Urban Canvas: Unfreezing Street-view Imagery with Semantically Compressed LIDAR Pointclouds

Thommen Korah* Yun-Ta Tsai †

Nokia Research Center Hollywood, CA

ABSTRACT

Detailed 3D scans of urban environments are increasingly being collected with the goal of bringing more location-aware content to mobile users. This work converts large collections of LIDAR scans and street-view panoramas into a representation that extracts semantically meaningful components of the scene. Compressing this data by an order of magnitude or more enables rich user interactions with mobile applications that have a very good knowledge of the scene around them. These representations are suitable for integrating into physics engines and transmission over mobile networks – key components of modern AR entertainment solutions.

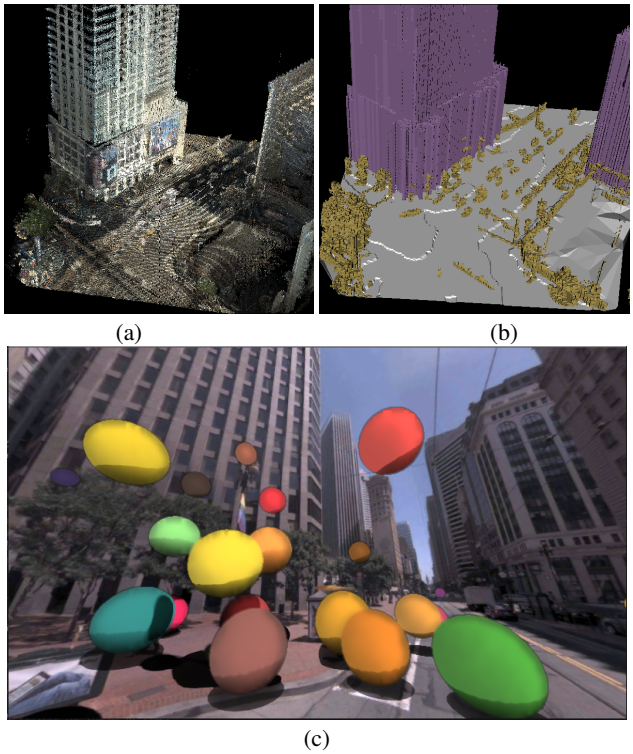
Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities; I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Tracking

1 INTRODUCTION

Augmented Reality is one of the driving technologies motivating large-scale collection of location content. Realizing the power of places and mobile information browsing, companies have invested heavily on acquiring detailed maps of the real world. This has resulted in applications such as maps, turn-by-turn navigation, exploration of high-resolution street-view panoramas, and detailed 3D models of urban cities. A big gap still exists between just presenting the petabytes of collected data and realizing the AR vision of virtual objects seamlessly interacting with the real – anywhere and anytime. We introduce a fully automatic method to recover semantic geometry, which we call the urban canvas, from raw sensor data captured in the form of LIDAR pointclouds and panoramic street view imagery. This is integrated with physics and lighting for outdoor AR applications.

Our approach builds the urban canvas in an off-line stage by processing pre-collected LIDAR data of a large urban environment. LIDAR provides a more attractive option to recovering scene geometry, albeit as a set of raw unstructured points, at the scale of multiple cities. The canvas also serves as the platform to interpret and reason about the world on which virtual objects are augmented. An automatic and efficient segmentation algorithm first extracts individual objects from millions of 3D points. The separated regions allow us to compress the points in a semantically meaningful manner, reducing storage/transmission costs by more than an order of magnitude. The recovered scene geometry can be aligned with our scene aware panorama viewer and integrated into a light-weight game engine. This allows augmented content to obey correct occlusion, physics, and lighting in the form of cast shadows.

Previous work in outdoor AR [3] using vision-based algorithms are too brittle to run in large city-scale environments. LIDAR data is more reliable under adverse environmental conditions and provides better geometric registration. Panoramic street view images are increasingly popular for remote exploration and finding landmark sites. Wither [4] recently proposed an indirect AR approach that works around the registration problem by simulating an AR experience with panoramas of the scene. By capturing the world behind the image, we aim to provide a more immersive experience than using panoramas alone.



