REGISTRATION TECHNOLOGIES and THEIR CLASSIFICATION IN AUGMENTED REALITY THE KNOWLEDGE-BASED Registration, COMPUTER VISION-BASED REGISTRATION and TRACKER-BASED REGISTRATION TECHNOLOGY

Prabha Shreeraj Nair, S.Vijayalakshmi and P.Durgadevi

School of Computing Science and Engineering, Galgotias University, Greater Noida, Uttar Pradesh

ABSTRACT

The registration in augmented reality is process which merges virtual objects generated by computer with real world image caught by camera. This paper describes the knowledge-based registration, computer vision-based registration and tracker-based registration technology. This paper mainly focused on tracker-based registration technology in augmented reality. Also described method in tracker- based technology, problem and solution.

1. INTRODUCTION

Why has Augmented Reality become so popular? There are several reasons, some from the past and some recent. First, it's because Augmented Reality is a natural way of exploring 3D objects and data, as it brings virtual objects into the real world where we live. Second, it's because the possibilities of AR are endless, such as information visualization, navigation in real-world environments, advertising, military, emergency services, art, games, architecture, sightseeing, education, entertainment, commerce, performance, translation and so on [1].

Augmented Reality as a technology that has the following five features [2]:

• It combines the real world with computer graphics.
• It provides interaction with objects in real-time.
• It tracks objects in real-time.
• It provides recognition of images or objects.
• It provides real-time context or data.

Registration is the big issue in augmented reality. So we focused in this paper mainly on registration technology classification

DOI : 10.5121/ijcsea.2016.6103

2. CLASSIFICATION OF REGISTRATION TECHNOLOGY

In general the registration technology can be classified into three kinds [3]:

- Knowledge-based registration technology
- Computer vision-based registration technology and
- Tracker-based registration technology

3. KNOWLEDGE-BASED REGISTRATION TECHNOLOGY [3]

Knowledge-based registration technology first proposed by Columbia University through developing augmented reality project in graphics and interface lab of computer science department. It is mainly used in 3D game development purpose. Here trackers are fixed on the equipment with known structure to ensure the position and direction. Some 3D trackers are fixed on the key components to monitor the position and state of the system.

The problems of knowledge-based registration technology are that we must realize the structure of key components in advanced and there are time delay and errors among trackers.

4. COMPUTER VISION-BASED REGISTRATION TECHNOLOGY

Computer vision-based registration technology is easy and high potential in application of AR system. It has very high registration precision which can reach pixel level. Computer based registration technology can be separated into registration based on affine transformation and registration based on camera model (camera calibration).

4.1 Registration based on affine transformation:

Affine transformation theory is introduced into the AR registration and the complex calibration process is translated into the positioning process of datum points in 2D projection plane [3].
Kyriako's N.Kutulakosand James R.Vallino proposed a non-calibration registration method according to affine transformation theory. Gordon has investigated affine transformation algorithm based on stereo image data used for tracking the position and 3D direction of user's viewpoint,which can work in an environment with complexnatural background.

Ming De-lie described a fast affine transformation virtual-real registration method based on numerical background expression by combining image analysis technology. Li Li-jun introduced a virtual-real registration method based on affine transformation feature match. Jeffrey Ho and Ming-Hsuan Yang proposed an affine transformation registration algorithm which uses 2D characteristic point for matching. It does not need optimization and has no disturbance of data noise.

4.2 Camera Model:

Registration based on camera calibration is a common method of computer vision-based registration technology. This method puts some special markers in the real environment firstly, recognizes them bycomputer vision, and then computes the position and direction of camera corresponding to markers, and finally computes the exact position and direction of virtual objects. Camera model can be separated into ***frame differential technique***, ***detection and tracking of feature points*** and calculation of ***camera position information [4].***

4.2.1 Frame Differential Technique

The frame differential technique, based on image processing, is used to test the accurate location of objects using two images captured with a camera. It detects the changed parts using the differences in pixels between the two frames. This method is not difficult and it is appropriate for real-time use at construction sites [4].

