INDIVIDUAL OBJECT INTERACTION FOR CAMERA CONTROL AND MULTIMEDIA SYNCHRONIZATION

Richard Y.D. Xu

Faculty of Information Technology, University of Technology, Sydney, Australia
richardx@it.uts.edu.au

Jesse S. Jin

School of Design, Communication & I.T
The University of Newcastle, Australia
jesse.jin@newcastle.edu.au

ABSTRACT

In recent times, most of the computer-vision assisted automatic camera control policies are based on human events, such as speaker position changes. In addition to these events, in this paper, we introduce a set of natural camera control and multimedia synchronization schemes based on *individual* object interaction. We present our methods in detail, including head-pose calculation and laser pointer guidance, which are used to estimate the region of interest (ROI) for both hand-held and object-at-distance. We explain, from our results, of how these set of approaches have achieved robustness, efficiency and unambiguous object interaction during real-time video shooting.

1. INTRODUCTION

Computer vision technologies have recently been applied to human-less camera control and automated multimedia-to-video synchronization. These applications have made video presentation more professional and reduced labor cost. Many of these applications are applied, typically to real-time e-learning [1] [2]. In these systems, the camera control policies are mostly based on the presenter and other scenario-dependant events. For example, the authors in [2] have proposed to use blackboard writing region and teacher's position changes to control the camera during lecture video-shooting.

Several other systems have recognized the importance of associating the presenting objects with its camera control scheme. For example, in the system described by [3], the authors have applied primarily hand tracking techniques to control video shooting for desktop-based TV presentation. The video shooting policies in these works, however, do not recognize the presenting object itself. Therefore, the same video shooting policy is applied regardless which object presenter is currently interacting with.

2. MOTIVATION AND CHALLENGES

In our work, we aim to further improve the camera control capability by associating a policy to each individual presenting object. As a consequence, video shooting rules can be customized based on presenter's interaction with a particular object, such as variable close-up shooting time. Object recognition can also assist multimedia synchronization. This is illustrated in Fig.1, such that, when local system detects the presenter's interaction with a pre-trained *"tool"* object, then, the remote audiences are automatically informed of the *tool's* detected region and its pre-authored multimedia is shown simultaneously.

Fig 1 (left) is the illustration of multimedia synchronization based on object detection. **Fig. 2 (right)** is the camera system hardware

The hardware used in our work is based on our design of a *portable* robotic camera, shown in Fig.2. The system contains one static and one PTZ camera placed closed together. This robotic system requires one standard PC for its computer vision processing. The advantage of our design in comparison with a multi-processor and fixture-to-room camera system [1][2][3] is stated in [4].

At the moment, we are supporting two presenter interactions. They are depicted in Fig 3. For hand-held objects, shown in Fig 3.a, the presenter holds an object one at the time while he/she is presenting. In this interaction, the object is brought close to the static camera, and usually appears sufficiently large for recognition. For objects not easily (or feasible) to pickup by the presenter or at a distance to the camera, the presenter can specify its position by drawing a virtual ellipse using a laser pointer around it to indicate its ROI. This is shown in Fig 3.b.

2.1. Challenges to apply SIFT based object matching

The object detention algorithm used in our work is based on Scale Invariant Feature Transform (SIFT) [5], a popular

(a) (b)

Fig 3. (a) is presenting a hand-held object, (b) is presenter drawing a virtual ellipse around a remote object using laser pointer

local invariant feature matching algorithm used in image database retrieval.

When we apply SIFT to real-time video object recognition, we must improve its *robustness* and *efficiency*.

For robustness, we have applied our novel distance ratio test under reference scale approximation to further improve SIFT's robustness against variations between training and video objects in affine transformations and illumination changes. This is discussed in detailed [6].

The efficiency issue arises, as SIFT process is too computational for real-time video object matching if SIFT is generated on every video frame using single PC processing (recall that our robotic camera uses only one PC). Therefore, in order to achieve real-time efficiency, we have applied a head-pose estimation to determine the most likely ROI containing the presenting object in an effort to reduce SIFT computation. This will be discussed in Section 3.

On the other hand, the distant-object specified by laser pointer, may be too small to generate sufficient SIFT features for matching. Therefore, we are performing matching after obtaining PTZ camera's close-up view of the region. Its technical overview will be given in Section 4.

3. HAND-HELD OBJECT RECOGNITION

In our work, we have used presenter's head-pose to estimate ROI before SIFT computation on hand-held object. To validate this method's appropriateness, the following observations were made from watching several presenters presenting hand-held objects in front of a camera.