(a) (b) (c)

Figure 1: (a) A dense set of colored LIDAR pointclouds from downtown Los Angeles containing over 18 million points and occupying 17 megabytes on disk; (b) semantically compressed representation of the underlying scene occupying only 220 kilobytes; (c) demonstration of a scene aware panorama viewer that can integrate the semantic representation with physics, environmental lighting, and shadows.

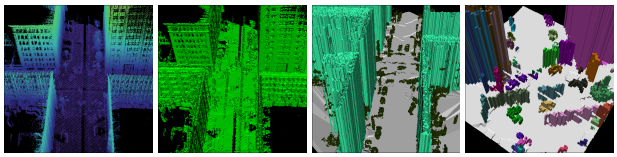
2 BUILDING THE URBAN CANVAS

We now describe the overall architecture for the proposed urban canvas system. The back-end is composed of algorithms that process LIDAR data and street-view panoramas captured by NAVTEQ. This stage, designed to run offline, transforms the data into a more convenient form for information extraction. The methods include automatic alignment of the LIDAR points and correction of drift based on shape priors [2], segmentation of the 3D points into individual objects and object recognition [1], terrain extraction, and semantic classification. The front end client is an interface for remote viewing of street-view imagery on a mobile interface. It consists

*email: thommen.korah@nokia.com

†email:yun-ta.tsai@nokia.com

IEEE International Symposium on Mixed and Augmented Reality 2011
Science and Technolgy Proceedings
26 -29 October, Basel, Switzerland
978-1-4577-2185-4/10/$26.00 ©2011 IEEE



(a) (b) (c) (d)

Figure 2: Different compression schemes. (a) Input pointcloud requiring 59 megabytes (MB). (b) A reconstructed isosurface that can be stored in 11.3 MB. (c) Segmented Voxel Grid that stores the segmentation labels as well as the semantic classification of ground, object, or building wall. This can be stored in 340 KB and is most suitable for applications like games that can better exploit the preserved geometry. (d) Segmented Strip Grid only stores the strip height from the grid of vertical histograms and the segmentation label. This scheme occupies only 58KB and is most useful for occlusion reasoning or visibility computation when augmenting simple content in AR applications.

of a viewer that is capable of presenting situated content layered on panoramas or live camera views. The platform is intended to enable games and other forms of location-based entertainment that can exploit a true understanding of the 3D canvas on which the virtual content is augmented.

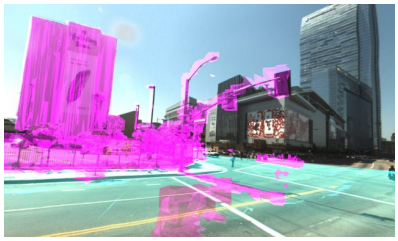


Figure 3: Mesh recovered from 3D processing engine overlaid with panoramic imagery. Cyan indicates terrain and magenta indicates vertical surfaces.

Pointcloud segmentation allows us to classify the unstructured set of points into 3 semantic classes – ground, building wall, and other objects. An example of the classification is shown in Figure 1(b). The result of [1] provides us a local ground estimate at uniform points on a 2D grid. We first build a continuous bare-earth terrain map mesh from these point samples by performing Delaunay Triangulation. The recovered ground is vital for reasoning about the scene. The pointcloud is represented as a grid of vertical histograms called the Strip Histogram Grid [1]. Objects in the scene are then segmented by grouping adjacent strips with similar characteristics.

