Nima Soltani (nsoltani@stanford.edu)
Mehmet Mustafa Yilmaz (myilmaz@stanford.edu)

Friday, April 30, 2010

EE368 – Final Project Proposal

Using Augmented-Reality on Planar Surfaces for Previewing Décor Changes

Overview

When consumers look to purchase paintings or posters, an important consideration is whether the painting would go well with their current décor. A customer's choice of a painting would depend on criteria such as its color, size and content, in comparison to their room. The current method is to take a picture of the room to the store to get a good idea of what would go well or to take the painting home to see how well it fits in the room. The proposed project aims to address this uncertainty, giving more confidence to consumers when shopping for paintings while making the process of selling paintings a more efficient one for vendors.

Using the DROID smartphone's image-processing capabilities, this project aims to superimpose the painting on the wall in real-time. The simulated painting should be as realistic as possible. This means the orientation of the painting should be such that it fits the wall, and the size of the painting should be to scale. Also, the simulated painting should be adjusted so its colors appear correctly in the lighting conditions of the room.

While the project covered will place posters and paintings on walls, it could easily be extended to other planar objects such as flat screen TVs or wallpaper. A more challenging extension would be to place 3-D objects in the scene, which would require a rendering of the object at the calculated angle.

Image-Processing Components

For this seemingly simple application, there are many image-processing techniques that need to be employed:

- Planar surface detection and characterization
  - The intended surface must be detected in the image and a mapping must be developed to transform the painting to what it would look like if it was placed on the wall
- Scale estimation
  - While the perspective of the painting is important in making it look natural on the wall, the size the wall must also be estimated so the superimposed painting will be to scale
- Color matching
  - The lighting conditions of the room will be different from the lighting conditions used to take the photo of the painting, so the colors of the painting need to be modified to correspond to the ambient light on the wall

Nima Soltani (nsoltani@stanford.edu)
Mehmet Mustafa Yilmaz (myilmaz@stanford.edu)

Friday, April 30, 2010

Methodology

Placing a virtual 2D object or a texture on a planar surface doesn't explicitly require estimation of the camera pose, if drawing the painting to the exact scale is not a concern. For this type of a problem an affine tracker could be used to detect the rotation, scaling and translation parameters between successive frames. The resulting affine transformation matrix can then be applied to the virtual object to keep its position fixed with respect to the background. The most commonly used affine tracker is pyramidal Lucas-Kanade tracker [1]. The basic idea behind this tracker is to detect some robust features on the scene [2] and to compute the optical flow between successive frames. While the tracker itself gives relatively accurate estimates, in our work we are planning to use an enhanced version proposed in [3] which simply adds an extra step to the tracker to double-check the result of the estimation and refine it if necessary.

While the affine tracker would help us to add 2D textures on the plane, we would need an estimation of the camera pose and the projection matrix if we would like to embed 3D virtual objects in the scene. So, if any time remains, we are thinking of extending the idea above to insert 3D virtual objects on the scene (indoor AR). In order to do that, we need to get an estimate of the projection matrix and the camera pose. Two types of methods are used for this goal. The first family of methods simply tracks natural features on a planar surface and computes the homography between frames after eliminating the outliers using RANSAC [4]. This technique either requires the intrinsic parameters of the camera to be known beforehand or it requires the user to specify four points forming a rectangular shape in the first two frames. One of the major shortcomings of this type of augmented reality methods is that they are exposed to the drifting effect in a long sequence [4]. The other problem is that a sufficient number of features are needed in each frame and the loss of tracked features can stop the process.

An alternative method is to use markers and tags for the augmented reality applications. The advantage of using markers is that they can be localized easily on the picture and since their geometry are known beforehand, they can estimate the projection matrix without needing the knowledge of intrinsic camera parameters [5]. However, an explicit dependency on a marker is a huge shortcoming for this type of methods.

In our project if time permits, as an extension of 2D virtual reality we would like to follow the ideas in [4] to get a 3D virtual reality on planar surfaces. A translational KLT tracker will still be intact and will help us to compute the homography between successive frames together with RANSAC which will help to detect the inliers. For the initial estimate of the projection matrix the ideas in the paper will be followed. After the computation of initial projection matrix, it will be updated in each frame using the homography.

However, in order to realize this method we are going to need some substantial amount of features in each frame, if we can't get the required number of features in an indoor sequence the marker-based method in [5] will be our alternative method. After the computation of projection matrix and intrinsic parameters we are planning to use OpenGL in Android to render the 3D objects.

Nima Soltani (nsoltani@stanford.edu)
Mehmet Mustafa Yilmaz (myilmaz@stanford.edu)

Friday, April 30, 2010

Coding Stages

Project steps:

Augmenting 2D virtual objects on planar surfaces

1. Feature detection based on Good Features to Track
2. Implementation of pyramidal KLT affine tracker with a correction step [3]

Extension to 3D virtual objects on planar surfaces

3. Implementation of RANSAC and Homography computation
4. Implementation of user-interaction step for the computation of initial projection matrix and estimation of intrinsic parameters as in[4]
5. If the algorithm specified in the two previous steps fails, marker-based technique in [5] will be considered as an alternative

References

[1] Baker, S. and Matthews, I. 2002. Lucas-kande 20 years on: A unifying framework: Part 1. Technical Report CMU-RI-TR-02-16, Carnegie Mellon University Robotics Institute.

[2]J. Shi, C. Tomasi, “Good Features to Track”, *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on* , vol., no., pp.593-600, 21-23 Jun 1994

[3]U. Neumann, S. You, “Natural Feature Tracking for Augmented Reality”, *IEEE TRANSACTIONS ON MULTIMEDIA*, VOL. 1, NO. 1, MARCH 1999

[4]G. Simon, M. Berger, “Reconstructing While Registering: A Novel Approach for Markerless Augmented Reality”, *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR’02)*

[5] D. Stricker, G.Klinker and D. Reiners, A fast and robust line-basedoptical tracker for augmented reality applications. In *IWAR ‘98*, pp.31-46, 1998.