira de Cartografia*, 67(5). <https://doi.org/10.14393/rbcv67n5-44627>
- Schnürer, R., Eichenberger, R., Sieber, R., & Hurni, L. (2017). Animations for 3D Solid Charts in a Virtual Globe – Techniques, Use cases, and Implementation. *28th International Cartographic Conference (ICC)*.
https://www.researchgate.net/publication/335014544_Animations_for_3D_Solid_Charts_in_a_Virtual_Globe_-_Techniques_Use_cases_and_Implementation
- Schnürer, R., Ritzi, M., Çöltekin, A., & Sieber, R. (2020). An empirical evaluation of three-dimensional pie charts with individually extruded sectors in a geovisualization context. *Information Visualization*, 19(3), 183-206. <https://doi.org/10.1177/1473871619896103>
- Schnürer, R., & Sieber, R. (2012). Assessment of standards-compliant Web Mapping Tools for Atlas Creation. *Proceedings of the 5th All-Ukrainian Scientific and Practical Conference on National Mapping*.
<https://www.research-collection.ethz.ch/handle/20.500.11850/62341>
- Schnürer, R., Sieber, R., & Çöltekin, A. (2015). The Next Generation of Atlas User Interfaces: A User Study with “Digital Natives”. In J. Brus, A. Vondrakova, & V. Vozenilek (Eds.), *Modern Trends in Cartography* (pp. 23-36). Springer International Publishing. https://doi.org/10.1007/978-3-319-07926-4_3
- Sieber, R., & Schnürer, R. (2016). Atlas der Schweiz – Fit für die Zukunft. *Geomatik Schweiz*, 4, 111-114.
- Sieber, R., Schnürer, R., Eichenberger, R., & Hurni, L. (2015). Designing Graphical User Interfaces for 3D Atlases. *Proceedings of the 27th International Cartographic Conference (ICC)*.
https://icaci.org/files/documents/ICC_proceedings/ICC2015/papers/20/421.html
- Sieber, R., Serebryakova, M., Schnürer, R., & Hurni, L. (2016). Atlas of Switzerland Goes Online and 3D–Concept, Architecture and Visualization Methods. In G. Gartner, M. Jobst, & H. Huang (Eds.), *Progress in Cartography* (pp. 171-184). Springer International Publishing.
https://doi.org/10.1007/978-3-319-19602-2_11
- Thöny, M., Schnürer, R., Sieber, R., Hurni, L., & Pajarola, R. (2018). Storytelling in Interactive 3D Geographic Visualization Systems. *ISPRS International Journal of Geo-Information*, 7(3), 123.
<https://doi.org/10.3390/ijgi7030123>
- Wu, S., Schnürer, R., Heitzler, M., & Hurni, L. (2022). Unsupervised historical map registration by a deformation neural network. *Proceedings of the 5th ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery*, 76-81. <https://doi.org/10.1145/3557918.3565871>

References

- Abgottspon, A. (2011). *Procedural modelling in Houdini based on Function Representation* [Master thesis]. Bournemouth University.
- Alexis, K., Kaffes, V., & Giannopoulos, G. (2020). Boosting toponym interlinking by paying attention to both machine and deep learning. *Proceedings of the Sixth International ACM SIGMOD Workshop on Managing and Mining Enriched Geo-Spatial Data*, 1-5. <https://doi.org/10.1145/3403896.3403970>
- Amiraghdam, A., Diehl, A., & Pajarola, R. (2020). LOCALIS: Locally-adaptive Line Simplification for GPU-based Geographic Vector Data Visualization. *Computer Graphics Forum*, 39(3), 443-453. <https://doi.org/10.1111/cgf.13993>
- Amiraghdam, A., Diehl, A., & Pajarola, R. (2022). LOOPS: LOcally Optimized Polygon Simplification. *Computer Graphics Forum*, 41(3), 355-365. <https://doi.org/10.1111/cgf.14546>
- Antoniou, V., Ragia, L., Nomikou, P., Bardouli, P., Lampridou, D., Ioannou, T., Kalisperakis, I., & Stentoumis, C. (2018). Creating a Story Map Using Geographic Information Systems to Explore Geomorphology and History of Methana Peninsula. *ISPRS International Journal of Geo-Information*, 7(12), Article 12. <https://doi.org/10.3390/ijgi7120484>
- Arundel, S., Morgan, T. P., & Thiem, P. T. (2022). Deep learning detection and recognition of spot elevations on historic topographic maps. *Frontiers in Environmental Science*, 10, 1-10. <https://doi.org/10.3389/fenvs.2022.804155>
- Balme, C. B. (2005). Freytag, Gustav. In *The Oxford Encyclopedia of Theatre and Performance*. Oxford University Press. <https://www.oxfordreference.com/display/10.1093/acref/9780198601746.001.0001/acref-9780198601746-e-1406>
- Barreda Luna, A. A., Kuri, G. H., Rodríguez-Reséndiz, J., Zamora Antuñano, M. A., Altamirano Corro, J. A., & Paredes-García, W. J. (2022). Public space accessibility and machine learning tools for street vending spatial categorization. *Journal of Maps*, 18(1), 43-52. <https://doi.org/10.1080/17445647.2022.2035836>
- Berger, M. (2021). Exploring and Transforming Spaces Through High-Dimensional Gestural Interactions. *Advances in Cartography and GIScience of the ICA*, 3, 1-8. <https://doi.org/10.5194/ica-adv-3-2-2021>
- Beyerer, J., Richter, M., & Nagel, M. (2017). *Pattern Recognition* (1st edition). De Gruyter Oldenbourg.
- Blaeu, J. (1648). *Nova Totius Terrarum Orbis Tabula* [Map]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Joan_Blaeu_-_Map_of_the_World_1648.jpg
- Bleisch, S. (2011). *Evaluating the appropriateness of visually combining quantitative data representations with 3D desktop virtual environments using mixed methods* [Dissertation]. City University London.
- Bogunov, K., & Istomin, S. (2023). Designing a 3D Application Based on Digital Models of Railway Infrastructure. In A. Guda (Ed.), *Networked Control Systems for Connected and Automated Vehicles* (pp. 419-428). Springer International Publishing. https://doi.org/10.1007/978-3-031-11051-1_41
- Bonassi, N., & Sieber, R. (2017). Story Telling in Atlases—The intrinsic way. *28th International Cartographic Conference (ICC)*. <https://www.research-collection.ethz.ch/handle/20.500.11850/225474>
- Borkin, M. A., Vo, A. A., Bylinskii, Z., Isola, P., Sunkavalli, S., Oliva, A., & Pfister, H. (2013). What Makes a Visualization Memorable? *IEEE Transactions on Visualization and Computer Graphics*, 19(12), 2306-2315. <https://doi.org/10.1109/TVCG.2013.234>
- Bösch, J., Goswami, P., & Pajarola, R. (2009). RASrER: Simple and efficient terrain rendering on the GPU. *EUROGRAPHICS 2009*, 35-42. <https://doi.org/10.5167/uzh-29729>
- Buchholz, H. (2006). *Real-time visualization of 3D city models* [Dissertation]. Universität Potsdam.