Table 1: Head pose with respect to object presentation

	Observed head pose	Likelihood
O1	Presenter faces the camera while presenting	True most of times
O2	Presenter's face is usually not occluded before the camera	Almost true always
O3	The presenter looks into the direction of the presenting object for a short period of time at least once during presenting	True most of the times
O4	The distance between presenter's face to the hand-held object is usually within three times the width of the face horizontally and twice height of the face vertically	True most of the times

In our experiments, we have also tested other methods for ROI calculations, including hand tracking and background subtraction. We have found that the head pose is a superior approach.

Compared with the hand-tracking method, the presenter's face is usually not occluded at all during presentation (*Observation 2*). The face motion is much slower and more linear than that of the hands. Compared with background subtraction (we assume foreground being the presenter and the object), pose orientation results smaller ROI (56% smaller on average during our experiment, when presenter moves closer to the camera.) than does background subtraction.

3.1. Pose orientated hand-held object recognition

Our algorithm illustrated in Fig 4, is used to detect newly presented object and tracking of the recognized ones. We notice that SIFT computation is reduced significantly from every frame to be interaction-driven instead.

```
Pose-driven object recognition routine:

INITIALIZE: Frontal Face detection
WHILE (frontal face AND eyes are detected)
    Compute face orientation
    WHILE Presenter is NOT looking directly in front AND
            NO object is being recognized
        1. Compute ROI candidates from detected face size
        2. Calculate best ROI candidate according to face orientation
        IF (Presenter is looking at the same ROI continuously for a
            short period of time (observation 3) ) {
            Perform SIFT-based Recognition }
        IF (Object recognized) {
            Update object colour distribution
            Call Object Tracking sub routine }
    END WHILE
END WHILE

Object tracking subroutine:
WHILE
    IF (pdf similarity measure is higher than threshold)
        Tracking object
    ELSE
        Restart pose-driven object recognition routine
END WHILE
```

Fig 4 is the pseudo code for pose-oriented SIFT ROI estimation

3.2. ROI candidate estimations

By Observation 4 in Table 1, we have stated that hand-held object is usually close to presenter's face during presentation. Since face detection is a precursor in pose calculation, therefore, at no additional cost, this information is also used to determine a set of ROI candidates. When static camera is set to 320 * 240 frames, four ROIs were drawn, shown in Fig 5.

3.3 Pose calculation

There are many researches on head pose estimation. Most of these approaches require many feature points on the face [7], which are too computational for our purpose, as pose calculation only serves as an overhead task.

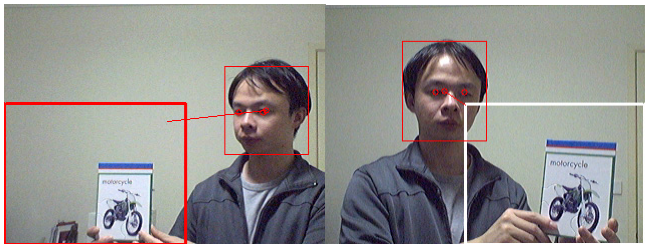


Fig. 5 is the two of the four overlapping ROI candidates from detected face size determined from pose orientation

Our pose orientation is based on a modified approach to [8]. The method only requires two points: the centroid of the head and the midpoint of the eyes, to determine α (pitch angle), β (yaw angle) and γ (roll angle):

$$\alpha = \sin^{-1}\left(\frac{V_Y - C_Y}{r}\right) - \sin^{-1}\left(\frac{U_Y - C_Y}{r}\right) \quad r = \sqrt{(U_Y - C_Y)^2 + (E_Z - C_Z)^2}$$

$$\beta = \sin^{-1}\left(\frac{V_X - C_X}{r}\right) - \sin^{-1}\left(\frac{U_X - C_X}{r}\right) \quad r = \sqrt{(U_X - C_X)^2 + (E_Z - C_Z)^2}$$

$$\gamma = \tan^{-1}\left(\frac{V_Y}{-V_X}\right) - \tan^{-1}\left(\frac{U_Y}{-U_X}\right)$$

To determine the centroid of the head required in [8], we have applied face detection algorithm using Haar-like feature algorithm by [9]. The result from face detection outputs a rough rectangular region for the head. We have then used combination of edge detection (Sobel filter) and Support Vector Machine (SVM) colour training to obtain a rough head shape. An ellipse fitting based on this information is used to determine the head shape.

3.3 Object tracking

To avoid object re-detection, once the object is detected, the system is then switched to a computationally efficient object tracking method by color features [10]. The tracking continues until the object is lost.

3.4 Results on hand held object recognition

By applying our pose-driven method, we have limited the object matching process to where and when interaction occurs. In Fig 6, ROI is determined from the pose analysis, SIFT features in each ROI is calculated.