4.2.2 Detection And Tracking Of Feature Points

As the features in the captured images must be detected in the same background image in each frame, it is generally more useful to detect corner points that have a large differential coefficient because brightness changes sharply. In this paper we described three methods for Detection and tracking of feature points. These are Shi-Tomasi method, Harris corner detection method, and The Lukas-Kanade Tracker (LKT)[4].

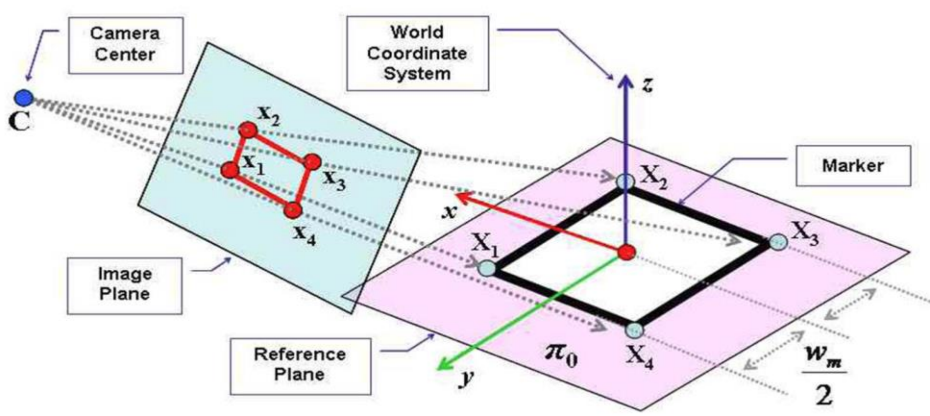

Figure.1. Geometric relationship between features and an image taken by camera [4]

4.2.2.1 Shi-Tomasi Method

Shi-Tomasi method to detect feature points that are useful for tracking, such as corner points. Shi-Tomasi corner detection method determines that a pixel is a good object for tracking if the smaller of the two Eigen values of the hessian matrix is greater than the specified critical value. In order to extract the geometric measurements of the detected feature point, real number-type coordinates must be used instead of whole number-type coordinates. When peaks in an image are searched, they are rarely located at the center of the pixels. To detect them, sub-pixel corner detection is used. The coordinates of peaks that exist between pixels are found by fitting with curves, such as parabolas [4].

4.2.2.2 Harris corner detection method

Harris corner detection method [4] uses the second derivatives of the image brightness. A pixel in the image is regarded as a corner point if all the Eigen values of a hessian matrix consisting of second derivatives at the pixel are large. As second derivative images do not have values at uniform gradient, they are useful for detecting corners.

4.2.2.3 Lukas-Kanade Tracker (LKT)

The Lukas-Kanade Tracker (LKT)[4] is used to track the detected feature points. This is a sparse optical flow method that only uses the local information obtained from a small window that covers predefined pixels; the points to be tracked are specified beforehand. LKT is based on three assumptions: ***brightness constancy, temporal persistence***, and ***spatial coherence [4],***

4.2.2.3.1 Brightness constancy

Brightness constancy assumes that the brightness values of specific continuous pixels are constant in frames of different times. Thus, the partial differential value for the time axis *t*t in expression (1) is 0 shown below [4]. By tracking the areas that have the same brightness value in this way, the speed between consecutive frames can be calculated. *Ix*, *Iy*, and *It* are the partial differentiation results of *x*, *y*, and time axes, respectively. *u* and *v* are the coordinate changes in the *x* and *y* axes, respectively.

$$\frac{\partial I}{\partial x}\left(\frac{\partial x}{\partial t}\right)+\frac{\partial I}{\partial y}\left(\frac{\partial y}{\partial t}\right)+\frac{\partial I}{\partial t}=0$$
$$\frac{\partial I}{\partial x}=I_x,\ \frac{\partial I}{\partial y}=I_y,\ \frac{\partial I}{\partial t}=I_t,\ \frac{\partial x}{\partial t}=u,\ \frac{\partial y}{\partial t}=v \qquad (1)$$