References

- Buschmann, S., Trapp, M., & Döllner, J. (2014). Real-Time Animated Visualization of Massive Air-Traffic Trajectories. *2014 International Conference on Cyberworlds*, 174-181. <https://doi.org/10.1109/CW.2014.32>
- Cannon, K. (2013). *The Appendix Guide to ...* [Map]. <https://theappendix.net/>
- Caquard, S. (2011). Cartography I: Mapping narrative cartography. *Progress in Human Geography*, 37(1), 135-144. <https://doi.org/10.1177/0309132511423796>
- Caquard, S., & Dimitrovias, S. (2017). Story Maps & Co. The state of the art of online narrative cartography. *Mappemonde. Revue Trimestrielle Sur l'image Géographique et Les Formes Du Territoire*, 121, Article 121. <https://doi.org/10.4000/mappemonde.3386>
- Caquard, S., & Fiset, J.-P. (2014). How can we map stories? A cybercartographic application for narrative cartography. *Journal of Maps*, 10(1), 18-25. <https://doi.org/10.1080/17445647.2013.847387>
- Caquard, S., Pyne, S., Igloliorte, H., Mierins, K., Hayes, A., & Taylor, D. R. F. (2009). A "Living" Atlas for Geospatial Storytelling: The Cybercartographic Atlas of Indigenous Perspectives and Knowledge of the Great Lakes Region. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 44(2), 83-100. <https://doi.org/10.3138/carto.44.2.83>
- Carter, B., & Rinner, C. (2014). Locally weighted linear combination in a vector geographic information system. *Journal of Geographical Systems*, 16(3), 343-361. <https://doi.org/10.1007/s10109-013-0194-3>
- Cartwright, W. (2009). Applying the Theatre Metaphor to Integrated Media for Depicting Geography. *The Cartographic Journal*, 46(1), 24-35. <https://doi.org/10.1179/000870409X415561>
- Cartwright, W. (2004). Using the web for focussed geographical storytelling via gameplay. *Proceedings of the First International Joint Workshop on Ubiquitous, Pervasive and Internet Mapping*. UPIMap2004, Tokyo.
- Cartwright, W., & Field, K. (2015). Exploring cartographic storytelling. Reflections on mapping real-life and fictional stories. *Proceedings of the 27th International Cartographic Conference (ICC)*. https://icaci.org/files/documents/ICC_proceedings/ICC2015/papers/10/209.html
- Cartwright, W., & Peterson, M. P. (2007). Multimedia Cartography. In W. Cartwright, M. P. Peterson, & G. Gartner (Eds.), *Multimedia Cartography* (pp. 1-10). Springer. https://doi.org/10.1007/978-3-540-36651-5_1
- Chase, E. D., Diego, J., Jacobson, E. G., Vogelear, W. E., & Walker, A. (1943). *Story map of ...* [Map]. Colortext Publications, Inc.; David Rumsey Map Collection. <https://www.davidrumsey.com/>
- Chen, Y., Carlinet, E., Chazalon, J., Mallet, C., Duménieu, B., & Perret, J. (2021). Vectorization of Historical Maps Using Deep Edge Filtering and Closed Shape Extraction. In J. Lladós, D. Lopresti, & S. Uchida (Eds.), *Document Analysis and Recognition - ICDAR 2021* (pp. 510-525). Springer International Publishing. https://doi.org/10.1007/978-3-030-86337-1_34
- Cheng, R., & Chen, J. (2021). A location conversion method for roads through deep learning-based semantic matching and simplified qualitative direction knowledge representation. *Engineering Applications of Artificial Intelligence*, 104, 104400. <https://doi.org/10.1016/j.engappai.2021.104400>
- Chiang, Y.-Y., Duan, W., Leyk, S., Uhl, J. H., & Knoblock, C. A. (2020). Training Deep Learning Models for Geographic Feature Recognition from Historical Maps. In Y.-Y. Chiang, W. Duan, S. Leyk, J. H. Uhl, & C. A. Knoblock (Eds.), *Using Historical Maps in Scientific Studies: Applications, Challenges, and Best Practices* (pp. 65-98). Springer International Publishing. https://doi.org/10.1007/978-3-319-66908-3_4
- Christen, M. (2008). The Future of Virtual Globes-The Interactive Ray-Traced Digital Earth. *Proceedings of the XXI ISPRS Congress*, 969-974. <https://www.isprs.org/proceedings/XXXVII/congress/tc2.aspx>
- Christen, M., Nebiker, S., & Loesch, B. (2012). Web-Based Large-Scale 3D-Geovisualisation Using WebGL: The OpenWebGlobe Project. *International Journal of 3-D Information Modeling (IJ3DIM)*, 1(3), 16-25. <https://doi.org/10.4018/ij3dim.2012070102>

- Christophe, S., Mermet, S., Laurent, M., & Touya, G. (2022). Neural map style transfer exploration with GANs. *International Journal of Cartography*, 8(1), 18–36. <https://doi.org/10.1080/23729333.2022.2031554>
- Clarke, K. (2016, June 4). Maps Mania: Inside Asia. *Maps Mania*. <https://googlemapsmania.blogspot.com/2016/06/inside-asia.html>
- Clinton, C. (2022). *Colortext Maps of the 1930s* [Map]. <https://www.oldimprints.com/collecting-colortext-maps-of-the-1930s.php>
- Comber, S., & Arribas-Bel, D. (2019). Machine learning innovations in address matching: A practical comparison of word2vec and CRFs. *Transactions in GIS*, 23(2), 334–348. <https://doi.org/10.1111/tgis.12522>
- Cosgrove, D. (2005). Maps, Mapping, Modernity: Art and Cartography in the Twentieth Century. *Imago Mundi*, 57(1), 35–54. <https://doi.org/10.1080/0308569042000289824>
- Courtial, A., El Ayedi, A., Touya, G., & Zhang, X. (2020). Exploring the Potential of Deep Learning Segmentation for Mountain Roads Generalisation. *ISPRS International Journal of Geo-Information*, 9(5), Article 5. <https://doi.org/10.3390/ijgi9050338>
- Courtial, A., Touya, G., & Zhang, X. (2021). Generative adversarial networks to generalise urban areas in topographic maps. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2021, 15–22. <https://doi.org/10.5194/isprs-archives-XLIII-B4-2021-15-2021>
- Cozzi, P., & Ring, K. (2011). *3D Engine Design for Virtual Globes* (1st edition). A K Peters/CRC Press.
- Cozzi, P., & Stoner, F. (2010). GPU ray casting of virtual globes. *ACM SIGGRAPH 2010 Posters*, Article 128. <https://doi.org/10.1145/1836845.1836982>
- Croci, J. A., Amiraghdam, A., & Pajarola, R. (2022). Terrender: A Web-Based Multi-Resolution Terrain Rendering Framework. *Proceedings of the 27th International Conference on 3D Web Technology*, 1–11. <https://doi.org/10.1145/3564533.3564567>
- Cron, J. (2006). *Graphische Benutzeroberflächen interaktiver Atlanten* [Diploma thesis]. Hochschule für Technik und Wirtschaft Dresden.
- Cui, G., Bian, W., & Wang, X. (2021). Hidden Markov map matching based on trajectory segmentation with heading homogeneity. *Geoinformatica*, 25(1), 179–206. <https://doi.org/10.1007/s10707-020-00429-4>
- Denil, M. (2017). Storied Maps. *Cartographic Perspectives*, 84, 5–22. <https://doi.org/10.14714/CP84.1374>
- Dickmann, F., Keil, J., Dickmann, P. L., & Edler, D. (2021). The Impact of Augmented Reality Techniques on Cartographic Visualization. *KN - Journal of Cartography and Geographic Information*, 71(4), 285–295. <https://doi.org/10.1007/s42489-021-00091-2>
- Diego, J. (1935). *The story map of Spain* [Map]. David Rumsey Map Collection. <https://www.davidrumsey.com/luna/servlet/detail/RUMSEY~8~1~274016~90047777:The-story-map-of-Spain>
- Dobesova, Z. (2020). Experiment in Finding Look-Alike European Cities Using Urban Atlas Data. *ISPRS International Journal of Geo-Information*, 9(6), Article 6. <https://doi.org/10.3390/ijgi9060406>
- Döllner, J., & Hinrichs, K. (2000). Dynamic 3D maps and their texture-based design. *Proceedings Computer Graphics International 2000*, 325–334. <https://doi.org/10.1109/CGI.2000.852349>
- Döllner, J., & Kersting, O. (2000). Dynamic 3D Maps As Visual Interfaces for Spatio-temporal Data. *Proceedings of the 8th ACM International Symposium on Advances in Geographic Information Systems*, 115–120. <https://doi.org/10.1145/355274.355291>
- Dransch, D. (1997). *Computer-Animation in der Kartographie—Theorie und Praxis*. Springer.
- Du J., & Wu F. (2022). An ensemble learning simplification approach based on multiple machine-learning algorithms with the fusion using of raster and vector data and a use case of coastline simplification.

