Automatic 2D-3D Registration

Student: Lingyun Liu

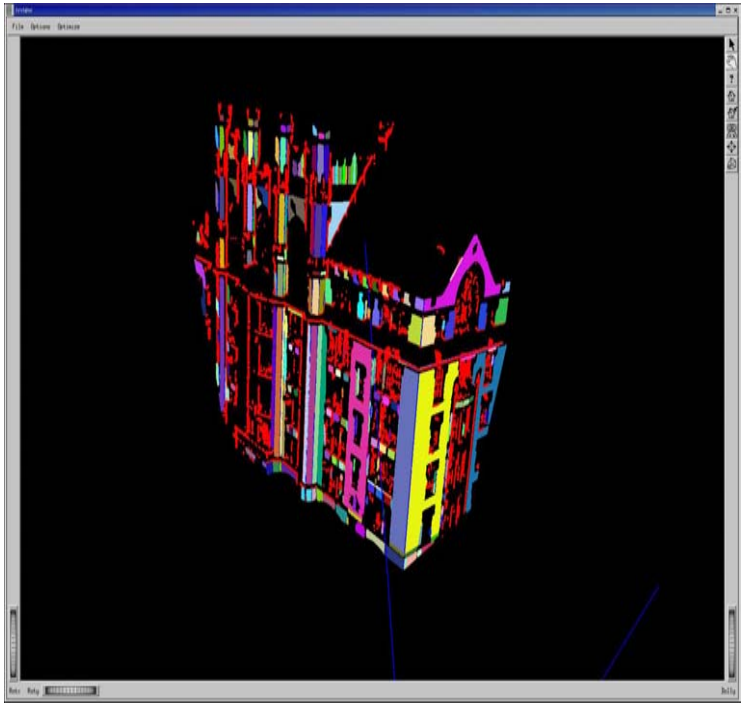
Advisor: Prof. Ioannis Stamos

Abstract:

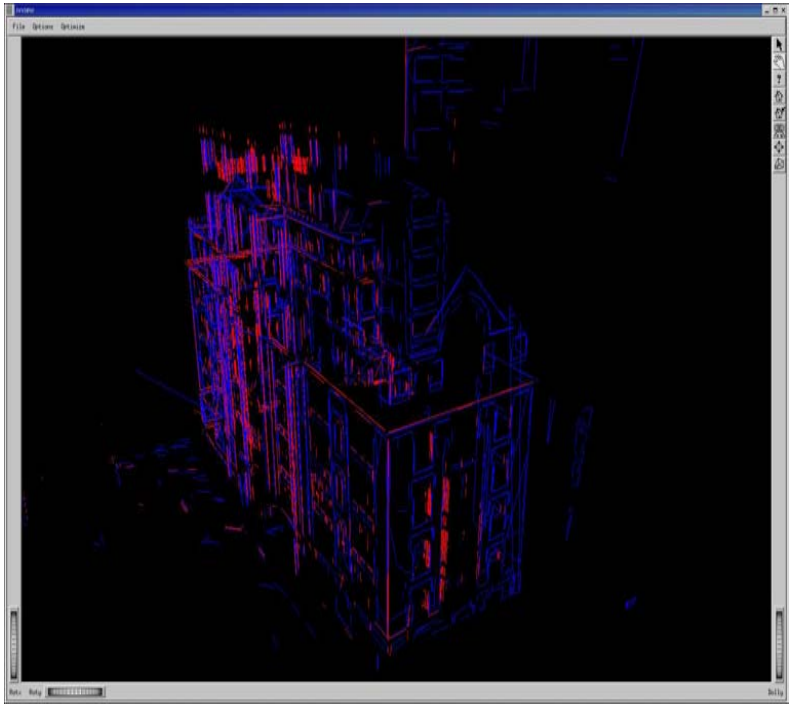
- Given 3D model constructed from range images of a real-world scene and set of 2D images, we want to apply textures from those 2D images to the model automatically. We propose an approach that uses line features to automatically find the correspondences between 2D and 3D images, once the correspondences are established, we compute texture coordinates mapping portions of the 2D images to model surfaces.

