Visualizing Urban Analysis in Mixed Reality

Lukas Treyer, Anastasia Koltsova, Dr. Sofia Georgakopoulou
Chair of Information Architecture
Swiss Federal Institute of Technology
Zurich, Switzerland
{treyer, koltsova, georgakopoulou}@arch.ethz.ch

Abstract—This paper focuses on the computational creation of animated data visualization used to communicate urban analysis and design in a video. The goal is to combine abstract analysis information with concrete qualitative impressions of a location in the case study area of Zurich. Different urban analysis methods are used to procedurally create an animation with the open source software Blender that facilitates also the compositing process of animation with video. The resulting videos are created during an elective course at ETH Zurich. Anticipating the advent of augmented and mixed reality applications for daily life, we elaborate on their usability for urban design, education and collaborative planning with relatively easy to learn video effect methods. Not only different levels of abstraction are visualized but also – inherent to the medium of video – the relation to time.

mixed reality, urban design and analysis, architectural communication

I. INTRODUCTION

Architects have a long tradition of communicating a design with static / printed media. Though with the increasing amount of data introduced with dynamic simulation methods the understandability of traditional media is being pushed to its limits. Also, the possibilities for collaboration over the Internet through social media websites and similar communities ask for dynamic architectural communication.

Dynamic media like video or games allow the viewer to understand time related aspects of a design potentially better. They also facilitate the creation of immersive environments that would let the viewer experience physical reality being merged with virtual overlays.

Videos and animations are accessible media for designers and planners when they want to communicate with an audience of non-specialists. With concrete imagery designers can link to layman people’s memorized images while at the same time visualizing also abstract information on top of it and therefore teach the untrained eye. Only in the recent years technology evolved far enough to enable designers and film makers not only to take video as an input and composite additional information on top of it. But also the means to produce more high quality video streams became more ubiquitous[1]. New in our approach is that the videos are used not only for communicating design proposals but also to visualize the output of computational analysis, as shown with the visibility analysis method in section II.A. This allows the designer to show computational and qualitative analysis in the same media.

II. METHOD

A. Teaching

In the elective course held at ETH Zurich the students learn to visualize urban analysis directly in video. Even though some of them have experience with creating 3D visualizations, animated visualizations are new to most of them. The goal of the course is to elaborate on suitability and usability of video effect visualization for architects. Furthermore the students learn how to use programming not only for the creation of their designs but also for their problem analysis and communication.

B. Urban Analysis

1) Visibility Analysis

Our visibility analysis method differs from the well known Isovist method that was defined 1979 by M.L. Benedikt in order to uniformly quantify visibility in space[2]. Since we are more interested in the visibility of façades rather than the visibility of the ground floor area, we ray-trace every point on the façade to a given vantage point. Thereby not only visibility is being detected but also the angle between the ray and the normal of the façade is being calculated. The points on the façade result from its subdivision into smaller faces. In the resulting video the vantage points have been attached to real people appearing in the input video. The visibility analysis algorithm is implemented in Python using math-functions built into Blender[3].

2) Labeling

We define labeling simply as the process of naming and highlighting hidden information. The technical details will be covered in section D. Compositing and Motion Tracking. On one hand this allows the students to integrate qualitative analysis into their visualization. On the other hand they can visualize datasets in the form of diagrams or symbols and attach those to real objects in the video. Datasets we have access to include data from the open-data initiative of the City of Zurich[4], as well as some traffic count numbers of certain bus lines.

Despite automatic labeling algorithms have been developed in combination with image segregation[5], we argue that this state-of-the-art technology is not suitable for a teaching class where students have to learn to express themselves with video
and information visualization composited into video. Furthermore such algorithms are limited to the database used to compare segmented image parts. This might lead to situations in which the designer or student has to label certain qualities by hand anyway.

C. Video

Video (film) is a medium to qualitatively understand how people perceive their environment. Therefore for urban designers as well as for geographers qualitative methods are vital to “understand public perceptions of the world”, as Jon Anderson points out [6]. He puts the students’ experience while filming into the context of psychogeography, a term introduced by Guy Dubord in 1955 [7]. Also Joreon Beekmans reports on a revival of psychography [8]. He adds: “In fact, psychogeography is the art of strolling, or just about anything that gets pedestrians off their predictable paths and leads them to a new awareness of the urban landscape”.

D. Compositing and Motion Tracking

The tool used to create animated visualizations of urban analysis and then composite it over video footage is called “Blender”. It is open-source and actively developed by the Blender Foundation and its online community [3]. It is designed to cover the whole workflow of creating a video animation and/or video effects right after the storyboard stage. Besides that, it is used for game development as well as scientific and architectural visualizations.

Since 2012 Blender has been equipped also with a “Motion Tracking” editor. It allows users to track pixel patterns in an arbitrary video stream in a non-realtime fashion. With eight of these points being tracked in two pictures taken from two different angles, the built-in structure-from-motion algorithm is able to calculate the position of the tracking points in space and the movement of the camera. In other words the algorithm is only able to position the tracking points in space, if the camera is moving, because only then two frames can be compared, that were taken at a different position.

If the camera is not changing its position but only rotating on a tripod, the tool can calculate a “Tripod Motion”. The 3D points that correspond to the tracking points then are being positioned in a circle around the camera, and the virtual camera’s rotation is being animated according to the rotation speed of the real camera.

In order to fully integrate a virtual object in a movie, Blender has a mask tool that is able to cut out portions of the movie and composite them over the virtual object. Masks can be animated with key frames. Therefore the artist can make them exactly following an object in the video footage.

The compositing can be done Blender as well. Compositions in Blender are node-trees. The nodes represent images or manipulation of image-data. Two images for instance can be merged with a mix node that in the end would send its result to an output node that would store the frame.

III. RESULTS

The resulting videos are represented by the two still images below. They are the result of a student course in fall 2012 (Fig.1) and of the preparation for it in spring 2012 (Fig. 2).

![Figure 1. Visibility Analysis composited over video footage of ETH Zurich Science City campus.](image1)

![Figure 2. Labeling with symbols and traffic visualization. Towers in the foreground are masked out.](image2)

IV. CONCLUSION

The potential for architects using video visualizations can be found in the better possibilities for public communication. The process of creating the video itself can generate some insights about urban qualities since the students have to stroll through the city, i.e., they have to deal with the urban fabric in a concrete fashion, not only discussing about with the help of a plan. The combination of abstract and concrete imagery might also lead to a clearer understanding of the analysis itself, though proving this hypothesis is subject of further research.

The course also showed the most important steps that need to be taken to make mixed reality applications for architects more plausible. From a technical viewpoint these would be
real-time motion tracking in order to composite a game-like visualization on top of an input video stream. Also the integration of masking into the tracking algorithm would be necessary for real-time applications, in order to avoid the most common error of a tracking pattern passing behind another object.

From the architectural point of view we observe, that a visualization culture with video has not yet been established. The reason might be too little appreciation by clients and/or a shortage of time among students who learn other tools first. We observed as well that the learning curve of the tool “Blender” and mixed reality applications in general are rather steep in comparison to CAD-tools that mostly conform to each other. Nonetheless, mixed reality applications attract architects. The idea of immersion, to fusion virtual and physical reality, is still considered to have a strong potential in architectural communication – for clients as well as students. In future work we would aim at flattening the learning curve and probably provide more accessible workflows that also allow for real-time data processing and spatio-temporal analysis.

ACKNOWLEDGMENT

We thank our students for their passionate contributions, especially Li Li, PhD student at CAAD ETH Zurich, who helped with the implementation of the visibility visualization in Python.

REFERENCES