References

- Acta Geodaetica et Cartographica Sinica*, 51(3), 373-387.
<https://doi.org/10.11947/j.AGCS.2022.20210135>
- Du, J., Wu, F., Xing, R., Gong, X., & Yu, L. (2022). Segmentation and sampling method for complex polyline generalization based on a generative adversarial network. *Geocarto International*, 37(14), 4158-4180.
<https://doi.org/10.1080/10106049.2021.1878288>
- Duan, W., Chiang, Y.-Y., Leyk, S., Uhl, J. H., & Knoblock, C. A. (2020). Automatic alignment of contemporary vector data and georeferenced historical maps using reinforcement learning. *International Journal of Geographical Information Science*, 34(4), 824-849. <https://doi.org/10.1080/13658816.2019.1698742>
- Duan, W., Chiang, Y.-Y., Leyk, S., Uhl, J. H., & Knoblock, C. A. (2021). A Label Correction Algorithm Using Prior Information for Automatic and Accurate Geospatial Object Recognition. *2021 IEEE International Conference on Big Data*, 1604-1610. <https://doi.org/10.1109/BigData52589.2021.9671657>
- Eccles, R., Kapler, T., Harper, R., & Wright, W. (2008). Stories in GeoTime. *Information Visualization*, 7(1), 3-17.
<https://doi.org/10.1057/palgrave.ivs.9500173>
- Eidler, D., Keil, J., WiedenlÜbbert, T., Sossna, M., Kühne, O., & Dickmann, F. (2019). Immersive VR Experience of Redeveloped Post-industrial Sites: The Example of "Zeche Holland" in Bochum-Wattenscheid. *KN - Journal of Cartography and Geographic Information*, 69(4), 267-284.
<https://doi.org/10.1007/s42489-019-00030-2>
- Ekman, S. (2013). *Here Be Dragons: Exploring Fantasy Maps and Settings* (Illustrated Edition). Wesleyan University Press.
- Elgandy, M. (2020). *Deep Learning for Vision Systems* (1st edition). Manning.
- Eligüznel, N., Çetinkaya, C., & Dereli, T. (2020). Comparison of different machine learning techniques on location extraction by utilizing geo-tagged tweets: A case study. *Advanced Engineering Informatics*, 46, 101151. <https://doi.org/10.1016/j.aei.2020.101151>
- Elleström, L. (2018). A medium-centered model of communication. *Semiotica*, 2018(224), 269-293.
<https://doi.org/10.1515/sem-2016-0024>
- ESRI. (2012). *Telling Stories with Maps (White Paper)*.
<http://storymaps.esri.com/downloads/Telling%20Stories%20with%20Maps.pdf>
- Fang, Z., Qi, J., Fan, L., Huang, J., Jin, Y., & Yang, T. (2022). A topography-aware approach to the automatic generation of urban road networks. *International Journal of Geographical Information Science*, 36(10), 2035-2059. <https://doi.org/10.1080/13658816.2022.2072849>
- Farmakis-Serebryakova, M., Heitzler, M., & Hurni, L. (2022). Terrain Segmentation Using a U-Net for Improved Relief Shading. *ISPRS International Journal of Geo-Information*, 11(7), Article 7.
<https://doi.org/10.3390/ijgi11070395>
- Faroqi, H., Mesbah, M., & Kim, J. (2020). Modelling socioeconomic attributes of public transit passengers. *Journal of Geographical Systems*, 22(4), 519-543. <https://doi.org/10.1007/s10109-020-00328-0>
- Feeney, A. E. (2017). Beer-trail maps and the growth of experiential tourism. *Cartographic Perspectives*, 87, 9-28. <https://doi.org/10.14714/CP87.1383>
- Feng, J., Li, Y., Zhao, K., Xu, Z., Xia, T., Zhang, J., & Jin, D. (2022). DeepMM: Deep Learning Based Map Matching With Data Augmentation. *IEEE Transactions on Mobile Computing*, 21(7), 2372-2384.
<https://doi.org/10.1109/TMC.2020.3043500>
- Fish, C. (2020). Storytelling for Making Cartographic Design Decisions for Climate Change Communication in the United States. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 55(2), 69-84. <https://doi.org/10.3138/cart-2019-0019>
- Fryazinov, O., & Pasko, A. (2008). Interactive ray shading of FRep objects. *WSCG '2008: Communication Papers. The 16th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision in Co-Operation with EUROGRAPHICS*, 145-152.
<http://hdl.handle.net/11025/11115>

- Fuhrmann, S., Schmidt, B., Berlin, K., & Kuhn, W. (2001). Anforderungen an 3D-Interaktionen in geo-virtuellen Visualisierungsumgebungen. *Kartographische Nachrichten*, 51(4), 191-195. <https://doi.org/10.1007/BF03544822>
- Fujita, H., & Arikawa, M. (2011). A User Study of a Map-Based Slideshow Editor. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 46(2), 74-82. <https://doi.org/10.3138/carto.46.2.74>
- Gaitán, R., Ten, M., Lluch, J., & Sevilla, L. (2006). Geoviewer3D: 3D Geographical Information Viewing. In P. Brunet, N. Correia, & G. Baranoski (Eds.), *Ibero-American Symposium on Computer Graphics (SIACG)* (pp. 1-4). <http://www.upv.es/upl/U0433582.pdf>
- García-Molsosa, A., Orengo, H. A., Lawrence, D., Philip, G., Hopper, K., & Petrie, C. A. (2021). Potential of deep learning segmentation for the extraction of archaeological features from historical map series. *Archaeological Prospection*, 28(2), 187-199. <https://doi.org/10.1002/arp.1807>
- Gavriil, K., Muntingh, G., & Barrowclough, O. J. D. (2019). Void Filling of Digital Elevation Models With Deep Generative Models. *IEEE Geoscience and Remote Sensing Letters*, 16(10), 1645-1649. <https://doi.org/10.1109/LGRS.2019.2902222>
- Gede, M. (2015). Novel Globe Publishing Techniques Using WebGL. *E-Perimetron*, 10(2), 7. http://www.e-perimetron.org/Vol10_2.htm
- Gede, M., & Jeney, J. (2017). Thematische Kartierung mit Verwendung von Cesium. *KN - Journal of Cartography and Geographic Information*, 67(4), 210-213. <https://doi.org/10.1007/BF03544604>
- Germanchis, T., Cartwright, W., & Pettit, C. (2007). Virtual Queenscliff: A Computer Game Approach for Depicting Geography. In W. Cartwright, M. P. Peterson, & G. Gartner (Eds.), *Multimedia Cartography* (pp. 359-368). Springer. https://doi.org/10.1007/978-3-540-36651-5_25
- Germanchis, T., Pettit, C., & Cartwright, W. (2004). Building a three-dimensional geospatial virtual environment on computer gaming technology. *Journal of Spatial Science*, 49(1), 89-95. <https://doi.org/10.1080/14498596.2004.9635008>
- Glander, T., & Döllner, J. (2009). Abstract representations for interactive visualization of virtual 3D city models. *Computers, Environment and Urban Systems*, 33(5), 375-387. <https://doi.org/10.1016/j.compenvurbsys.2009.07.003>
- Glassner, A. (2021). *Deep Learning: A Visual Approach* (Illustrated edition). No Starch Press.
- Golze, J., Zourlidou, S., & Sester, M. (2020). Traffic Regulator Detection Using GPS Trajectories. *KN - Journal of Cartography and Geographic Information*, 70(3), 95-105. <https://doi.org/10.1007/s42489-020-00048-x>
- Góralski, R. (2009). *Three-dimensional interactive maps. Theory and practice* [Dissertation]. University of South Wales.
- Gore, A. (1998). The Digital Earth. *Australian Surveyor*, 43(2), 89-91. <https://doi.org/10.1080/00050348.1998.10558728>
- Griffin, D. (2017). Beautiful Geography: The Pictorial Maps of Ruth Taylor White. *Imago Mundi*, 69(2), 233-247. <https://doi.org/10.1080/03085694.2017.1312117>
- Gullu, M., & Narin, O. G. (2019). Georeferencing of the Nile River in Piri Reis 1521 map, Using Artificial Neural Network Method. *Acta Geodaetica et Geophysica*, 54(3), 387-401. <https://doi.org/10.1007/s40328-019-00255-7>
- Guo, D., Ge, S., Zhang, S., Gao, S., Tao, R., & Wang, Y. (2022). DeepSSN: A deep convolutional neural network to assess spatial scene similarity. *Transactions in GIS*, 26(4), 1914-1938. <https://doi.org/10.1111/tgis.12915>
- Haines, E., Hoffman, N., & Akenine-Möller, T. (2018). *Real-Time Rendering* (4th edition). A K Peters/CRC Press.

