Vision-Based Object Registration for Real-Time Image Overlay

Michihiro Uenohara[1] and Takeo Kanade[2]

[1] Toshiba R&D Center, Kawasaki, Japan
mue@mel.uki.rdc.toshiba.co.jp
[2] The Robotics Institute, Carnegie Mellon University,
Pittsburgh, PA 15213, U.S.A.
tk@cs.cmu.edu

Abstract- This paper presents computer vision based techniques for object registration, real-time tracking, and image overlay. The capability can be used to superimpose registered images such as those from CT or MRI onto a video image of a patient's body. Real-time object registration enables an image to be overlaid consistently onto objects even while the object or the viewer is moving. The video image of a patient's body is used as input for object registration. Reliable real-time object registration at frame rate (30 Hz) is realized by a combination of techniques, including template matching based feature detection, feature correspondence by geometric constraints, and pose calculation of objects from feature positions in the image. Two types of image overlay systems are presented. The first one registers objects in the image and projects pre-operative model data onto a raw camera image. The other computes the position of image overlay directly from 2D feature positions without any prior models. With the techniques developed in this paper, interactive video, which transmits images of a patient to the expert and sends them back with some image overlay, can be realized.

Category - on line tracking of patient or organ motion

1 Introduction

Due to the significant improvements in computer vision techniques in recent years[1][2][3], real-time and interactive imaging of complex biomedical systems have become a great priority within medicine.

One major application is to integrate the precise pre-operative information currently found within CT and MRI into intra-operative surgical procedures. The display of correctly registered medical images on a patient provides a new method of surgical guidance which can enhance human perception and skills [4]. Most previous methods of registration, however, are either off-line or assume that the patient does not move during the surgery. Real-time computer vision techniques for object registration can realize a new type of non-intrusive image overlay without using special positioning devices. The overlaid image can be kept at the same position of the patient in the image as if the overlay were physically attached to the patient. The overlay remains fixed to the patient even with movement of the patient and the camera.

Interactive video is another application. In telemedicine, rural surgeons would send patient records, X-rays and CT scans to an expert surgeon at a center who would use them to plan the operation on a surgical simulator. The expert would send the surgical plan to the remote doctor or medic and guide him through the surgery. The interactive video, which transmits images of a patient to the expert and sends them back with some image overlay, enables the expert to guide surgeons as if the expert were across the operating table from him. It could keep showing the surgeon the place on the patient's body to which the expert points, while the patient and the camera are moving in three dimensions.

This paper presents object registration and tracking techniques appropriate for the realization of real-time image overlay. Two image overlay systems are shown. The first one registers objects in the image and projects pre-operative model data onto a raw camera image. The other computes the position of image overlay directly from 2D feature positions without any prior models.

2 Object Registration for Image Overlay

Object registration is required to superimpose pre-operative model data onto a raw camera image accurately at the right place. Registration is the process of computing the object pose parameters in camera-centered coordinates. Camera-centered coordinates have the origin at the optical center of the camera with which raw camera image is taken. When the pose of the object has been computed, the remaining step is to generate an image of prestored data, such as a pre-operative bone model derived from CT, appropriately projected onto the image plane, and add it to a raw camera image.

Most previous work on object registration in medicine utilizes 3D image data (as from a scanning laser rangefinder) and searches their best match with 3D model data sets by using a least squares minimization of distances between data sets [4][5].

In the computer vision area, a few methods have been developed for visual tracking of known three-dimensional objects using only 2D images. They locate and track predefined features, such as edges and corners, on the object in the images, and use these measurements to calculate the estimate of position and orientation of the object [6][7][8][9]. In the case where we have object-centered coordinates of features in the models, the problem is formulated as an inverse problem to solve the nonlinear relationship between object pose and feature positions in the image. This problem can be solved by recursive methods.

The system described in this paper utilizes 2D intensity images and detects feature points by template matching. The change of intensity patterns due to view change is compensated by skewing reference images with computed object pose parameters. The system has been implemented on multiple DSPs and performs tracking at the frame rate.