4.2.2.3.2 Temporal Persistence

Temporal persistence assumes that the pixels around a specific pixel that has a movement change over time will have consistent changes of coordinates. As two variables (*u* and *v*) cannot be calculated with one function in expression (1), it is assumed that 25 neighbor pixels have the same change. Then the values of u and v can be calculated using the least square method, as shown in expression (2) shown bellow [4]. The computation speed of LKT is fast because it uses a small local window area under the assumption of temporal persistence that coordinate changes are not large compared to time changes. However, its disadvantage is that it cannot calculate movements larger than the small local window. To address this problem, the Gaussian image pyramid is used. A Gaussian image pyramid is created from the original image, tracking starts from a small stratum, and tracking changes are gradually accumulated to larger strata. Thus, the changes of sharp coordinates can be detected even if a window of limited size is used[5][6].

$$\begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \dots & \dots \\ I_x(p_{25}) & I_y(p_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix} \qquad (2)$$

4.2.3 Camera Position Information

Three-dimensional object coordinates are calculated by back-projection of two-dimensional (2D) coordinates to a three-dimensional (3D) space. The proposed system is preceded by an initialization step in which the 2-D coordinates of the feature points detected from the captured images are converted to the coordinates of a 3D space. As the coordinates detected from the camera input images are 2D, it is assumed that the depth(z-axis) is zero when they are converted to a 3D space. One problem is that if the detected feature pixels are not located on the same plane, errors may be generated. Because of this, the proposed method requires the precondition that it is carried out for relatively flat background images, such as desk top images. Then, the camera position information is calculated through the relationship of the 2D coordinates of the feature points acquired from consecutive frames and the 3-D coordinates acquired from the initialization step. As shown in Figure 2[4], assuming that the homogeneous coordinates of the 3D coordinates $M = (X, Y,Z)$ and the 2D image coordinates $m = (x, y)$ are $\bar{\mathbf{M}} = [XY\,Z1]T$ and $\bar{\mathbf{m}} = [XY\,Z1]T$, the projection relationship between them are defined as expression (3) by the 3X4 camera matrix . λ is the scale variable of the projection matrix $\bar{\mathbf{P}}$, and **R** is the 3X3 matrix by the rotational displacement of the camera. Furthermore, **r** is the Ith column of the matrix **R**, and t is the 3X1 translation vector that signifies the camera movement. In addition, the 3X3 matrix **K** is a non-singular matrix, indicating the camera correction matrix that has the intrinsic parameters of the camera as its elements, and it is generally defined as follows. In the matrix of expression (4), fx and fy are the scale values in the coordinate axis directions of the image, and s is the skew parameter of the image. Furthermore, $(x0, y0)$ is the principal point of the image. To obtain the camera matrix in expression (4), a separate camera correction is generally needed. In this study, the camera matrix was obtained using the camera correction method proposed by Zhang