Acquire the lines from 3D images

1. Border lines [1,2]



Segmentation result of 1 scan

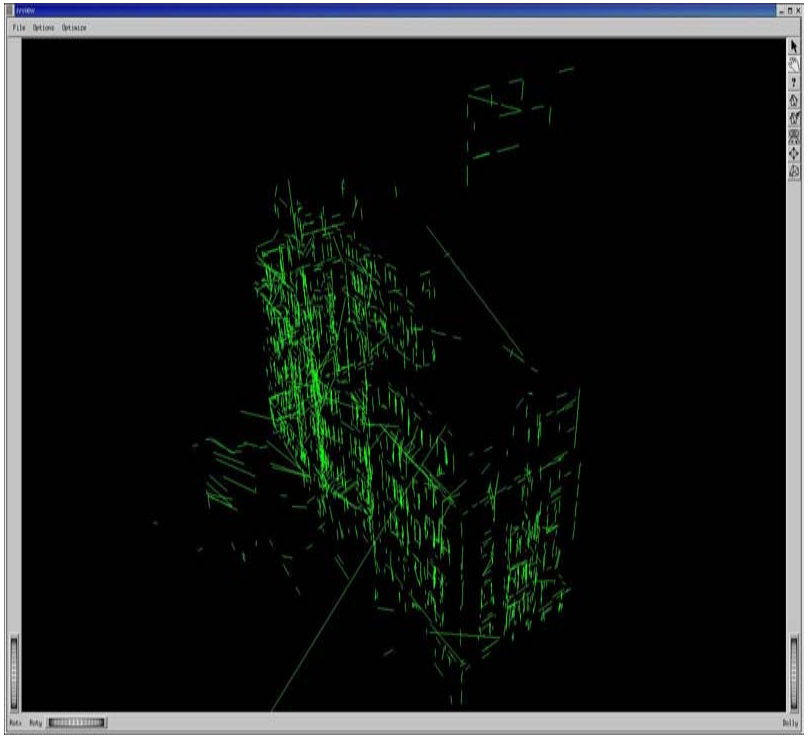


The border lines from 15 scans (registered)