References

- Hardisty, F., MacEachren, A., & Takatsuka, M. (2001). Cartographic Animation in Three Dimensions: Experimenting with the Scene Graph. *Proceedings of the 20th International Cartographic Conference (ICC)*. https://icaci.org/files/documents/ICC_proceedings/ICC2001/icc2001/topic17.htm
- Harrie, L., Oucheikh, R., Nilsson, Å., Oxenstierna, A., Cederholm, P., Wei, L., Richter, K.-F., & Olsson, P. (2022). Label Placement Challenges in City Wayfinding Map Production—Identification and Possible Solutions. *Journal of Geovisualization and Spatial Analysis*, 6(1), Article 16. <https://doi.org/10.1007/s41651-022-00115-z>
- Harvey, F. (2009). More than Names—Digital Earth and/or Virtual Globes? *International Journal of Spatial Data Infrastructures Research*, 4, 111-116. <https://doi.org/10.2902/1725-0463.2009.04.art6>
- Henrik. (2008). *This diagram illustrates the ray tracing algorithm for rendering an image*. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Ray_trace_diagram.svg
- Herman, L., Russnak, J., Stuchlík, R., & Hladík, J. (2018). Visualization of Traffic Offences in the City of Brno (Czech Republic): Achieving 3D Thematic Cartography through Open Source and Open Data. *Proceedings of 25th Central European Conference*. Useful Geography: Transfer from Research to Practice. <https://doi.org/10.5817/CZ.MUNI.P210-8908-2018>
- Hobona, G., James, P., & Fairbairn, D. (2006). Web-based visualization of 3D geospatial data using Java3D. *IEEE Computer Graphics and Applications*, 26(4), 28-33. <https://doi.org/10.1109/MCG.2006.94>
- Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., Melo, G. D., Gutierrez, C., Kirrane, S., Gayo, J. E. L., Navigli, R., Neumaier, S., Ngomo, A.-C. N., Polleres, A., Rashid, S. M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., & Zimmermann, A. (2021). Knowledge Graphs. *ACM Computing Surveys*, 54(4), 71:1-71:37. <https://doi.org/10.1145/3447772>
- Hu, Y., Gui, Z., Wang, J., & Li, M. (2022). Enriching the metadata of map images: A deep learning approach with GIS-based data augmentation. *International Journal of Geographical Information Science*, 36(4), 799-821. <https://doi.org/10.1080/13658816.2021.1968407>
- Indans, R., Hauthal, E., & Burghardt, D. (2019). Towards an Audio-Locative Mobile Application for Immersive Storytelling. *KN - Journal of Cartography and Geographic Information*, 69(1), 41-50. <https://doi.org/10.1007/s42489-019-00007-1>
- Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017). Image-to-Image Translation with Conditional Adversarial Networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5967-5976. <https://doi.org/10.1109/CVPR.2017.632>
- Jenny, H., Jenny, B., Cartwright, W. E., & Hurni, L. (2011). Interactive Local Terrain Deformation Inspired by Hand-painted Panoramas. *The Cartographic Journal*, 48(1), 11-20. <https://doi.org/10.1179/1743277411Y.0000000002>
- Jenny, H., Jenny, B., & Hurni, L. (2010). Interactive Design of 3D Maps with Progressive Projection. *The Cartographic Journal*, 47(3), 211-221. <https://doi.org/10.1179/000870410X12786821061495>
- Jin, Z., Kim, J., Yeo, H., & Choi, S. (2022). Transformer-based map-matching model with limited labeled data using transfer-learning approach. *Transportation Research Part C: Emerging Technologies*, 140, 103668. <https://doi.org/10.1016/j.trc.2022.103668>
- Kada, M., Wichmann, A., & Hermes, T. (2015). Smooth transformations between generalized 3D building models for visualization purposes. *Cartography and Geographic Information Science*, 42(4), 306-314. <https://doi.org/10.1080/15230406.2015.1039588>
- Kanas, N. (2019). Terrestrial and celestial pictorial maps. In N. Kanas (Ed.), *Star Maps: History, Artistry, and Cartography* (pp. 422-458). Springer International Publishing. https://doi.org/10.1007/978-3-030-13613-0_11
- Karsznia, I., & Weibel, R. (2018). Improving settlement selection for small-scale maps using data enrichment and machine learning. *Cartography and Geographic Information Science*, 45(2), 111-127. <https://doi.org/10.1080/15230406.2016.1274237>
- Keil, J., Edler, D., Schmitt, T., & Dickmann, F. (2021). Creating Immersive Virtual Environments Based on Open Geospatial Data and Game Engines. *KN - Journal of Cartography and Geographic Information*, 71(1), 53-65. <https://doi.org/10.1007/s42489-020-00069-6>

- Kennelly, P., & League, C. (2015). Modified Helix Structures for Visualizing Maximum Daily Temperature Data. *Proceedings of the 27th International Cartographic Conference (ICC)*.
https://icaci.org/files/documents/ICC_proceedings/ICC2015/papers/31/244.html
- Kolbe, T. H. (2009). Representing and Exchanging 3D City Models with CityGML. In J. Lee & S. Zlatanova (Eds.), *3D Geo-Information Sciences* (pp. 15-31). Springer. https://doi.org/10.1007/978-3-540-87395-2_2
- Koutsabasis, P., Partheniadis, K., Gardeli, A., Vogiatzidakis, P., Nikolakopoulou, V., Chatzigrigoriou, P., Vosinakis, S., & Filippidou, D. E. (2022). Co-Designing the User Experience of Location-Based Games for a Network of Museums: Involving Cultural Heritage Professionals and Local Communities. *Multimodal Technologies and Interaction*, 6(5), Article 5. <https://doi.org/10.3390/mti6050036>
- Kraak, M.-J. (1994). Interactive Modelling Environment for Three-dimensional Maps: Functionality and Interface Issues. In A. M. MacEachren & D. R. F. Taylor (Eds.), *Modern Cartography Series* (Vol. 2, pp. 269-285). Academic Press. <https://doi.org/10.1016/B978-0-08-042415-6.50021-1>
- Kraak, M.-J., & Kveladze, I. (2017). Narrative of the annotated Space-Time Cube - revisiting a historical event. *Journal of Maps*, 13(1), 56-61. <https://doi.org/10.1080/17445647.2017.1323034>
- Krisp, J., & Fronzek, S. (2003). Visualising thematical spatial data by using the third dimension. In K. Virrantaus & H. Tveite (Eds.), *Proceedings of the 9th Scandinavian Research Conference on Geographical Information Science (ScanGIS)* (pp. 157-166).
- Kwok, T. C. K., Kiefer, P., Schinazi, V. R., Adams, B., & Raubal, M. (2019). Gaze-Guided Narratives: Adapting Audio Guide Content to Gaze in Virtual and Real Environments. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 1-12.
<https://doi.org/10.1145/3290605.3300721>
- Laksono, D., & Aditya, T. (2019). Utilizing A Game Engine for Interactive 3D Topographic Data Visualization. *ISPRS International Journal of Geo-Information*, 8(8), 361. <https://doi.org/10.3390/ijgi8080361>
- Lan, T., Li, Z., Wang, J., Gong, C., & Ti, P. (2022). An ANNs-Based Method for Automated Labelling of Schematic Metro Maps. *ISPRS International Journal of Geo-Information*, 11(1), 36.
<https://doi.org/10.3390/ijgi11010036>
- Lehmann, C., & Döllner, J. (2014). Silhouette-Based Label Placement in Interactive 3D Maps. In M. Buchroithner, N. Prechtel, & D. Burghardt (Eds.), *Cartography from Pole to Pole: Selected Contributions to the XXVth International Conference of the ICA, Dresden 2013* (pp. 177-186). Springer. https://doi.org/10.1007/978-3-642-32618-9_13
- Lewis-Jones, H. (2018). *The Writer's Map: An Atlas of Imaginary Lands* (1st ed.). Thames & Hudson.
- Li, J. (2022). *Computational Cartographic Recognition: Exploring the Use of Machine Learning and Other Computational Approaches to Map Reading* [Dissertation]. The Ohio State University.
- Li, S., Yin, G., Ma, J., Wen, B., & Zhou, Z. (2022). Generation Method for Shaded Relief Based on Conditional Generative Adversarial Nets. *ISPRS International Journal of Geo-Information*, 11(7), 374.
<https://doi.org/10.3390/ijgi11070374>
- Li, Y., Lu, X., Yan, H., Wang, W., & Li, P. (2022). A Skeleton-Line-Based Graph Convolutional Neural Network for Areal Settlements' Shape Classification. *Applied Sciences*, 12(19), 10001.
<https://doi.org/10.3390/app121910001>
- Li, Y., Sakamoto, M., Shinohara, T., & Satoh, T. (2020). Automatic label placement of area-features using deep learning. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLIII-B4-2020*, 117-122. <https://doi.org/10.5194/isprs-archives-XLIII-B4-2020-117-2020>
- Limberger, D., Pursche, M., Klimke, J., & Döllner, J. (2017). Progressive high-quality rendering for interactive information cartography using WebGL. *Proceedings of the 22nd International Conference on 3D Web Technology*, Article 8. <https://doi.org/10.1145/3055624.3075951>
- Lin, Y., Kang, M., Wu, Y., Du, Q., & Liu, T. (2020). A deep learning architecture for semantic address matching. *International Journal of Geographical Information Science*, 34(3), 559-576.
<https://doi.org/10.1080/13658816.2019.1681431>