3 Initial Recognition of the Object

For human interface systems, it is reasonable to assume that users can roughly locate objects at the start of the execution of the system. The system, for example, can superimpose the desired position and orientation of the object onto the raw camera image. Users can set the pose of the object as indicated by moving either the object or the camera (Fig.1).

When the rough pose of an object is set by the user, the initial recognition is carried out to precisely calculate the object location. For this purpose, reference images of feature points are precaptured in various conditions of illumination while objects are set to be the predefined pose. “Template space”, which is the vector subspace involving not only the discrete reference images but also their interpolation, is computed. A set of images in the template space are considered as a template. The intensity pattern most similar to the input image in the template space is found and its normalized correlation to the input image is computed at each point in the search area. The point with the highest score is chosen, and it is recognized as a feature point when the highest score is over a threshold. When all the feature points are successfully found, the object pose is calculated and the system goes to the tracking phase.

3.1 Generation of Template Space

Precaptured reference images differ slightly from each other and are highly correlated. Therefore, the image vector subspace required for their effective representation can be defined by a small number of eigenvectors or eigenimages. The eigenimages which best account for the distribution of reference images can be derived by Karhunen-Loeve expansion [10][11][12][13].

Let the set of reference images be $\boldsymbol{x}_i$, $i=1,2,...,P$ which are represented as vectors of dimension N^2, describing N by N image templates. The vectors $\boldsymbol{e}_j$ and scalars λ_j are the eigenvectors and eigenvalues, respectively, of the covariance matrix:

Fig.1 Wire-frame overlay at initial recognition.

$$A = \frac{1}{P}\sum_{i=1}^{P}(x_i - c)(x_i - c)^T \tag{1}$$

where $c = \frac{1}{P}\sum_{i=1}^{P} x_i$ is the average image vector. We obtain optimal approximation of reference images by selecting eigenvectors in decreasing order of magnitude of eigenvalues and representing each reference image by a linear combination of first K largest eigenvectors as

$$x_i \approx c + \sum_{j=1}^{K} p_{ij} e_j \tag{2}$$

where $p_i = [e_1, e_2, \ldots, e_K]^T (x_i - c)$.

The major K eigenvectors and the average image vector c span a (K+1)-dimensional subspace ("template space") of all possible images, and a set of images in the subspace is considered as a template to be recognized. The dimension along the average image vector is added to make the recognition insensitive to the magnitude of image patterns. A set of reference images in the template space x are therefore expressed in terms of a linear combination of a finite set of orthonormal basis:

$$x = \sum_{j=0}^{K} p_j e_j \tag{3}$$

where

$$e_0 = \left(c - \sum_{j=1}^{K} p_j^c e_j\right) \Big/ \left\| c - \sum_{j=1}^{K} p_j^c e_j \right\| \tag{4}$$

and $p^c = [e_1 e_2 \ldots e_K]^T c$

3.2 Normalized Correlation in the Template Space

The input image is evaluated at each location how it fits the template by extracting the region and finding the most similar pattern in the template space and computing the normalized correlation between them. The most similar pattern in the template space is the projection of the extracted region vector y into the template space (Fig. 2). Its normalized correlation to the vector y is the largest. The normalized correlation between the vector y and a reference vector x is given by $C(y, x) = x^T y / \|x\|\|y\|$. Replacing the reference image vector x with the projection $\tilde{x} = \sum_{j=0}^{K}\left(e_j^T y\right) e_j$ yields

$$C(y, \tilde{x}) = \frac{\left(\sqrt{\sum_{j=0}^{K}\left(e^T{}_j y\right)^2}\right)}{\|y\|} \tag{5}$$

The normalized correlation score above is the measure of similarity considering not only prestored discrete P reference images but also their interpolation. This makes

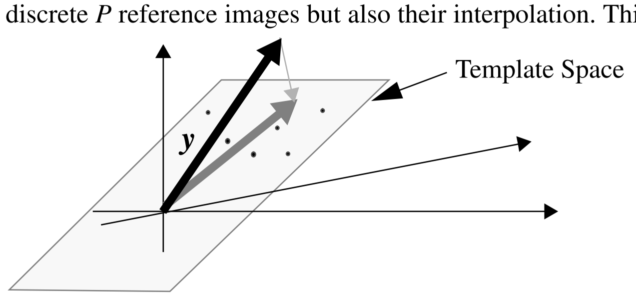


Fig.2 Template space and the projection of input image.