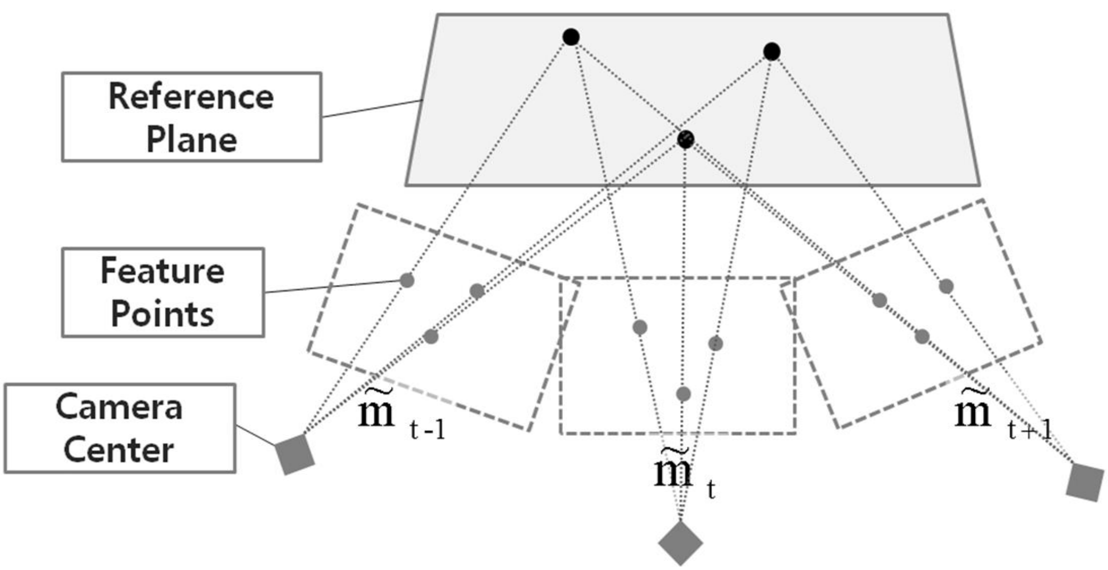

Figure 2. Projective geometry of input image sequences.

$$\widetilde{\mathbf{m}} = \lambda \widetilde{\mathbf{P}} \widetilde{\mathbf{M}} = \lambda \mathbf{K}[\mathbf{R}|\mathbf{t}] \widetilde{\mathbf{M}} = \lambda \mathbf{K}[\mathbf{r_1 r_2 r_3}|\mathbf{t}] \widetilde{\mathbf{M}} \quad (3)$$

$$\mathbf{K} = \begin{bmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

The camera correction method of Zhang calculates the Image of Absolute Conic (IAC) ω, which is projected to the images using the invariance of isometric transformation, which is one of the characteristics of absolute points on an infinite plane. Then the intrinsic parameter matrix of the camera is obtained from the relationship ω−**1** = **KKT**. Therefore, in order to implement Zhangs method, three or more images for the same plane, which have different directions and locations, are required.

5. TRACKER-BASED REGISTRATION TECHNOLOGY

Tracking technologies may be grouped into three categories [7]:

- active-target
- passive-target and
- inertial

5.1 Active-target

Active-target systems incorporate powered signal emitters and sensors placed in a prepared and calibrated environment. Examples of such systems use magnetic, optical, radio, and acoustic signals.

Limitation: The signal-sensing range as well as man-made and natural sources of interference limit active - target systems

5.2 Passive-target

Passive-target systems use ambient or naturally occurring signals. Examples include compasses sensing the Earth’s field and vision systems sensing intentionally placed fiducially (*e.g.,* circles, squares) or natural features.

Limitation: Passive-target systems are also subject to signal degradation, for example poor lighting or proximity to steel in buildings can defeat vision and compass systems.

5.3 Inertial

Inertial systems are completely self-contained, sensing physical phenomena created by linear acceleration and angular motion.

Limitation: Inertial sensors measure acceleration or motion rates, so their signals must be integrated to produce position or orientation. Noise, calibration error, and the gravity field impart errors on these signals, producing accumulated position and orientation drift. Position requires double integration of linear acceleration, so the accumulation of position drift grows as the square of elapsed time. Orientation only requires a single integration of rotation rate, so the drift accumulates linearly with elapsed time.

6. PROPOSED METHOD FOR TRACKER-BASED REGISTRATION

6.1 Setup

Here a planar canvas is used that has specific shape. The shape is used as the cue for the vision-based tracking. The system configuration is illustrated in Figure 3[8]. We also use a binocular video see through HMD which enables users to perceive depth. The HMD is connected to a video capture card ,which captures input videos from the cameras built into the HMD. The NVIDIA GeForce GTS 250 graphics processor is used for image processing. The positions and orientations of the HMD and Brush Device are tracked using Polhemus LIBERTY, a six-DOF tracking system that uses magnetic sensors. A transmitter is also used as a reference point for the sensors. The brush device is connected to the main PC through an input/output (I/O) box. The I/O box retrieves data from the devices and sends them to the main pc[8].