References

- Lindstrom, P., Koller, D., Ribarsky, W., Hodges, L. F., Faust, N., & Turner, G. A. (1996). Real-time, continuous level of detail rendering of height fields. *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, 109-118. <https://doi.org/10.1145/237170.237217>
- Liu, Z., Liu, J., Xu, X., & Wu, K. (2022). DeepGPS: Deep Learning Enhanced GPS Positioning in Urban Canyons. *IEEE Transactions on Mobile Computing*, 1-15. <https://doi.org/10.1109/TMC.2022.3208240>
- Lorenz, H., & Döllner, J. (2008). Dynamic Mesh Refinement on GPU using Geometry Shaders. *WSCG '2008: Full Papers: The 16th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision in Co-Operation with EUROGRAPHICS*, 97-104. <http://hdl.handle.net/11025/10924>
- Lorenz, H., & Döllner, J. (2010). 3D feature surface properties and their application in geovisualization. *Computers, Environment and Urban Systems*, 34(6), 476-483. <https://doi.org/10.1016/j.compenvurbsys.2010.04.003>
- Lorenz, H., Trapp, M., Döllner, J., & Jobst, M. (2008). Interactive Multi-Perspective Views of Virtual 3D Landscape and City Models. In L. Bernard, A. Friis-Christensen, & H. Pundt (Eds.), *The European Information Society: Taking Geoinformation Science One Step Further* (pp. 301-321). Springer. https://doi.org/10.1007/978-3-540-78946-8_16
- Lu, M., & Arikawa, M. (2013). Map-Based Storytelling Tool for Real-World Walking Tour. In J. M. Krisp (Ed.), *Progress in Location-Based Services* (pp. 435-451). Springer. https://doi.org/10.1007/978-3-642-34203-5_24
- Lütjens, M., Kersten, T. P., Dorschel, B., & Tschirschwitz, F. (2019). Virtual Reality in Cartography: Immersive 3D Visualization of the Arctic Clyde Inlet (Canada) Using Digital Elevation Models and Bathymetric Data. *Multimodal Technologies and Interaction*, 3(1), Article 9. <https://doi.org/10.3390/mti3010009>
- Ma, Y., & Tang, J. (2021). *Deep Learning on Graphs* (1st edition). Cambridge University Press.
- Maass, S., & Döllner, J. (2008). Seamless integration of labels into interactive virtual 3D environments using parameterized hulls. *Proceedings of the Fourth Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, 33-40. <https://doi.org/10.2312/COMPAESTH/COMPAESTH08/033-040>
- Mai, G., Huang, W., Cai, L., Zhu, R., & Lao, N. (2022). Narrative Cartography with Knowledge Graphs. *Journal of Geovisualization and Spatial Analysis*, 6(1), 4. <https://doi.org/10.1007/s41651-021-00097-4>
- Mao, B., Ban, Y., & Laumert, B. (2020). Dynamic Online 3D Visualization Framework for Real-Time Energy Simulation Based on 3D Tiles. *ISPRS International Journal of Geo-Information*, 9(3), Article 166. <https://doi.org/10.3390/ijgi9030166>
- Mao, X., Chow, J. K., Su, Z., Wang, Y.-H., Li, J., Wu, T., & Li, T. (2021). Deep learning-enhanced extraction of drainage networks from digital elevation models. *Environmental Modelling & Software*, 144, 105135. <https://doi.org/10.1016/j.envsoft.2021.105135>
- Marta, M., & Osso, P. (2015). Story Maps at school: Teaching and learning stories with maps. *Journal of Research and Didactics in Geography*, 2, 61-68. <https://doi.org/10.4458/6063-05>
- Mather, P., & Tso, B. (2009). *Classification Methods for Remotely Sensed Data* (2nd edition). CRC Press.
- Maxwell, A. E., Bester, M. S., Guillen, L. A., Ramezan, C. A., Carpinello, D. J., Fan, Y., Hartley, F. M., Maynard, S. M., & Pyron, J. L. (2020). Semantic Segmentation Deep Learning for Extracting Surface Mine Extents from Historic Topographic Maps. *Remote Sensing*, 12(24), Article 4145. <https://doi.org/10.3390/rs12244145>
- Mayr, E., & Windhager, F. (2018). Once upon a Spacetime: Visual Storytelling in Cognitive and Geotemporal Information Spaces. *ISPRS International Journal of Geo-Information*, 7(3), Article 96. <https://doi.org/10.3390/ijgi7030096>
- McCann, M. P. (2004). Using GeoVRML for 3D oceanographic data visualizations. *Proceedings of the Ninth International Conference on 3D Web Technology*, 15-21. <https://doi.org/10.1145/985040.985043>

- Medyńska-Gulij, B., Forrest, D., & Cybulski, P. (2021). Modern Cartographic Forms of Expression: The Renaissance of Multimedia Cartography. *ISPRS International Journal of Geo-Information*, 10(7), Article 484. <https://doi.org/10.3390/ijgi10070484>
- Meng, L., & Forberg, A. (2007). 3D Building Generalisation. In W. A. Mackaness, A. Ruas, & L. T. Sarjakoski (Eds.), *Generalisation of Geographic Information* (pp. 211–231). Elsevier Ltd. <https://doi.org/10.1016/B978-008045374-3/50013-2>
- Mocnik, F.-B., & Fairbairn, D. (2018). Maps Telling Stories? *The Cartographic Journal*, 55(1), 36–57. <https://doi.org/10.1080/00087041.2017.1304498>
- Moellering, H. (1980). The Real-Time Animation of Three-Dimensional Maps. *The American Cartographer*, 7(1), 67–75. <https://doi.org/10.1559/152304080784522892>
- Mohammadi, N., & Sedaghat, A. (2021). A framework for classification of volunteered geographic data based on user's need. *Geocarto International*, 36(11), 1276–1291. <https://doi.org/10.1080/10106049.2019.1641562>
- Moore, A. B., Nowostawski, M., Frantz, C., & Hulbe, C. (2018). Comic Strip Narratives in Time Geography. *ISPRS International Journal of Geo-Information*, 7(7), Article 245. <https://doi.org/10.3390/ijgi7070245>
- Moore, K. (1999). VRML and Java for Interactive 3D Cartography. In W. Cartwright, M. P. Peterson, & G. Gartner (Eds.), *Multimedia Cartography* (pp. 205–216). Springer. https://doi.org/10.1007/978-3-662-03784-3_20
- Mościcka, A., & Kuźma, M. (2018). Spatio-Temporal Database of Places Located in the Border Area. *ISPRS International Journal of Geo-Information*, 7(3), Article 108. <https://doi.org/10.3390/ijgi7030108>
- Mościcka, A., & Zwirowicz-Rutkowska, A. (2018). On the Use of Geographic Information in Humanities Research Infrastructure: A Case Study on Cultural Heritage. *ISPRS International Journal of Geo-Information*, 7(3), Article 106. <https://doi.org/10.3390/ijgi7030106>
- Muehlenhaus, I. (2014). Looking at the Big Picture: Adapting Film Theory to Examine Map Form, Meaning, and Aesthetic. *Cartographic Perspectives*, 77, 46–66. <https://doi.org/10.14714/CP77.1239>
- Nebiker, S. (2003). Support for Visualisation and Animation in a Scalable 3D GIS Environment – Motivation, Concepts and Implementation. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXIV-5/W10. <https://www.isprs.org/proceedings/XXXIV/5-W10/>
- Netten, D. (2020). The New World Map and the Old; The Moving Narrative of Joan Blaeu's Nova Totius Terrarum Orbis Tabula (1648). In B. Vannieuwenhuyze & Z. Segal (Eds.), *Motion in Maps, Maps in Motion: Mapping Stories and Movement through Time* (pp. 33–56). Amsterdam University Press. <https://doi.org/10.1017/9789048542956.002>
- Nielsen, M. A. (2015). *Neural Networks and Deep Learning*. Determination Press. <http://neuralnetworksanddeeplearning.com>
- OED Online. (2022b). *Narrative, n.; plot, n.; story, n.; storyline, n.; storytelling, n.* Oxford University Press. <https://www.oed.com/>
- Oh, B.-W. (2020). Map Detection using Deep Learning. *Journal of Advanced Information Technology and Convergence*, 10(2), 61–72. <https://doi.org/10.14801/JAITS.2020.10.2.61>
- Oleggini, L., Nova, S., & Hurni, L. (2009). 3D Gaming and Cartography—Design Considerations for game-based generation of Virtual Terrain Environments. *Proceedings of the 24th International Cartographic Conference (ICC)*. https://icaci.org/files/documents/ICC_proceedings/ICC2009/html/nonref/20.html
- Olmedo, É., & Caquard, S. (2022). Mapping the Skin and the Guts of Stories - A Dialogue between Geolocated and Dislocated Cartographies. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 57(2), 127–146. <https://doi.org/10.3138/cart-2021-0006>
- Pajarola, R., & Gobbetti, E. (2007). Survey of semi-regular multiresolution models for interactive terrain rendering. *The Visual Computer*, 23(8), 583–605. <https://doi.org/10.1007/s00371-007-0163-2>