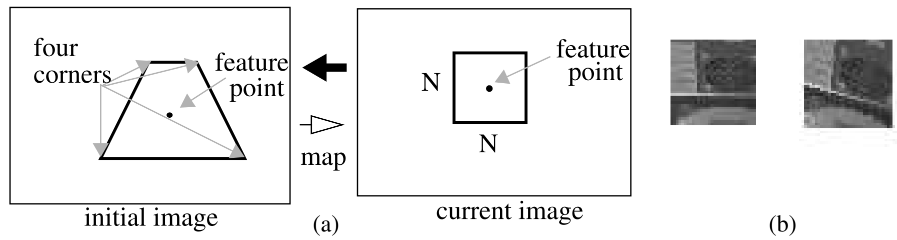


Fig.3 (a) Change of appearance of feature points. (b) Example of skewed image.

the system robust against the variation of illumination. The computation cost is greatly reduced since the original normalized correlation requires $N^2(P+1)$ operations for P reference images while (5) requires $N^2(K+2)$ operations where K can be much smaller than P.

4 Tracking of Features

Feature points that are easy to track are selected before execution, and their positions in object coordinates are given as part of the object models. When the object is recognized and the pose is calculated at the initial recognition phase, feature points are projected onto the image plane with the computed pose. A small region around each feature point is extracted as the reference image for the subsequent visual tracking. For visual tracking, normalized correlation to reference images is computed at every point in the small search areas. The positions with the best normalized correlation scores are determined as the positions of feature points in the image.

The appearance of a feature point varies during tracking due to view change. Skewing reference images using the object pose information in every cycle during tracking can compensate this effect. Reference images are small square windows of N by N pixels around feature points. Under projective transformation, straight lines are projected to straight lines and intersections are preserved in any view change. Generation of the skewed reference image is illustrated in Fig.3: first compute the skewed rectangle in the initial image which corresponds to the small square window in the current image, and then map the pixels of the initial image into the square window. The skewed rectangle in the initial image is computed from feature positions in the current image, object pose in the current and initial images, and surface orientation around feature points. Surface patches around the feature points are approximated as planar, and those equations are given as part of object models.

5 Feature Correspondence

Some features may be missed or mismatched during tracking. We need to select only those feature points which are successfully tracked. The value of normalized correlation itself can be used as the criterion, for the degree of matching at the image level. However, to cope with illumination changes and other difficulties, geometric constraints between feature points are quite useful.

5.1 Geometric Invariants

Geometric invariants [14, 15], popular in object recognition as useful descriptors of objects, are properties in the image that stay invariant under some transformation. Five coplanar points have the familiar cross ratio as their invariant. The cross-ratios I_1 and I_2 of four areas of triangles are invariant under projective transformation:

$$I_1 = \frac{S_{423}S_{125}}{S_{124}S_{523}} \qquad I_2 = \frac{S_{143}S_{125}}{S_{124}S_{153}} \tag{6}$$

where S_{ijk} is the area of a triangle with three points, i, j, and k. These values remain the same over view changes.

Another candidate is affine algebraic moment invariants [15], which are applicable to curved objects because they do not require feature points to be coplanar. If the geometric invariant values computed for tracked feature points change, it indicates that some of feature points are misrecognized.

5.2 Sensitivity of Invariants

Due to observation errors in tracking feature positions (typically 0.5 pixel), invariants vary. The sensitivity of invariants is also dependent on configuration of feature points. This makes it difficult to use constant thresholds to judge whether or not invariants are violated and thus tracking has failed. We adjust thresholds by the standard deviation of each invariant. Assume that observation errors of each feature position have a zero-means Gaussian distribution with covariance Λ_p. Invariants then have the distribution with a expected variance σ_I^2 [16]:

$$\sigma_I^2 = \boldsymbol{J}\Lambda_p\boldsymbol{J}^T \tag{7}$$

where $\boldsymbol{J}$ is the Jacobian matrix $[\partial I/\partial \boldsymbol{x}]$ of $I = I(\boldsymbol{x})$ that relates x-y coordinates of feature points to the invariant. The threshold for each invariant is set to the standard deviation of invariants multiplied by some constant c.