After SIFT features are generated in its corresponding ROI, it is then matched with SIFT features stored in the training image database. The results are shown in Fig 7.

3.6. Efficiency improvement

We have compared execution time using our pose-driven ROI preprocessing with SIFT generation on each video frame. The experiments were performed using a testing PC (P-M 1.5, 512 RAM) on 320 *240 resolution video captures with moderate texture in the background. The results are shown in Fig 8.

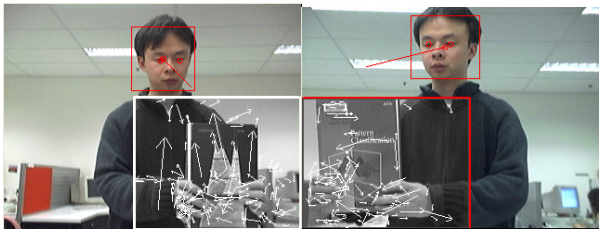


Fig 6. Calculation of SIFT within the ROI.

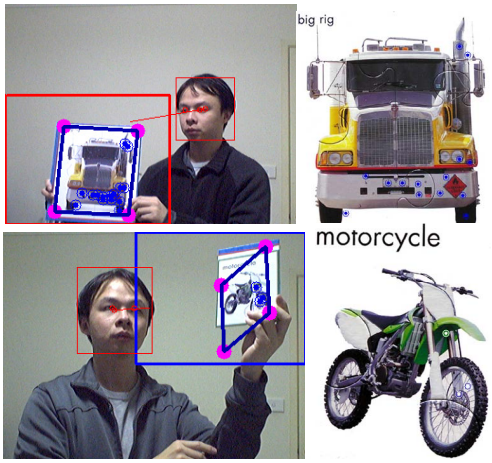


Fig 7. Pose-driven hand-held object recognition results, the blue line and pink corner is the calculated object pose determined from the matched SIFT key points (shown in blue circle)

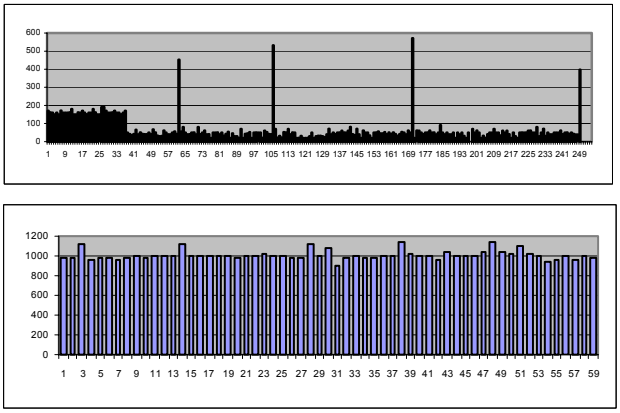
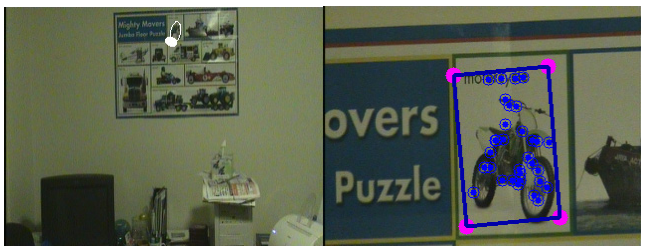


Fig. 8 Execution time comparison on object recognition event

The top figure in Fig 8 is the execution time using our method. Its initial computation, averaged around 160ms is due to presenter's face detection. Once face is identified, the system switches to a computational efficient face tracking algorithms, where pose analysis is then calculated in nearby pixels of face region. The average time for tracking and pose calculation (including eye detection) is about 50ms. The bottom figure is the execution using SIFT computation on every video frame, where this method averaged about 1000ms each computation. By comparisons, our method is much computational efficient.



(a) (b)

Fig. 9 Distant-object recognition (a) detected laser pointer virtual ellipse, the white dot is the last detected point. (b) PTZ camera view after zoom-in, showing calculated object pose and matched SIFT keypoint

4. DISTANT-OBJECT RECOGNITION

The result of distant-object recognition is shown in Fig 9. The most technical challenging tasks are the robust laser pointer detection and automatic PTZ camera control:

4.1 Robust laser pointer detection

Robust laser pointer detection is not as trivial as it looks. The most challenging task comes from a so-called CCD clipping phenomenon. This is because most digital camera's CCD sensors are only capable of capturing light with a maximum intensity close to white. Therefore, when a red laser dot is project onto a white surface, the amount of increase in camera's intensity is not significant. Hence simple color thresholding method is unable to achieve robustness. In addition, there are background noises, such as reflection of metallic surfaces causing false detection.