Acquire the lines from 3D images

2. Lines from reflectance images (edge detector)

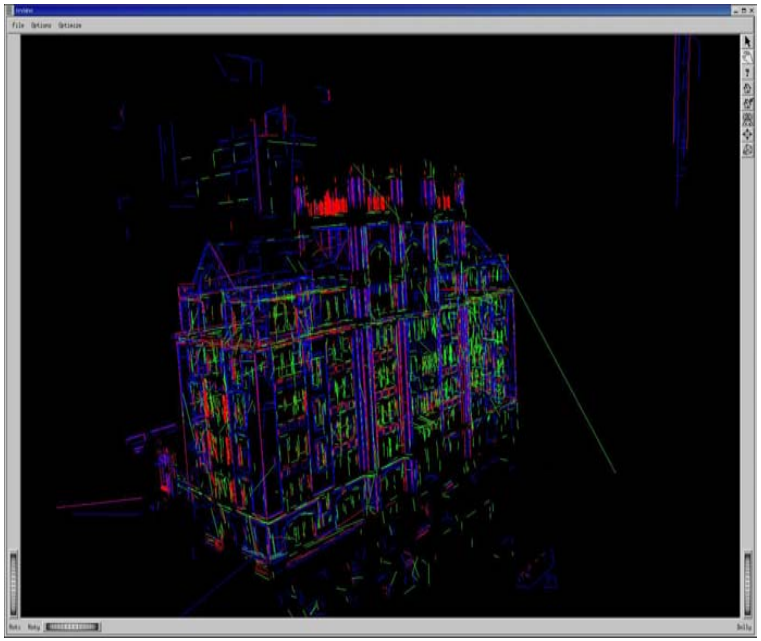
Reflectance image of 1 scan



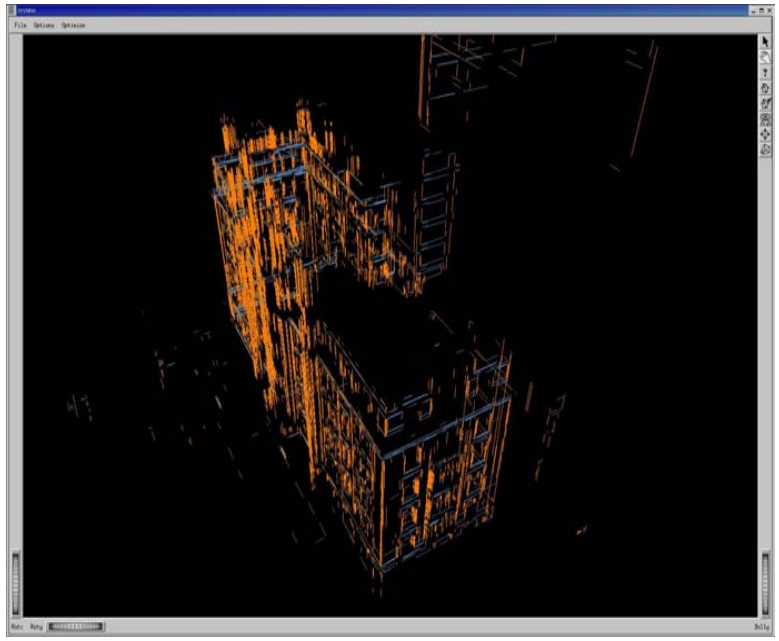
Edge lines from reflectance images of 15 scans (registered)

Acquire the lines from 3D images

Merge those 2 sets of lines and cluster them



Raw line model consisting of border lines and reflectance lines (registered)

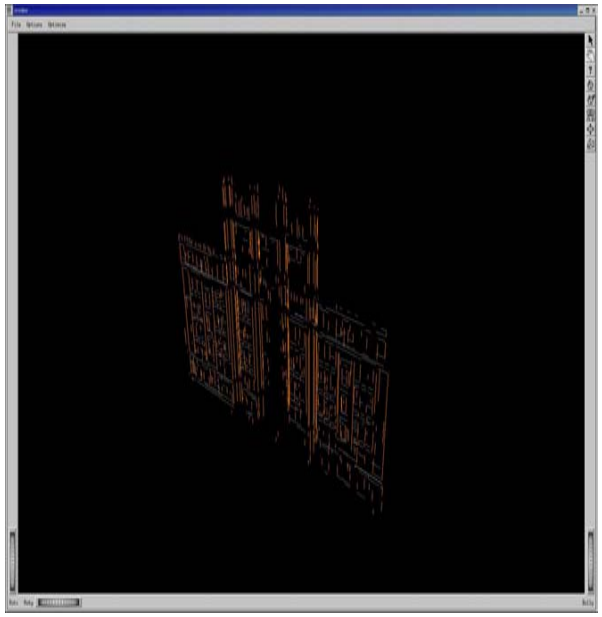


Updated line model with 3 major direction lines (x^,y^,z^) (registered)