References

- Pal, S., & Sarda, R. (2022). Modeling riparian flood plain wetland water richness in pursuance of damming and linking it with a methane emission rate. *Geocarto International*, 37(25), 7954–7982. <https://doi.org/10.1080/10106049.2021.1988726>
- Pasko, A., Adzhiev, V., Schmitt, B., & Schlick, C. (2001). Constructive Hypervolume Modeling. *Graphical Models*, 63(6), 413–442. <https://doi.org/10.1006/gmod.2001.0560>
- Persson, D., Gartner, D. G., & Buchroithner, D. M. (2006). Towards a Typology of Interactivity Functions for Visual Map Exploration. In D. E. Stefanakis, D. M. P. Peterson, D. C. Armenakis, & D. V. Delis (Eds.), *Geographic Hypermedia* (pp. 275–292). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-34238-0_15
- Petitpierre, R., Kaplan, F., & Lenardo, I. (2021). Generic Semantic Segmentation of Historical Maps. *CEUR Workshop Proceedings*, 2989(21), 228–248. <https://ceur-ws.org/Vol-2989/>
- Potié, Q., Touya, G., Beladraoui, C., El-Moutaouakkil, A., & Mackaness, W. A. (2022). Deep learning for anchor detection in multi-scale maps. In G. Gartner, A. Binn, & O. Ignateva (Eds.), *EuroCarto 2022* (Vol. 5, p. 82). Copernicus Publications. <https://doi.org/10.5194/ica-abs-5-82-2022>
- Prestby, T. (2022). Design Techniques for COVID-19 Story Maps: A Quantitative Content Analysis. *Cartography and Geographic Information Science*, Advance online publication. <https://doi.org/10.1080/15230406.2022.2102077>
- Price, M. E. (1937). Adventures through Maps. *Childhood Education*, 13(5), 206–210. <https://doi.org/10.1080/00094056.1937.10724077>
- Qian, C., Yi, C., Cheng, C., Pu, G., & Liu, J. (2020). A Coarse-to-Fine Model for Geolocating Chinese Addresses. *ISPRS International Journal of Geo-Information*, 9(12), Article 698. <https://doi.org/10.3390/ijgi9120698>
- Qu, H., Wang, H., Cui, W., Wu, Y., & Chan, M.-Y. (2009). Focus+Context Route Zooming and Information Overlay in 3D Urban Environments. *IEEE Transactions on Visualization and Computer Graphics*, 15(6), 1547–1554. <https://doi.org/10.1109/TVCG.2009.144>
- Quilez, I. (2015). *Arlo*. Shadertoy. <https://www.shadertoy.com/view/4dtGWM>
- Ran, W., Wang, J., Yang, K., Bai, L., Rao, X., Zhao, Z., & Xu, C. (2022). Raster Map Line Element Extraction Method Based on Improved U-Net Network. *ISPRS International Journal of Geo-Information*, 11(8), Article 439. <https://doi.org/10.3390/ijgi11080439>
- Ray, C., Góralski, R., Claramunt, C., & Gold, C. (2011). Real-Time 3D Monitoring of Marine Navigation. In V. V. Popovich, C. Claramunt, T. Devogele, M. Schrenk, & K. Korolenko (Eds.), *Information Fusion and Geographic Information Systems: Towards the Digital Ocean* (pp. 161–175). Springer. https://doi.org/10.1007/978-3-642-19766-6_14
- Reumont, F., Arsanjani, J. J., & Riedl, A. (2013). Visualization of geologic geospatial datasets through X3D in the frame of WebGIS. *International Journal of Digital Earth*, 6(5), 483–503. <https://doi.org/10.1080/17538947.2011.627471>
- Reuschel, A.-K., & Hurni, L. (2011). Mapping Literature: Visualisation of Spatial Uncertainty in Fiction. *The Cartographic Journal*, 48(4), 293–308. <https://doi.org/10.1179/1743277411Y.0000000023>
- Richard, D. (2000). Development of an internet atlas of Switzerland. *Computers & Geosciences*, 26(1), 45–50. [https://doi.org/10.1016/S0098-3004\(99\)00032-1](https://doi.org/10.1016/S0098-3004(99)00032-1)
- Riedl, A. (2000). *Virtuelle Globen in der Geovisualisierung*. Institut für Geographie und Regionalforschung der Universität Wien, Kartographie und Geoinformation.
- Rocca, L. (2013). I Geoblog: Strumenti per una ‘Cartografia Aumentata’. In EUT Edizioni Università di Trieste (Ed.), *Bollettino dell'Associazione Italiana di Cartografia* (Vol. 147, pp. 17–39). <http://hdl.handle.net/10077/11607>
- Röhlig, M., Luboschik, M., & Schumann, H. (2017). Visibility widgets for unveiling occluded data in 3D terrain visualization. *Journal of Visual Languages & Computing*, 42, 86–98. <https://doi.org/10.1016/j.jvlc.2017.08.008>

- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65, 386–408. <https://doi.org/10.1037/h0042519>
- Roth, R. E. (2012). Cartographic Interaction Primitives: Framework and Synthesis. *The Cartographic Journal*, 49(4), 376–395. <https://doi.org/10.1179/1743277412Y.0000000019>
- Roth, R. E. (2021). Cartographic Design as Visual Storytelling: Synthesis and Review of Map-Based Narratives, Genres, and Tropes. *The Cartographic Journal*, 58(1), 83–114. <https://doi.org/10.1080/00087041.2019.1633103>
- Ruan, S., Long, C., Bao, J., Li, C., Yu, Z., Li, R., Liang, Y., He, T., & Zheng, Y. (2020). Learning to Generate Maps from Trajectories. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01), 890–897. <https://doi.org/10.1609/aaai.v34i01.5435>
- Russell, S. J., & Norvig, P. (2016). *Artificial Intelligence: A Modern Approach. Global Edition* (3rd edition). PEV.
- Saha, A., Pal, S. C., Chowdhuri, I., Islam, A. R. Md. T., Chakraborty, R., & Roy, P. (2022). Application of neural network model-based framework approach to identify gully erosion potential hotspot zones in sub-tropical environment. *Geocarto International*, 37(26), 14758–14784. <https://doi.org/10.1080/10106049.2022.2091042>
- Sandvik, B. (2008). *Using KML for thematic mapping* [Master thesis]. University of Edinburgh.
- Santos, R., Murrieta-Flores, P., Calado, P., & Martins, B. (2018). Toponym matching through deep neural networks. *International Journal of Geographical Information Science*, 32(2), 324–348. <https://doi.org/10.1080/13658816.2017.1390119>
- Satriadi, K. A., Ens, B., Czauderna, T., Cordeil, M., & Jenny, B. (2021). Quantitative Data Visualisation on Virtual Globes. *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, Article 460. <https://doi.org/10.1145/3411764.3445152>
- Schneider, M., Guthe, M., & Klein, R. (2005). Real-time Rendering of Complex Vector Data on 3D Terrain Models. In H. Thwaites (Ed.), *Proceedings of the 11th International Conference on Virtual Systems and Multimedia* (pp. 573–582). <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=543551ed0e58f6badff0e4a6ff2020e419a51fbc>
- Schnürer, R., Dind, C., Schalcher, S., Tschudi, P., & Hurni, L. (2020). Augmenting Printed School Atlases with Thematic 3D Maps. *Multimodal Technologies and Interaction*, 4(2), Article 23. <https://doi.org/10.3390/mti4020023>
- Schnürer, R., Eichenberger, R., & Sieber, R. (2014). Creating Styles, Legends, and Charts for 3D Maps. A Mashup of D3.js, osgEarth, and the Chromium Embedded Framework (Poster). *AutoCarto 2014*. https://www.researchgate.net/publication/335014536_Creating_Styles_Legends_and_Charts_for_3D_Maps_A_Mashup_of_D3js_osgEarth_and_the_Chromium_Embedded_Framework
- Schnürer, R., Eichenberger, R., Sieber, R., & Hurni, L. (2015). 3D Charts – Taxonomy and Implementation in a Virtual Globe. *Revista Brasileira de Cartografia*, 67(5). <https://doi.org/10.14393/rbcv67n5-44627>
- Semmo, A., & Döllner, J. (2015). Interactive image filtering for level-of-abstraction texturing of virtual 3D scenes. *Computers & Graphics*, 52, 181–198. <https://doi.org/10.1016/j.cag.2015.02.001>
- Semmo, A., Trapp, M., Kyprianidis, J. E., & Döllner, J. (2012). Interactive Visualization of Generalized Virtual 3D City Models using Level-of-Abstraction Transitions. *Computer Graphics Forum*, 31(3), 885–894. <https://doi.org/10.1111/j.1467-8659.2012.03081.x>
- Shan, P., & Sun, W. (2023). Research on application of 3D GIS in urban landscape based on speech recognition system. *Soft Computing*, Advance online publication. <https://doi.org/10.1007/s00500-023-08714-8>
- She, J., Li, X., Liu, J., Chen, Y., Tan, J., & Wu, G. (2019). A building label placement method for 3D visualizations based on candidate label evaluation and selection. *International Journal of Geographical Information Science*, 33(10), 2033–2054. <https://doi.org/10.1080/13658816.2019.1606431>