Feature correspondence is carried out as follows, when cross ratios of areas of five coplanar points are used as invariants. The values I_1, I_2 in equation (6) are computed for all combinations of five points out of all feature points. The combinations whose variations of I_1 and I_2 from initial values are both below their thresholds are selected. If there is more than one combination of five feature points that satisfy the condition, we select the five feature points that produce the minimum of the maximum variation of I_1 and I_2 divided by the corresponding standard deviation. The five feature points thus selected are used to calculate the object pose.

6 Direct Computation of Image Overlay

The image invariant values help us to compute the position of points without registration. Since they are invariant to any view change, they enable us to calculate the position of one of the points from the other 2D feature positions in the current image with

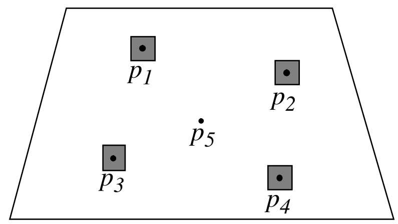


Fig.4 five coplanar points.

the value of invariants. That means that we can track a "virtual" feature point which may not have any particular pattern. We use a certain number of real trackable feature points around it, track them, and calculate the position of the virtual feature point by means of the invariant.

For example, referring to Fig. 4, with five coplanar points, the fifth point p_5 on the surface can be tracked. The values of two invariants I_1, I_2 in the first frame are calculated from the positions of p_5 and four other coplanar points. They remain constant over frames. Since they are functions of the positions of the five points, the unknown parameters are (x_5, y_5), x-y coordinates of the fifth point p_5 when we keep tracking four points p_1, p_2, p_3, p_4. We have two invariants I_1, I_2, so that we can calculate the position of the p_5 by solving these two linear equations in terms of (x_5, y_5).

7 Experimental System for Real-Time Image Overlay

We will present a real-time image overlay system. It is used for three example tasks. The first example is the tracking of a desktop PC and image overlay of the image of an I/O board. The second is the tracking of a phantom leg with some marks on it and the overlay of a bone model on its view. The third is the overlay of a virtual pin onto a leg model.

The system is implemented on multiple TMS320C40 (C40), Texas Instruments digital signal processors. We use low latency vision hardware developed at CMU[13] which has a digitizer with the high-speed data link. Image data are transferred through this high-speed data link into C40 communication ports, and then transferred to the local memory of the processor and other processors' communication ports by DMA in order to minimize the delay.

7.1 Overlay of an Image on a PC

Visual tracking of objects without attaching specific marks is tested on a desktop PC. The hypothetical task is to consistently overlay the word "Board" to indicate the I/O board that the repair person should service. At the beginning of the operation, the system displays a wire frame of the PC on the monitor and requires a user to move the camera so that the PC and the wire frame are approximately aligned (Fig.1). When the PC is roughly aligned to the wire frame, the system recognizes it, "latches" onto it, and starts tracking it. Initial recognition is executed by template matching of three regions

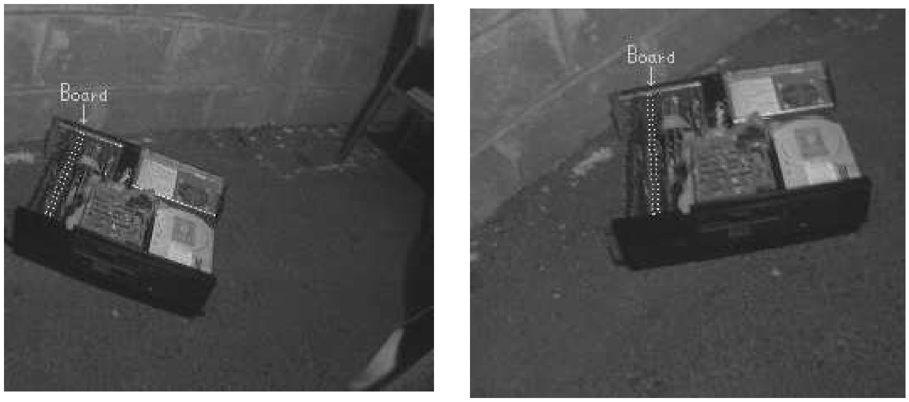


Fig.5 Real-time overlay of information (“Board”) on a desktop PC image.

