

AUGMENTED REALITY 3D ROOM DECORE PREVIEW

Diplav (201430116) and Parv Parkhiya (201430100)
Advisor: Dr. Vineet Gandhi

Abstract—In this project we have developed a Matlab application to add 3D model of chair, lamp, cycle, vase etc in a captured real world image so that it could simulate how would room look like if the object was really present in that environment. For that, we place a physical marker where we have to put the 3D model and acquire the information of placement and orientation of the marker in 3d space, we place our 3D model with same orientation as that of marker. Followed by blending it with captured image.

I. INTRODUCTION

AUGMENTED reality is the integration of digital/virtual information with the user's real world environment. Unlike virtual reality which creates a totally artificial environment from scratch, Augmented Reality relies on understanding the present surrounding and is suppose to enhance the same. This application will help user to visualize the addition of things in environment which were not present earlier. Using which people can choose whether to buy that object or not.

II. MAIN PART

First objective is to robustly detect the marker in the 2d image using its special property. One method to do this is to use one strong conditions but we may end up not detecting marker in some cases. Other method is to use multiple weak conditions followed by intersection or pipeline format. In this paper, we have choose the later method.

Second objective is to determine the orientation of marker in the environment followed by applying same for the 3d model and then blend it with the input image.

III. MARKER CREATION

Marker is a A4 sheet paper with three different color Black, Grey and White such that Black and Grey area are parallel to each and white area surrounding them. Marker was such that it can be identified in any environment with ease. Black and grey area made it easy to find the orientation of the marker. Direction of the marker is from center of black rectangle to center of grey rectangle. White area ensured that inner edges of white and grey surface are preserved irrespective of the color of the surface on which it is kept.

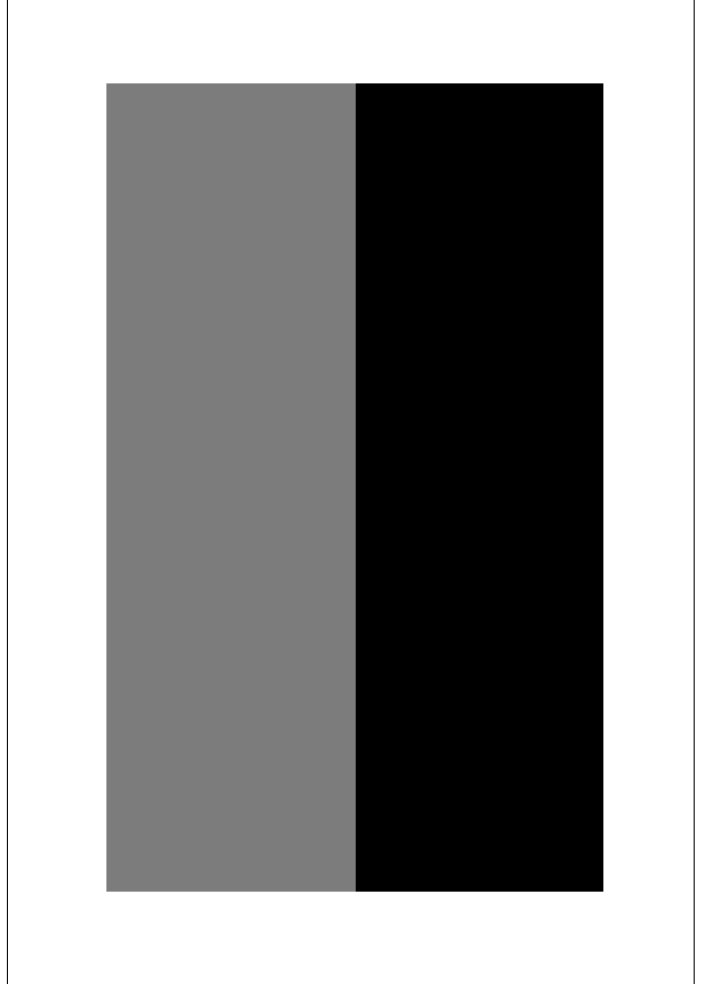


Fig. 1: A4 size Marker (Including White Border)

IV. WHITE BALANCING

It is necessary to make the image invariant of the illumination. So we performed white balancing using Von Kries method. The Main idea of von kries method is to convert the white pixel in the image to desired white pixel and apply the corresponding transformation to each of the pixel in LMS space and convert it back to RGB space. In matrix form this is represented as: Let L_w, M_w, S_w are LMS value corresponding to white pixel of current image.