We used OpenGL and the OpenGL Utility Toolkit (GLUT) for the graphics API. In creating the MR space, we first set the videos captured by the Osprey-440 as the background and then create a virtual viewing point in OpenGL by obtaining the position and orientation of the HMD from Polhemus LIBERTY. In done so, users feel as if they are manipulating virtual objects in the real world [8].

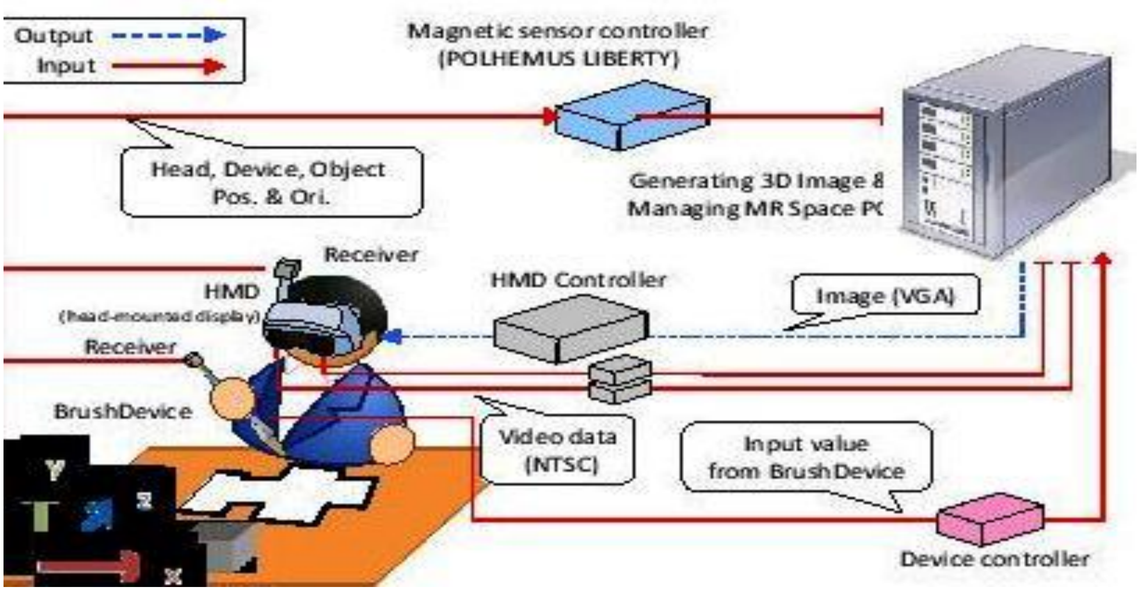

Figure 3. System configuration.

6.2 Contour tracker

6.2.1 Descriptor

The first step in our method is extracting the region of the canvas. The popular and recent method such as MSER can be applied. We also can use the simple color segmentation for simple outline as illustrated in Figure 4[8].

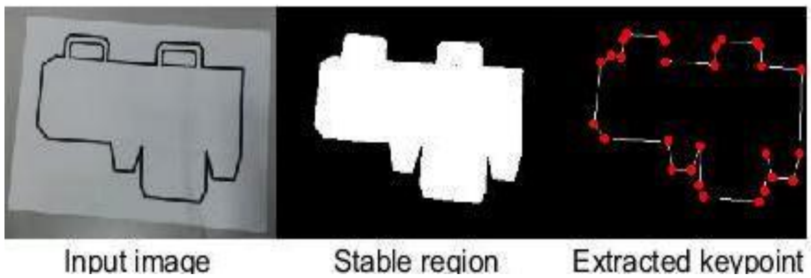

Figure 4: Keypoint extraction

The next step is extracting the outline by using contour estimation. The outline that may contain many points is simplified using DP polygon simplification. The remaining points are used as keypoints. The next step is computing the descriptor *r* (relevance measure) that is computed using three consequent points in the simplified polygon [8]. *r* is defined as

$$\frac{\theta l_1 l_2}{{l_1}^2 + {l_2}^2} \qquad (1)$$

where l_1 and l_2 are the length of two connected segments (lines) and θ is the angle between where l_1 and l_2 . *r* depends on θ and the ratio of both segments. We assume these properties will not change drastically due to scale and rotation changes. More specifically, θ will not change due to rotation and the ratio will not change due to scale changes. However, these properties may change due to perspective change. We handle the perspective change by tracking step that will be explained later on the registration method subsection. Computing descriptors of the shape is done both off-line and on-line. In off-line process, the descriptors are stored in the database whereas in on-line process, the descriptors of unknown shape are matched with the descriptors in the database (see Figure5)[8]