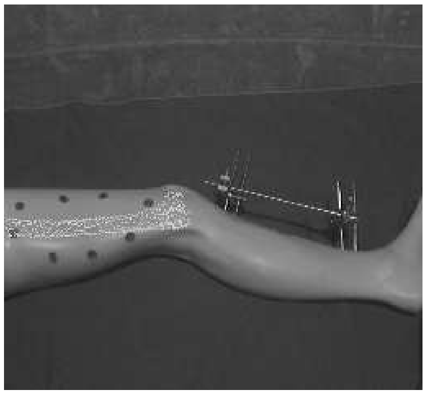
on the PC. Eleven images of the PC under different illumination conditions had been precaptured. Three 32 x 32 regions are extracted from each image as reference images, and template spaces are generated by four major eigenvectors.

When the three regions are successfully found, the pose of the PC is calculated from three feature positions in the image by Newton’s method [8]. Feature points for tracking are projected onto the image with the computed pose. Small windows of size 16 x 12 of eight feature points are extracted from the image and are then used as reference images in the tracking phase (Fig. 5). In the tracking phase, the maximum normalized correlation of the extracted window image is searched for in 14 x 14 a region whose center position of the search area is usually set to the feature position in the last frame. In the case where the feature point is missed in the last frame, the projected feature position in the image, computed using the pose of the PC in the last frame, is used to define the center position of the search area. This allows for the recovery of tracking.

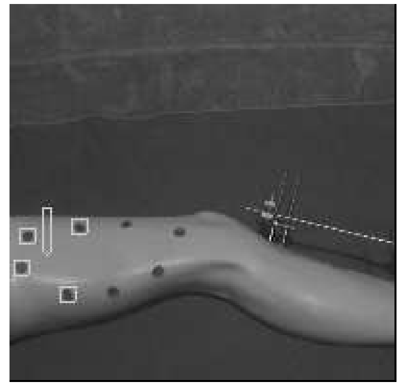
The tracking results of eight features are checked by calculating cross ratios of areas of five coplanar points. The best five points which have minimum change are selected, and the pose of the PC is computed with these five points. Feature points whose normalized correlation value is less than 0.7 are rejected before the computation of invariants. A check by geometric invariants, combined with the normalized correlation peak score, makes the tracking much more robust. The system can track the PC and superimpose the information as the camera and the PC translates and rotates in 3D, even when up to three feature points are occluded by other objects such as human hands. The system operates at the video frame rate (30 Hz). Three C40s are used in parallel: two C40s for tracking eight feature points and one for checking tracking results, pose calculation, and image overlay.

7.2 Overlay of an Image of a Bone onto a Leg

The task is to overlay a bone surface model derived from CT data on a phantom leg (Fig. 6(a)). Since there are no complex features around marks, the normalized correla-



(a)



(b)

Fig.6 (a) Overlay of a bone on a leg. (b) Overlay of a virtual pin on a leg.

tion gives us reliable matching and we did not use invariants for feature selection. As in the PC case, the overlaid image of the bone appears to remain attached to the leg despite three-dimensional motions of the leg, camera, and certain occlusions.

7.3 Pin Overlay without Models

The last task is to overlay a virtual pin onto a phantom leg (Fig.6 (b)). In the case of interactive video, when experts touch the screen to indicate the specific position of patients' bodies, the touched position is transferred to the remote site as the given pin tip position and the virtual pin is superimposed on the image of patients. Surgeons can recognize the place on patients to which the experts point, even after some motion of the patients' bodies.

The initial pin tip position is given in advance in this experiment. An overlaid image of the pin remains fixed onto the image of the leg over some motion of the leg. The tip of the pin is supposed to be attached on the leg. Four marks around the pin tip are kept tracking and the position of the pin tip is computed directly from these 2D mark positions as described before.