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} \frac{1}{L_w} & 0 & 0 \\ 0 & \frac{1}{M_w} & 0 \\ 0 & 0 & \frac{1}{S_w} \end{bmatrix} * \begin{bmatrix} L' \\ M' \\ S' \end{bmatrix}$$

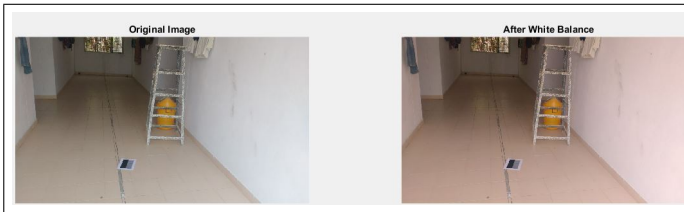


Fig. 2: White Balancing



Fig. 3: Global Thresholding

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.7328 & 0.4296 & -0.1624 \\ -0.7036 & 1.6975 & 0.0061 \\ 0.0030 & 0.0136 & 0.9834 \end{bmatrix} * \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

To convert X,Y,Z to LMS we use above matrix transformation and for RGB to XYZ we use inbuilt Matlab function.

V. INTENSITY THRESHOLDING

After illumination correction Gaussian filter is applied followed by global thresholding. Gauss filter ensures there are no holes inside black region. Main aim of this step is to separate black filled rectangle from grey filled rectangle.

VI. MORPHOLOGICAL OPERATION

Then we utilize characteristic of marker and performed certain morphological operation such as convex shape , area , solidity , Euler number and eccentricity to narrow down the possible candidate for marker. Since our marker is squared shape the solidity is close to 1 and there is no hole in it so its Euler no is 1 and eccentricity is also quite high.

VII. CREATING TEMPLATE AND PATCH

We extracted a template (of thresholded image) centered around the centroid of each of the possible candidate and the size of the patch was equal to the 1.2 times the longest diagonal of the connected component of current candidate. After that we applied edge detection in each of the extracted patch to get template of boundary of candidate. We take 3 times diagonal size of patch at centroid location (of original grey scale image). Edge detection is performed on this patch as well.

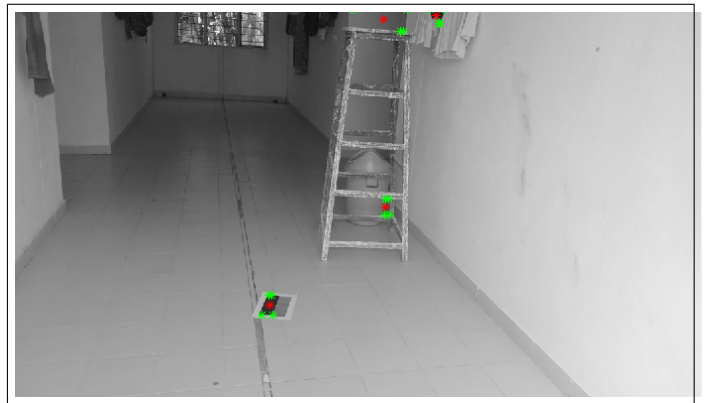


Fig. 4: Candidates after Morphological Operation (in red dots)

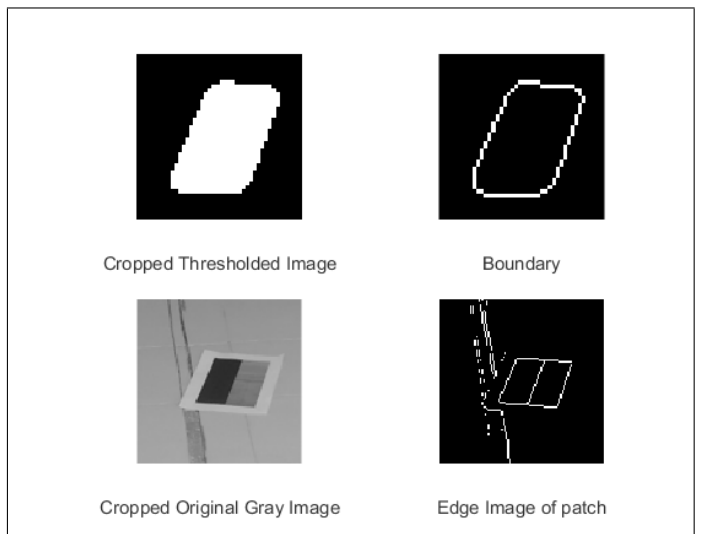


Fig. 5: Boundary Template and Edge image of Patch

VIII. CHAMFER MATCHING

Distance transform is performed on edge image of surrounding of candidate followed by convolution with with respective template the template of marker. On desired patch there should be at least two minima separated by some distance as there are two parallel black and grey patch in the marker image having same boundary. (eg The image shown in figure 6 has two very dark clouds which indicates 2 very good match and third partial match.)