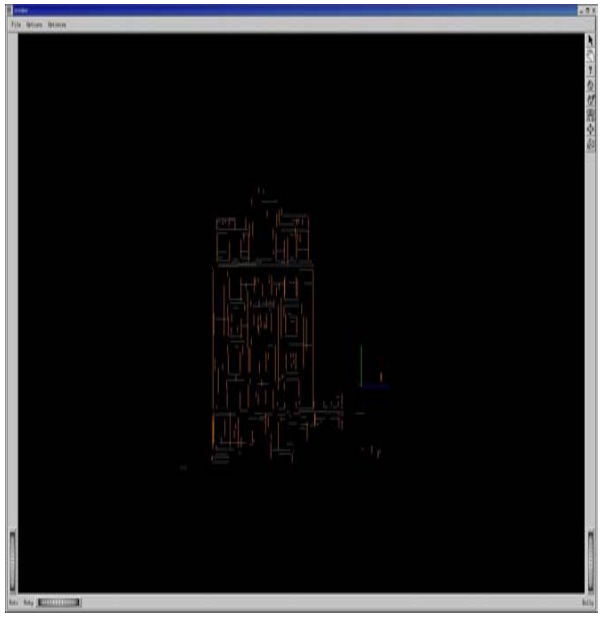
Acquire the lines from 3D images

Extract the face lines

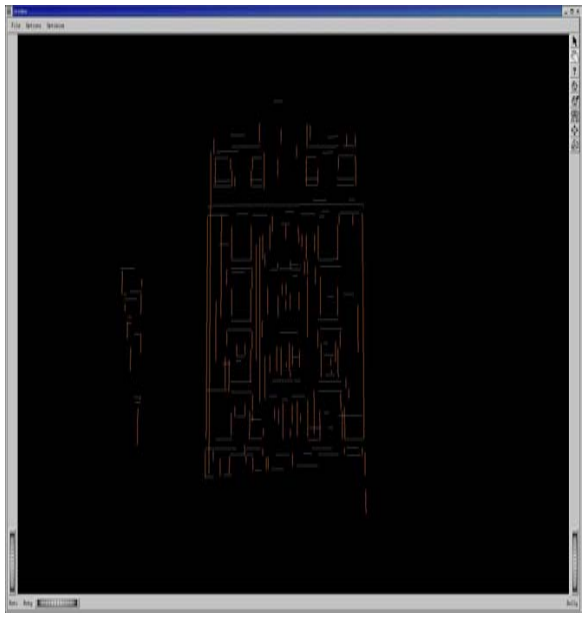
front



left



right



Advanced clustered lines, each set of lines belong to 1 face of the model. (extracted from updated line model)

Acquire the lines from 2D images

Edge Detection (canny edge detector)

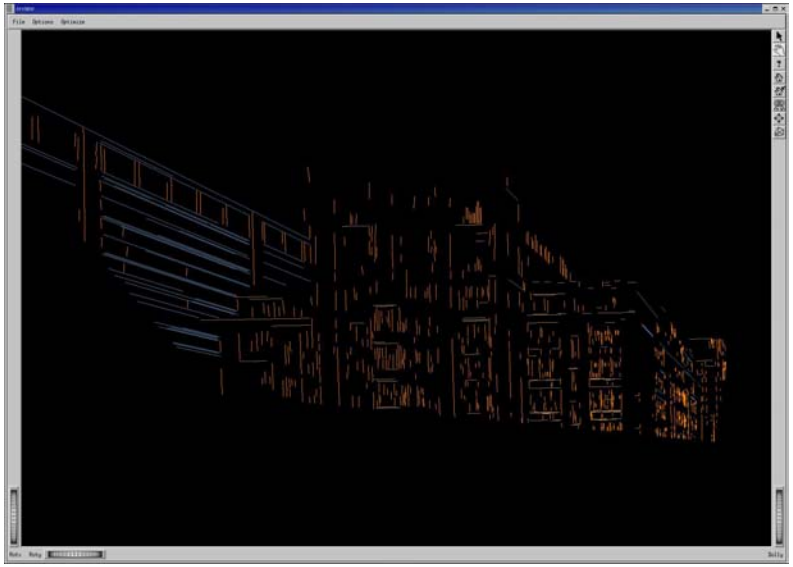
Input 2D image

After edge detection (red lines are the lines extracted from blue edges)