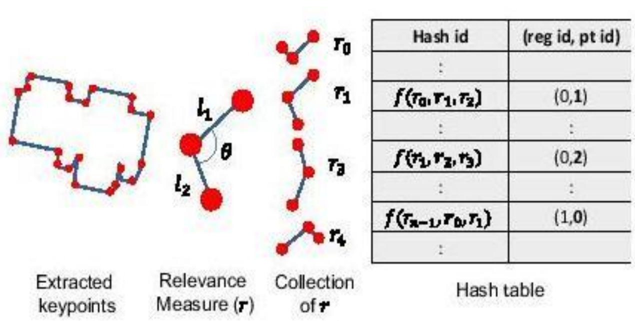

Figure5: descriptor matching

6.2.2 Registration

During the shapes registration, the keypoints of unidentified shape are extracted. The sequences of relevance measures are then computed and the corresponding tuple (region or shape id, keypoint id) is looked up in the hash table (see Figure 5). Since the hash table is many-to-one relationship, in order to get the matched region and keypoint, histogram matching (voting) is performed. This process yields keypoints correspondences between a shape captured in the camera and a shape in the database. We choose three neighboring relevance measure values to represent a keypoint of a shape. In this case, one keypoint is actually described by its four neighbor's keypoints. For example *r0, r1, r2* represents a keypoint with id=1. In our implementation, we increase the number of relevance measure to four in order to create distinctive representation of a keypoint for example *r0, r1, r2, r3* for keypoint with id=1. During runtime, when a shape is matched to one in the database, sequences of relevance measure of the matched shape are added to the hash table so that the registration becomes robust against perspective change.

6.2.3 Pose estimation

The camera pose is estimated using homography that is calculated using at least four keypoints correspondences as the result of the shape registration. The outliers from the keypoints are removed using the inverse homography. The camera pose is then optimized using Levenberg-Marquardt [9] by minimizing the re-projection error that is the distance between the projected keypoints from the shape database and the extracted keypoints in the captured frames. The camera pose is then refined by considering the keypoints correspondence to the detected shape in previous frame. These two optimizations produce a stable camera pose[8].

6.2.4 Unifying the coordinate system

One issue to switch from a magnetic sensor to a visual tracker is the unification of coordinate system. Since in the vision-based tracking, the coordinate system is independent from the sensor coordinate system, the transformation from the camera coordinate system into the sensor coordinate system is necessary. This unification is also required in order to enable the painting interaction since the brush and eraser device are located in the sensor coordinate system.

The unification is quite straightforward since the tracking is done using the camera attached in the HMD. Note that a magnetic sensor is also attached in the HMD which make the unification becomes simple. Therefore, the unification is done by computing the transformation matrix for any object in the camera coordinate system. The transformation matrix is computed by multiplying the view matrix of the HMD by the rotation and translation matrix retrieved using homography as the result of the shape registration. As a result, the virtual canvas can be painted using the brush device that has a magnetic sensor on it. The result of the painting can be seen in Figure 6[8].