IX. FINALIZING THE LOCATION OF MARKER

We utilize the main intensity property of our marker that it has black and gray patch adjacent to each other and white boundary. We travel in 4 direction from centroid on gray scale image. Up, down, right and left. We mark down values in four direction when there is a jump in intensity value. Orientation of the marker can be anything but at most only two out of four can be gray and that to consecutive. At least one should be gray. We put all 4 values in two beans. White bean should have 2 or 3 and Grey bean should have 1 or 2. Candidate satisfying all this condition are extremely likely to be marker. This also helps us to determine the direction of grey rectangle with respect to black rectangle.

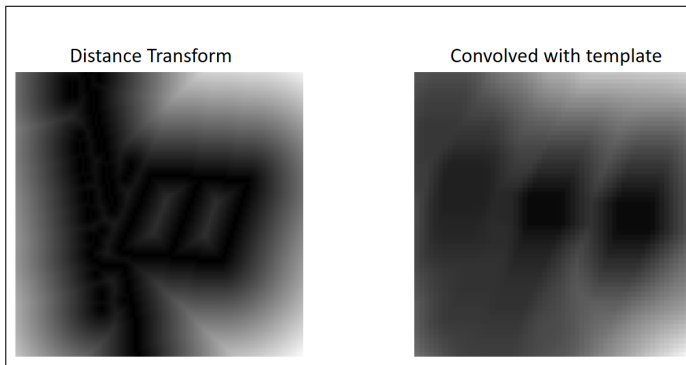


Fig. 6: Chamfer Matching.

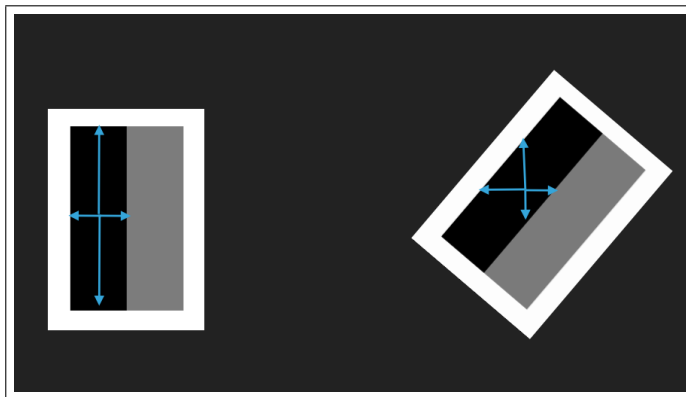


Fig. 7: Checking pixel values: up, down, left, right

X. DETECTING ORIENTATION, ELEVATION AND SCALE OF THE MARKER

Task is to find out the orientation elevation and scale of marker so that we can further import 3D model corresponding to that orientation.

For finding the **orientation** of the connected component corresponding to the marker we find direction of major axis of connected component. (Same was implemented using hough transform on edge image of patch, but major axis direction gives slightly better result.)

Scale of marker is decided by finding the distance of centroid to the extreme coordinates of connected components. The size of the added 3D object would be directly proportional to square root of diagonal length. Scale factor is stored.

Elevation/Altitude will also be proportional to distance from marker, given person is taking photo's from same height. Elevation angle is angle made by observer's line of sight (blue line in figure 8) with surface.