Acquire the lines from 2D images

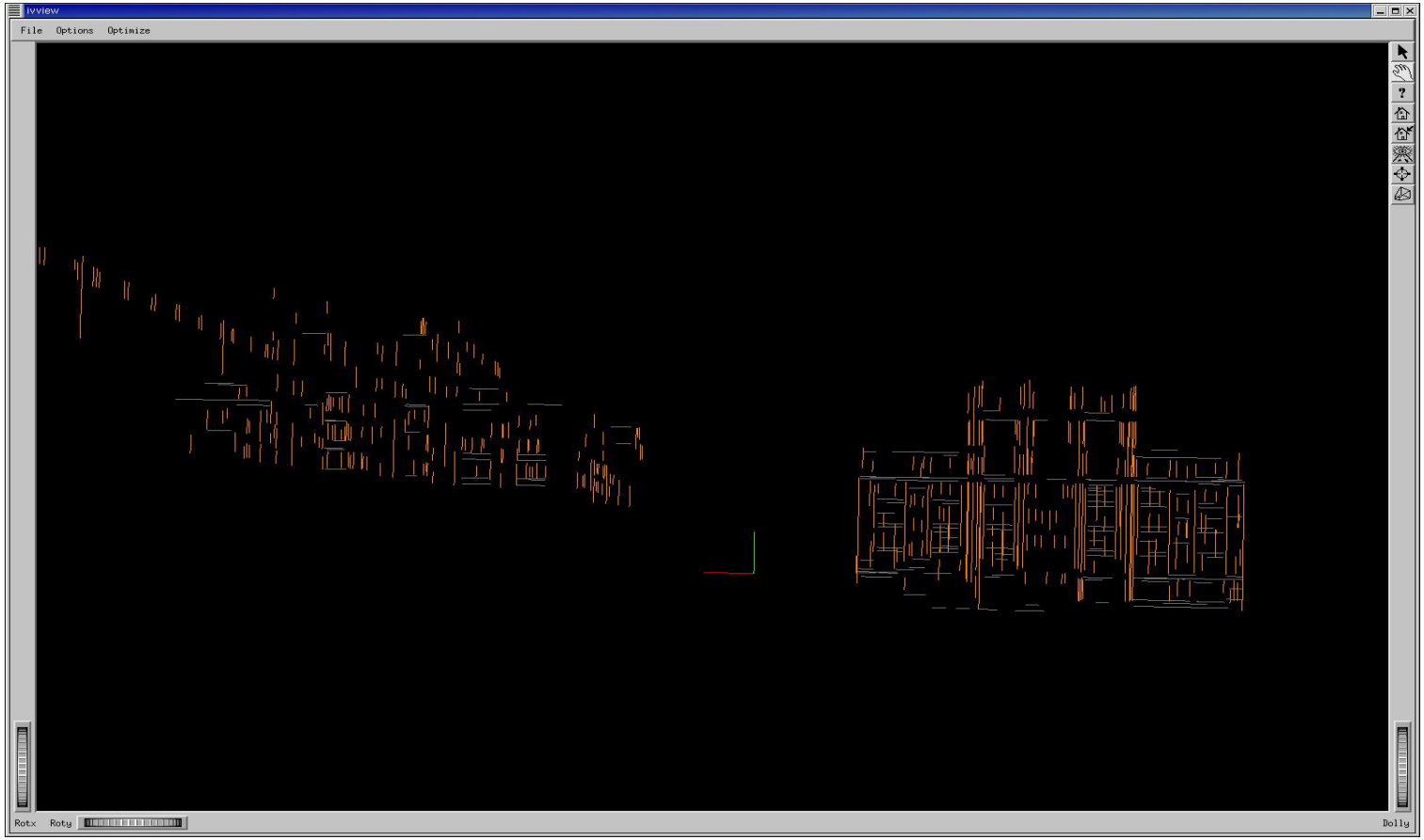
Using Vanishing Point to extract the major direction lines [1]

Extracting the vanishing points and clustering lines



Rotate vanishing points to their corresponding 3D directions (x^,y^,z^)

Matching 2D lines to 3D lines



Results from collecting data (ready for matching).
Left - 2D line set; Right - 3D face line set

Algorithms (still working on it)

To estimate the transformation between 2D and 3D lines. Using that transformation to find some “candidate” matches, then re-compute the transformation by using those candidates, apply it to all lines, find the correspondence.

References

1. ***Geometry and Texture Recovery of Scenes of Large Scale****,* Ioannis Stamos and P. K. Allen, Journal of Computer Vision and Image Understanding, Vol.88, No. 2, pp. 94–118, Nov. 2002

2. ***Automated Feature-Based Range Registration of Urban Scenes of Large Scale***, Ioannis Stamos and Marius Leordeanu, IEEE International Conference of Computer Vision and Pattern Recognition 2003, pp. 555-561, Vol. II, Madison, WI