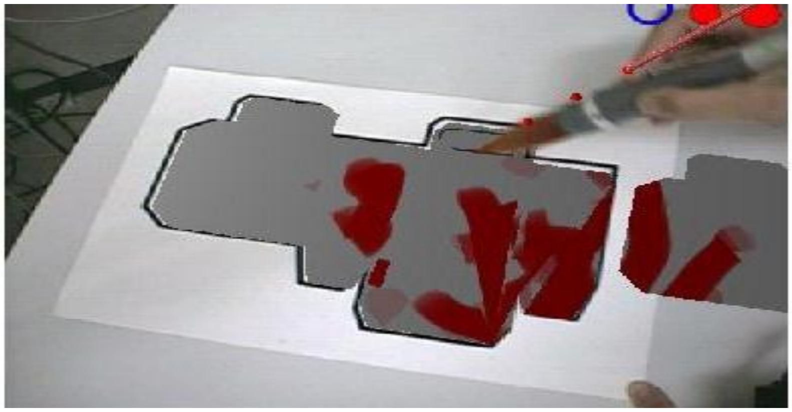

Figure 6. Painting result on a canvas that has a specific shape**.**

7. CONCLUSION

In this paper we proposed tree registration technology in augmented reality. These technics are Knowledge-based registration technology, computer vision-based registration technology, and tracker-based registration technology. In computer vision-based technology we described how differentiated the frames and how detect and tracking the feature points in image registration purpose. In tracker based technology we described three type of tracking technics such as active-target, passive-target, and inertial and our next plan to use their hybrid concept in augmented reality. We also proposed setup and contour tracker method where we used the asymmetry 2D object as the tracking marker as the canvas, but in the future, we are planning to extend our method to 3D objects. Using 3D objects as the marker, it is possible to paint on any object that of course must be detectable objects.

REFERENCES

[1] Wikipedia. Augmented reality. Available at:http://en.wikipedia.org/wiki/augmented_reality (Accessed 20 February 2012) .

[2] Augmented Reality Technology and Art: Analysis and Visualization of Evolving Conceptual Models. 1550-6037/12 $26.00 © 2012 IEEE DOI 10.1109/IV.2012.77

[3] Li Yi-bo, Kang Shao-peng, Qiao Zhi-hua, Zhu Qiong "Development Actuality and Application of Registration technology in Augmented Reality" 978-0-7695-3311-7/08 $25.00©2008 IEEE DOI 10.1109/ ISCID .2008. 120 .

[4] Jae-Young Lee Jong-Soo Choi, and Oh-Seong Kwon Chan-Sik Park:" A study on Construction Defect Management Using augmented reality technology" 978-1-4673-1401-5/12/$31.00 ©2012 IEEE

[5] J. Barron, N. Thacker, "Computing 2D and 3D Optical Flow," Tina-Vision, 2005.

[6] G. Bradski, A. Kaehler, "Learning Opencv: Computer Vision with the Opencv Library," O'REILLY, 2008.

[7] Suya you, Ulrich Neumann, Ronald Azuma:" Hybrid Inertial and Vision Tracking for Augmented Reality Registration"

[8] Sandy Martedi1*, Maki Sugimoto, Hideo Saito, Mai Otsuki2*, Asako Kimura, Fumihisa Shibata:" A Tracking Method for 2D canvas in MR-based interactive painting system" 978-1-4799-3211-5/13 $31.00 © 2013 IEEE DOI 10.1109/SITIS.2013.128

[9] Lourakis, M. "levmar: Levenberg-marquardt nonlinear least squares algorithms in C/C++." [Web page] http://www.ics.forth.gr/~lourakis/levmar/, Jul.2004.[Accessed on 23 Aug. 2013.].

AUTHORS

Prabha Shreeraj Nair was born in chhattisgarh, India, in 1973. She received the B.E. Degree in Computer Technology from Nagpur University, in 1996 and Masters in Computer Science and Engineering from Kakatiya University, in 2007.She has been working as an Assistant Professor, Galgotias University, Greater Noida, Uttar Pradesh, India she has 18 years of teaching experience. She is engaged in research on Software Engineering, Augmented reality.

Vijayalakshmi S was born in the year 1975. She received the B.Sc. degree in Computer Science from Bharathidasan University, Trichirapalli, India in 1995, the MCA degree from the same University in 1998 and the M.phil. degree from the same University in the year 2006. She received her doctorate in 2013. She has been working as an Assistant Professor (grade-III), Galgotias University, Greater Noida, Uttar Pradesh, India she has 17 years of teaching experience and 10 years of research experience. She has published many papers in the area of image processing especially in medical imaging