References

- She, J., Liu, J., Tan, J., Dong, J., & Biao, W. (2020). Local terrain modification method considering physical feature constraints for vector elements. *Cartography and Geographic Information Science*, 47(5), 452-470. <https://doi.org/10.1080/15230406.2020.1770128>
- Shen, D. Y., Takara, K., Tachikawa, Y., & Liu, Y. L. (2006). 3D simulation of soft geo-objects. *International Journal of Geographical Information Science*, 20(3), 261-271. <https://doi.org/10.1080/13658810500287149>
- Shenghua, X., Jiping, L., Yong, W., Fuhao, Z., & Rongshuang, F. (2008). Visualization of 3D Moving Geographic Objects Based on Terrain Matching. *2008 International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, 605-610. <https://doi.org/10.1109/ISSNIP.2008.4762056>
- Sherlock, M. J., Hasan, M., & Samavati, F. F. (2021). Interactive data styling and multifocal visualization for a multigrid web-based Digital Earth. *International Journal of Digital Earth*, 14(3), 288-310. <https://doi.org/10.1080/17538947.2020.1822452>
- Sieber, R., Schmuki, M., & Hurni, L. (2021). Storytelling in Interactive Atlases – Following the Intrinsic Map-Centered Approach. *Abstracts of the ICA*, 3, 248. <https://doi.org/10.5194/ica-abs-3-248-2021>
- Sieber, R., Schnürer, R., Eichenberger, R., & Hurni, L. (2013). The Power of 3D Real-Time Visualization in Atlases – Concepts, Techniques and Implementation. *Proceedings of the 26th International Cartographic Conference (ICC)*. https://icaci.org/files/documents/ICC_proceedings/ICC2013/_extendedAbstract/
- Sieber, R., Serebryakova, M., Schnürer, R., & Hurni, L. (2016). Atlas of Switzerland Goes Online and 3D– Concept, Architecture and Visualization Methods. In G. Gartner, M. Jobst, & H. Huang (Eds.), *Progress in Cartography* (pp. 171-184). Springer International Publishing. https://doi.org/10.1007/978-3-319-19602-2_11
- Sigurjónsson, T., Bjerva, T., & Græsli, J. A. (2020). Gender differences in children's wayfinding. *International Journal of Cartography*, 6(3), 284-301. <https://doi.org/10.1080/23729333.2020.1757214>
- Singh, J. M., & Narayanan, P. J. (2010). Real-time ray tracing of implicit surfaces on the GPU. *IEEE Transactions on Visualization and Computer Graphics*, 16(2), 261-272.
- Slusallek, P., Shirley, P., Mark, W., Stoll, G., & Wald, I. (2005). Introduction to real-time ray tracing. *ACM SIGGRAPH 2005 Courses*. <https://doi.org/10.1145/1198555.1198740>
- Stähli, L., Rudi, D., & Raubal, M. (2018). Turbulence Ahead—A 3D Web-Based Aviation Weather Visualizer. *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, 299-311. <https://doi.org/10.1145/3242587.3242624>
- Steiniger, S., Taillandier, P., & Weibel, R. (2010). Utilising urban context recognition and machine learning to improve the generalisation of buildings. *International Journal of Geographical Information Science*, 24(2), 253-282. <https://doi.org/10.1080/13658810902798099>
- Streifeneder, T., & Piatti, B. (2021). Matthew Picton's Urban Narratives. Or how a three-dimensional paper map can beam you into the London bombing nights of 1940. *International Journal of Cartography*, 7(2), 233-239. <https://doi.org/10.1080/23729333.2021.1921379>
- Takahashi, S., Ohta, N., Nakamura, H., Takeshima, Y., & Fujishiro, I. (2002). Modeling Surperspective Projection of Landscapes for Geographical Guide-Map Generation. *Computer Graphics Forum*, 21(3), 259-268. <https://doi.org/10.1111/1467-8659.t01-1-00585>
- Tateosian, L., Glatz, M., & Shukunobe, M. (2020). Story-telling maps generated from semantic representations of events. *Behaviour & Information Technology*, 39(4), 391-413. <https://doi.org/10.1080/0144929X.2019.1569162>
- Techt, R. (2020). *Entwicklung eines semi-automatischen Workflows zur Ableitung ikonographischer Kartenzeichen* [Master's thesis]. Technische Universität Dresden.
- Terribilini, A. (1999). Maps in transition: Development of interactive vector-based topographic 3D-maps. *Proceedings of the 19th International Cartographic Conference (ICC)*, 9. https://icaci.org/files/documents/ICC_proceedings/ICC1999/proceedings_ICC1999.zip

- Tewari, A., Thies, J., Mildenhall, B., Srinivasan, P., Tretschk, E., Yifan, W., Lassner, C., Sitzmann, V., Martin-Brualla, R., Lombardi, S., Simon, T., Theobalt, C., Nießner, M., Barron, J. T., Wetzstein, G., Zollhöfer, M., & Golyanik, V. (2022). Advances in Neural Rendering. *Computer Graphics Forum*, 41(2), 703-735. <https://doi.org/10.1111/cgf.14507>
- Thöny, M., & Pajarola, R. (2015). Vector Map Constrained Path Bundling in 3D Environments. *Proceedings of the 6th ACM SIGSPATIAL International Workshop on GeoStreaming*, 33-42. <https://doi.org/10.1145/2833165.2833168>
- Tian, J., Song, Z., Gao, F., & Zhao, F. (2016). Grid pattern recognition in road networks using the C4.5 algorithm. *Cartography and Geographic Information Science*, 43(3), 266-282. <https://doi.org/10.1080/15230406.2015.1062425>
- Tomczak, L. J. (2012). *GPU Ray Marching of Distance Fields* [Master thesis]. Technical University of Denmark.
- Tominski, C., & Schulz, H.-J. (2012). The Great Wall of Space-Time. In M. Goesele, T. Grosch, H. Theisel, K. Toennies, & B. Preim (Eds.), *Vision, Modeling and Visualization*. The Eurographics Association. <https://doi.org/10.2312/PE/VMV/VMV12/199-206>
- Tominski, C., Schumann, H., Andrienko, G., & Andrienko, N. (2012). Stacking-Based Visualization of Trajectory Attribute Data. *IEEE Transactions on Visualization and Computer Graphics*, 18(12), 2565-2574. <https://doi.org/10.1109/TVCG.2012.265>
- Torres, R. N., Fraternali, P., Milani, F., & Frajberg, D. (2018). A Deep Learning Model for Identifying Mountain Summits in Digital Elevation Model Data. *2018 IEEE First International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, 212-217. <https://doi.org/10.1109/AIKE.2018.00049>
- Touya, G., Brisebard, F., Quinton, F., & Courtial, A. (2020). Inferring the scale and content of a map using deep learning. *ISPRS Congress 2020, XLIII(B4)*, 17-24. <https://doi.org/10.5194/isprs-archives-XLIII-B4-2020-17-2020>
- Trapp, M., Beesk, C., Pasewaldt, S., & Döllner, J. (2011). Interactive Rendering Techniques for Highlighting in 3D Geovirtual Environments. In T. H. Kolbe, G. König, & C. Nagel (Eds.), *Advances in 3D Geo-Information Sciences* (pp. 197-210). Springer. https://doi.org/10.1007/978-3-642-12670-3_12
- Trapp, M., & Dollner, J. (2009). Dynamic Mapping of Raster-Data for 3D Geovirtual Environments. *Proceedings of the 2009 13th International Conference Information Visualisation*, 387-392. <https://doi.org/10.1109/IV.2009.28>
- Trapp, M., Glander, T., Buchholz, H., & Döllner, J. (2008). 3D Generalization Lenses for Interactive Focus + Context Visualization of Virtual City Models. *2008 12th International Conference Information Visualisation*, 356-361. <https://doi.org/10.1109/IV.2008.18>
- Traxler, C., Ortner, T., Hesina, G., Barnes, R., Gupta, S., Paar, G., Muller, J.-P., Tao, Y., & Willner, K. (2022). The PRoViDE Framework: Accurate 3D Geological Models for Virtual Exploration of the Martian Surface from Rover and Orbital Imagery. In *3D Digital Geological Models* (pp. 33-55). John Wiley & Sons, Ltd. <https://doi.org/10.1002/9781119313922.ch3>
- Uhl, J. H., Leyk, S., Chiang, Y.-Y., Duan, W., & Knoblock, C. A. (2017, July). Extracting human settlement footprint from historical topographic map series using context-based machine learning. *8th International Conference of Pattern Recognition Systems (ICPRS 2017)*. <https://doi.org/10.1049/cp.2017.0144>
- Vaaranemi, M., Treib, M., & Westermann, R. (2011). High-quality cartographic roads on high-resolution DEMs. *Journal of WSCG*, 19, 41-48. http://wscg.zcu.cz/DL/wscg_DL.htm
- Vannieuwenhuyze, B. (2020). *Entangled Maps; Topography and Narratives in Early Modern Story Maps**. Motion in Maps, Maps in Motion: Mapping Stories and Movement through Time; Amsterdam University Press. <https://doi.org/10.1017/9789048542956.003>
- Vassányi, G., & Gede, M. (2021). Automatic vectorization of point symbols on archive maps using deep convolutional neural network. *Proceedings of the ICA*, 4, 109. <https://doi.org/10.5194/ica-proc-4-109-2021>