8 Conclusions

This paper has presented an image overlay system that uses real-time object registration and tracking. The system utilizes intensity images and detects feature points by template matching by normalized correlation. The change of intensity patterns due to view change is compensated by skewing reference images with computed object pose. Use of geometric invariants increases robustness in the feature correspondence between features in the image and the model. Real-time tracking of objects and overlaying image at frame rate (30 Hz) is achieved by the multiple DSP system with low latency vision hardware. It should be noted that no explicit model of object or display position was used in the experiment of overlaying a virtual pin on a phantom leg. This is made possible by using geometric invariants. Capable, real-time image overlay will have a broad application to interactive communication between rural surgeons and experts, which helps the delivery of expert care to geographically or socioeconomically isolated areas.

References

1. E.R. John, L.S. Prichep, J. Fridman and P. Easton, Neurometrics: computer-assisted differential diagnosis of brain disfunction, Science Vol. 239, pp.162-169 (1988).
2. L.S. Hibbard, J.S. McGlone, D.W. Davis, R.A. Hawkins, Three-Dimensional Representation and Analysis of Brain Energy Metabolism, Science, Vol. 236, pp.1641-1646 (1987).
3. C. Nastar and N. Ayache, Non-Rigid Analysis in Medical Images: a Physically Based Approach, Proc. 13th Int. Conf. on Information Processing in Medical Imaging, Berlin, Germany, pp. 17-32 (1993).
4. W.E.L. Grimson, T. Lozano-Perez, W.M. Wells , G.J. Ettinger, S.J. White, R. Kikinis, An Automatic Registration Method for Frameless Stereotaxy, Image Guided Surgery, and Enhanced Reality Visualization, Proc. CVPR'94, pp.430-436, Seattle, WA (1994).
5. C. Pelizzari, K. Tan, D. Levin, G. Chen, J. Balter, Interactive 3D Patient - Image Registration, Proc. 13th Int. Conf. on Information Processing in Medical Imaging, Berlin, Germany, pp.132-141 (1993).
6. D. Gennery, Tracking known three-dimensional objects, Proc. 2nd Nation. Conf. Artif. Intell., Pittsburgh, pp.13-17 (1982).
7. D.G. Lowe, Robust Model-Based Motion Tracking Through the Integration of Search and Estimation, Int. J. Computer Vision, Vol. 8, No.2, pp. 113-122 (1992).
8. D.G. Lowe, Fitting Parameterized Three-Dimensional Models to Images, IEEE Trans. Patt. Anal. Mach. Intell. Vol. 13, No.5, pp. 441-450 (1991).
9. D.B. Gennery, Visual Tracking of Known Three-Dimensional Objects, Int. J. Computer Vision, Vol. 7, No. 3, pp. 243-270 (1992).
10. M. Turk and A. Pentland, Face Recognition Using Eigenfaces, Proc. CVPR'91, pp.586-591, Maui, U.S.A. (1991).
11. H. Murase and S. Nayar, Parametric Eigenspace Representation for Visual Learning and Recognition, Tech. Rep. CUCS-054-92, Columbia University, NY (1992).
12. S. Yoshimura and T. Kanade, Fast Template Matching Based on the Normalized Correlation by Using Multiresolution Eigenimages, Proc. IROS'94, Munchen, Germany (1994).
13. O. Amidi, Y. Mesaki, T. Kanade, and M. Uenohara, Research on an Autonomous Vision-Guided Helicopter, Proc. RI/SME Fifth World Conf. on Robotics Research, Cambridge, Massachusetts (1994).
14. I. Weiss, Geometric Invariants and Object Recognition, Int. J. Computer Vision, Vol.10, No.3, pp. 207-231 (1993).
15. J. Mundy and A. Zisserman, Introduction-Towards a New Framework for Vision. In Geometric Invariance in Machine Vision, MIT Press, Cambridge, MA (1992).
16. H. F. Durrant-Whyte, Uncertain Geometry in Robotics, IEEE J. Robotics and Automation, Vol.4, No.1, pp.23-31 (1988).