These semantic classes convert the pointcloud into a compressed representation suitable for transmission over wireless connections and integrating into game engines. Location-based applications are better able to exploit scene knowledge instead of relying purely on GPS location. Previous work on 3D compression assumes a mesh of a single object and treat all the points as a single 3-dimensional signal. We introduce two schemes of semantic compression that reduce the pointcloud to a more meaningful representation. The *Segmented Voxel Grid* (SVG) scheme first discards all points at or below the ground level and replaces it with the terrain mesh. A voxel representation of the pointcloud, where each voxel stores the segmentation label of points within, is then written to disk. Standard compression schemes like gzip work best with this kind of data containing lots of repeating elements. The *Segmented Strip Grid* (SSG) is a more aggressive compression scheme where only the height and segmentation label of populated strips are saved with the ground mesh. Non-gaming AR applications that layer static content and only require approximations of object location and size can use this with significantly reduced storage requirements. Figure 2 illustrates the geometry of the different methods with typical storage sizes.

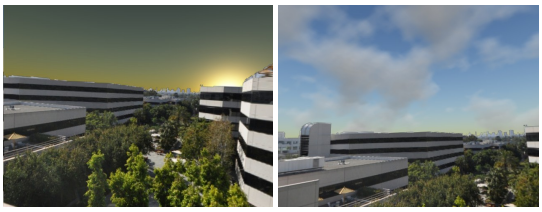


Figure 4: Synthetic sky and simulated weather replacement.

3 Scene-aware Panorama Viewer

The front end client integrates the representation computed above into a light-weight game engine on which street-view imagery can be viewed. The client goes beyond just presenting a 2D image and acts as the interface to a scene-aware panorama viewer. Recent studies [4] have shown that panoramas are often better than live camera views for location-based apps. The client submits its GPS location and retrieves the scene model from the server. The mesh is then aligned with the panorama captured at the location. As fig. 3 shows, the viewer is equipped with an accurate model of all the elements in the scene: traffic light, light standards, buildings, fences, and trees. With physics and lighting integrated into this framework, the client viewer becomes an ideal playing ground for virtual objects to interact seamlessly with the panoramic scene.

We apply a number techniques within this Mixed Reality framework to better mimick the real world. The simplified geometry is fed into a physics engine for simulating motion and collisions. We use image-based rendering techniques where a blurred panorama is used as environmental diffuse lighting for relighting virtual objects. We also take distance of the objects from the eye into consideration; further away the object, stronger the diffuse lighting and less the saturation due to attenuation and scattering. By estimating the sun position in the sky, we compute the shadow map to cast shadows on both panoramas and virtual content. In a similar vein, we can also alter the background panorama to better reflect current conditions. The segmented sky from the panorama can be replaced with a 24-hour cycle simulation of the sky as in Fig. 4. Additionally, color transfer methods allow us to alter the color tone of the original panorama.

4 Conclusion

This work presents a novel end-to-end pipeline to utilize the large amount of 3D LIDAR data acquired from urban environments. The pipeline to build the *urban canvas* includes 3D segmentation of pointclouds to extract semantically meaningful real objects from the data. We present a technique to compress points into a form that is amenable to transferring across low-bandwidth mobile networks. The captured geometry and semantics are exported to a front-end panorama viewer client that is aware of the 3D scene from which the image was captured. Integrated with physics and advanced rendering techniques, we demonstrate examples of mobile applications that can exploit the additional scene knowledge. Future work includes exploiting better knowledge of geometry and illumination conditions for more realistic augmentations on the canvas.

References

[1] T. Korah, S. Medasani, and Y. Owechko. Strip histogram grid for efficient lidar segmentation from urban environments. In *Workshop on Object Tracking and Classification Beyond the Visible Spectrum*, 2011.

[2] T. Pylvanainen, K. Roimela, R. Vedantham, J. Itaranta, R. Wang, and R. Grzeszczuk1. Automatic alignment and multi-view segmentation of street view data using 3d shape priors. In *Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 2010.

[3] G. Reitmayr and T. W. Drummond. Going out: Robust tracking for outdoor augmented reality. In *Proc. ISMAR*, 2006.

[4] J. Wither, Y.-T. Tsai, and R. Azuma. Indirect augmented reality. *Computers & Graphics*, In Press, Accepted Manuscript, 2011.