References

- Vollmer, J. O., Trapp, M., Schumann, H., & Döllner, J. (2018). Hierarchical Spatial Aggregation for Level-of-Detail Visualization of 3D Thematic Data. *ACM Transactions on Spatial Algorithms and Systems*, 4(3), 9:1-9:23. <https://doi.org/10.1145/3234506>
- Wartell, Z. J., Kang, E., Wasilewski, A. A., Ribarsky, W., & Faust, N. L. (2003). *Rendering Vector Data Over Global, Multi-resolution 3D Terrain* [Technical Report]. Georgia Institute of Technology. <https://smartech.gatech.edu/handle/1853/3219>
- Weber, A.-K. (2014). *Mapping literature: Spatial data modelling and automated cartographic visualisation of fictional spaces* [Dissertation]. ETH Zurich.
- Wu, A. N., & Biljecki, F. (2022). GANmapper: Geographical data translation. *International Journal of Geographical Information Science*, 36(7), 1394-1422. <https://doi.org/10.1080/13658816.2022.2041643>
- Wu, S., Schnürer, R., Heitzler, M., & Hurni, L. (2022). Unsupervised historical map registration by a deformation neural network. *Proceedings of the 5th ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery*, 76-81. <https://doi.org/10.1145/3557918.3565871>
- Wu, Y., Filippovska, Y., Schmidt, V., & Kada, M. (2019). Application of Deep Learning for 3D building generalization. *Proceedings of the ICA*, 2, 147. <https://doi.org/10.5194/ica-proc-2-147-2019>
- Wu, Z., Wang, N., Shao, J., & Deng, G. (2019). GPU ray casting method for visualizing 3D pipelines in a virtual globe. *International Journal of Digital Earth*, 12(4), 428-441. <https://doi.org/10.1080/17538947.2018.1429504>
- Xie, H., Li, D., Wang, Y., & Kawai, Y. (2022). A Graph Neural Network-Based Map Tiles Extraction Method Considering POIs Priority Visualization on Web Map Zoom Dimension. *IEEE Access*, 10, 64072-64084. <https://doi.org/10.1109/ACCESS.2022.3182497>
- Xie, X., Liu, Y., Xu, Y., He, Z., Chen, X., Zheng, X., & Xie, Z. (2022). Building Function Recognition Using the Semi-Supervised Classification. *Applied Sciences*, 12(19), Article 9900. <https://doi.org/10.3390/app12199900>
- Yan, X., Ai, T., Yang, M., & Yin, H. (2019). A graph convolutional neural network for classification of building patterns using spatial vector data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150, 259-273. <https://doi.org/10.1016/j.isprsjprs.2019.02.010>
- Yang, M., Jiang, C., Yan, X., Ai, T., Cao, M., & Chen, W. (2022). Detecting interchanges in road networks using a graph convolutional network approach. *International Journal of Geographical Information Science*, 36(6), 1119-1139. <https://doi.org/10.1080/13658816.2021.2024195>
- Yang, M., Yuan, T., Yan, X., Ai, T., & Jiang, C. (2022). A hybrid approach to building simplification with an evaluator from a backpropagation neural network. *International Journal of Geographical Information Science*, 36(2), 280-309. <https://doi.org/10.1080/13658816.2021.1873998>
- Yang, T.-H., Kim, J. R., Jin, H., Gil, H., Koo, J.-H., & Kim, H. J. (2021). Recent Advances and Opportunities of Active Materials for Haptic Technologies in Virtual and Augmented Reality. *Advanced Functional Materials*, 31(39), 2008831. <https://doi.org/10.1002/adfm.202008831>
- Yang, X., Wang, G., Yan, J., & Gao, J. (2021). T2I-CycleGAN: A CycleGAN for Maritime Road Network Extraction from Crowdsourcing Spatio-Temporal AIS Trajectory Data. In H. Gao, X. Wang, M. Iqbal, Y. Yin, J. Yin, & N. Gu (Eds.), *Collaborative Computing: Networking, Applications and Worksharing* (pp. 203-218). Springer International Publishing. https://doi.org/10.1007/978-3-030-67540-0_12
- Ye, X., Du, J., & Ye, Y. (2022). MasterplanGAN: Facilitating the smart rendering of urban master plans via generative adversarial networks. *Environment and Planning B: Urban Analytics and City Science*, 49(3), 794-814. <https://doi.org/10.1177/23998083211023516>
- Yu, W., & Chen, Y. (2022). Data-driven polyline simplification using a stacked autoencoder-based deep neural network. *Transactions in GIS*, 26(5), 2302-2325. <https://doi.org/10.1111/tgis.12965>
- Zagata, K., Gulij, J., Halik, Ł., & Medyńska-Gulij, B. (2021). Mini-Map for Gamers Who Walk and Teleport in a Virtual Stronghold. *ISPRS International Journal of Geo-Information*, 10(2), 96. <https://doi.org/10.3390/ijgi10020096>

- Zanda, A., Lutz, J., Heymann, A., & Bleisch, S. (2019). Technological infrastructure supporting the story network principle of the Atlas of the Ageing Society. *Geografie*, 124(2), 217-235. <https://doi.org/10.37040/geografie2019124020217>
- Zheng, J., Gao, Z., Ma, J., Shen, J., & Zhang, K. (2021). Deep Graph Convolutional Networks for Accurate Automatic Road Network Selection. *ISPRS International Journal of Geo-Information*, 10(11), Article 768. <https://doi.org/10.3390/ijgi10110768>
- Zhou, M., Chen, J., & Gong, J. (2013). A pole-oriented discrete global grid system: Quaternary quadrangle mesh. *Computers & Geosciences*, 61, 133-143. <https://doi.org/10.1016/j.cageo.2013.08.012>
- Zhou, Q., & Li, Z. (2017). A Comparative Study of Various Supervised Learning Approaches to Selective Omission in a Road Network. *The Cartographic Journal*, 54(3), 254-264. <https://doi.org/10.1179/1743277414Y.0000000083>
- Zhou, X., Li, W., Arundel, S., & Liu, J. (2018). *Deep Convolutional Neural Networks for Map-Type Classification*. arXiv. <https://doi.org/10.48550/arXiv.1805.10402>
- Zhou, Z., Fu, C., & Weibel, R. (2022). Building simplification of vector maps using graph convolutional neural networks. *Abstracts of the ICA*, 5, 86. <https://doi.org/10.5194/ica-abs-5-86-2022>

Notes

This list includes the references for Chapter 1 (Introduction), Chapter 2 (Background), Chapter 6 (Conclusion), and Chapter 7 (Outlook). The references of the research articles given in Chapters 3 to 5 are excluded.

Names and locations of conferences were omitted for articles published in conference proceedings for the sake of brevity.

All internet resources were last accessed at 11 July 2023.