There are a few approaches proposed to combat this problem[11]. In our work, we have used our own spatiotemporal training, where during training, we record the maximum integral image intensity changes from a training region. The result can then be modeled using bi-modal Gaussian Mixture Model (GMM), separating noise from the true laser pointer illumination, shown in Fig10.

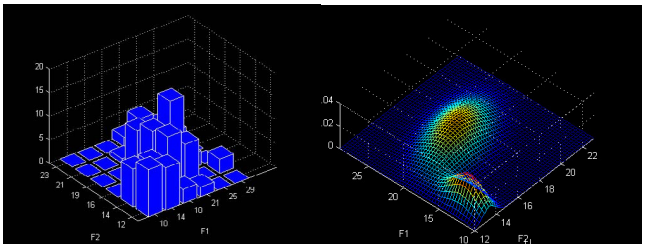


Fig. 10 Spatiotemporal features during laser pointer training: left is the histogram; right is the corresponding bi-modal GMM fittings.

4.2 PTZ camera movement

The PTZ camera movement control is detailed in [6]. In essence, the control algorithm is based on tracking the laser pointer specified region, while PTZ camera is performing self centering, when self-centering completes, the PTZ camera then performs zoom-in operation. The tracking is based on optical flow:

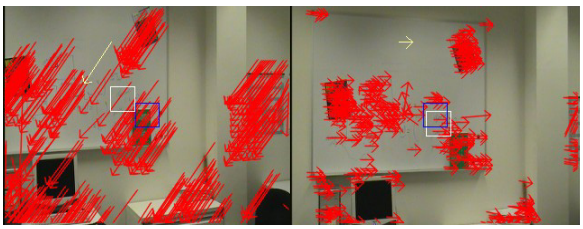


Fig 11 PTZ camera self-centering by tracking optical flow, until specified region (white) appears at centre of camera view (blue).

5. DISCUSSIONS

In this paper, we have presented a set of technical implementations in which SIFT-based object matching is used for camera control and multimedia synchronization. We have shown our motivation, and demonstrated a set of methods; including head pose estimation, robust laser pointer detection and PTZ camera control. The combined results are used together to achieve accurate ROI estimation for both hand-held and distant-object interactions.

During our experiment, the head pose has shown 90% accuracy rate in ROI estimation, while laser pointer specified region has accuracy rate in excess of 95%. Our future works have also included the recognition of additional presenter-object interactions.

REFERENCES

[1] A. Shimada, and A. Suganuma, Automatic camera control system for a distant lecture based on estimation of teacher's behavior, Seventh IASTED international Conference on Computers and Advanced Technology in Education (2004) 106-111.

[2] M. Onishi, and K. Fukunaga, Shooting the lecture scene using computer-controlled cameras based on situation understanding and evaluation of video images, 17th International Conference on Pattern Recognition 1 (2004) 781-784.

[3] M. Ozeki, Y. Nakamura, and Y. Ohta, Automated camerawork for capturing desktop presentations - camerawork design and evaluation in virtual and real scenes, 1st European Conference on Visual Media Production (CVMP) (2004) 211 - 220.

[4] R. Y. D. Xu, and J. S. Jin, Adapting Computer Vision Algorithms to Real-time Peer-to-Peer E-learning: Review, Issues and Solutions, To Appear in World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education (2005).

[5] D. Lowe, Distinctive image features from scale invariant key points, International Journal of Computer Vision 60 (2004) 91-110.

[6] R. Y. D. Xu, and J. S. Jin, IVDA: Rationale, Design and Implementation of a Computer-Vision Based Interactive E-learning System, Int. Journal of Distance Education Technologies (Under Review, 2005)

[7] J. Gemmell, C. L. Zitnick, T. Kang, K. Toyama, and S. Seitz, Gaze-awareness for videoconferencing: a software approach, IEEE Multimedia 7 (2000) 26-35.

[8] B. Yip, and J. S. Jin, Viewpoint determination and pose determination of human head in video conferencing based on head movement, 10th International Multi-Media Modelling Conference (2004) 130-135.

[9] R. Lienhart, and J. Maydt, An Extended Set of Haar-like Features for Rapid Object Detection, IEEE ICIP 2002 1 (2002) 900-903.

[10] D. Comaniciu, V. Ramesh, and P. Meer, Kernel-Based Object Tracking, IEEE Trans. Pattern Analysis and Machine Intelligence 25 (2003)

[11] Y. Shi, W. Xie, and G. Xu, Smart Remote Classroom: Creating a Revolutionary Real-Time Interactive Distance Learning System, Advances in Web-based Learning 1st Int. Conf. Web-based Learning 2002, 130-141.