XI. IMPORTING 3D MODEL AND APPLYING TEXTURE

We are importing .obj format 3D model in Matlab which creates (x,y,z) coordinate of vertices and faces of the 3D model. Then we add texture and lighting of the 3D model such that it matches the surrounding environment. (**Code for importing .obj in matlab is not written by us*) We know the orientation of the Marker in -90° to $+90^\circ$ and we also have decided grey location by going left, right, top and down. Either -90° (left) or $+90^\circ$ (right). So we have orientation now

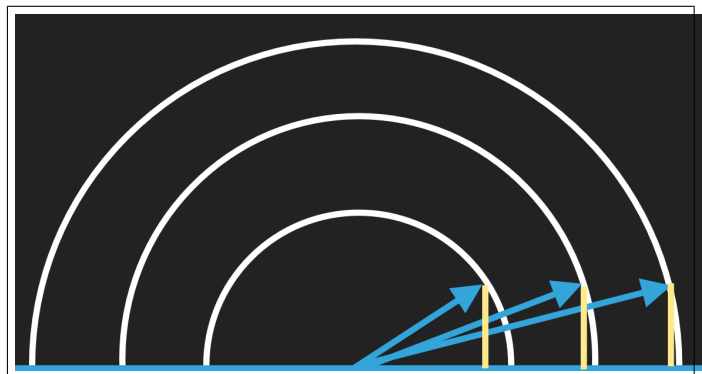


Fig. 8: Marker at origin, Yellow line represents position of person taking picture with Camera, Blue line is line of sight whose angle with origin is inversely proportional to distance from origin.

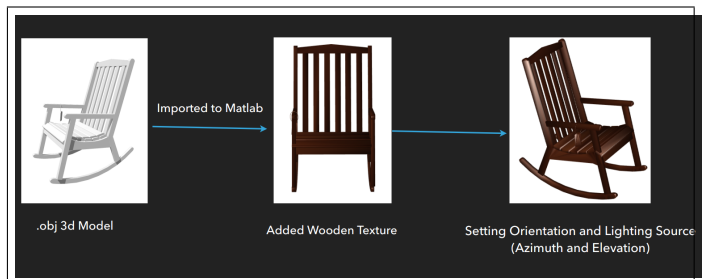


Fig. 9: Working with 3D Model

in -180° to $+180^\circ$. For a 3d model, we now set Azimuth (orientation in 360° surface plane just mentioned before) and altitude based as elevation found earlier. Image is also scaled down as per the scale factor found based on marker size in image. All the proportionality constant are found by running the code on multiple test images.

XII. CREATING MASK AND TARGET IMAGE

Now for blending the image of chair at the position of marker we have to create a mask image and target image. In Target image chair should be placed at the position of marker and in mask image where we have to place chair should be white and rest all place should be black. Mask is eroded to give better result when blended.

XIII. BLENDING

We are using Gaussian/Laplacian Pyramid method to blend original color image and target image using mask image. For

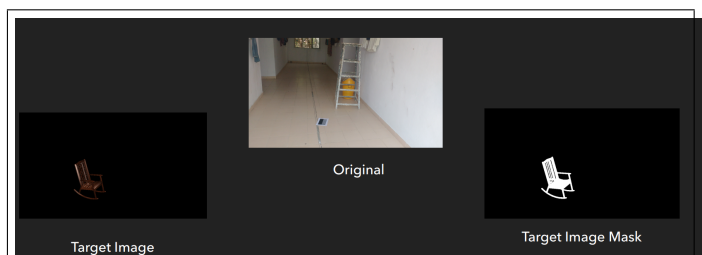


Fig. 10: Target Image and its Eroded Mask Image



Fig. 11: Final Blended Image 1 : Chair Added



Fig. 12: Final Blended Image 2: Chair Added

creating Gaussian and Laplacian Pyramid, we smooth, down sample and stored difference of up sample and non-smooth image. This is done on multiple level in iteration on all three target, source and mask image to get their Laplacian Pyramids. On each level, mask and 1-mask is multiplied with target and source and the resultant of them are added together. From Laplacian pyramid for blended image, we recreate one final blended image.

XIV. EXTENSION/FUTURE SCOPE

We physically add marker in real world and then take picture. Marker gives very high accuracy and uses very less



Fig. 13: Final Blended Image

computation power. But this gives limitations. The next step would be to implement this without using marker so that we can use this on any given image.

Without marker we would require to have some perception of 3D environment from 2D image we have. This would require some form of machine learning, AI.

XV. CONCLUSION

Augmented Reality is the next big thing that everyone in technology world is betting on. The following project barely touches the true potential of AR. AR has many practical applications. Microsoft's HoloLens, Facebook's Oculus Rift and Google's VR headset are trying to do AR in one form or other. Over the years where technology is heavily criticize for creating distance between human and nature/reality, AR is trying to get best of both the world.

REFERENCES

- [1] An Augmented Reality Application for Previewing 3D Dcor Changes by Mai Le, Aaron Zarraga, Kangrong Zhu
- [2] Digital Image Processing Classes (Monsoon 2016), IIIT