Real Time Single and Multiuser Gesture Recognition Based on Skin Colour and Optical Flow

Muhammad Raza Ali[1]
alim@cs.man.ac.uk

Tim Morris[1]
tmorris@cs.man.ac.uk

[1] Advanced Interfaces Group,
School of Computer Science,
University of Manchester, UK.

Abstract

Gestural interfaces offer the potential for a natural and non-cumbersome human computer interaction. High gesture recognition rates for both recorded sequences and real time video streams have been achieved. However, real time gesture recognition still remains a challenge especially for settings where ambient conditions (e.g. lighting, background) are subject to change and where there are multiple simultaneous users. In this paper we present a feature descriptor based on the radon transform to represent a gesture-making hand.

For multiuser gesture recognition we present a (non-Bayesian) mechanism for tracking arbitrarily changing numbers of skin regions and a method of selecting particular skin region(s) for feature extraction and gesture recognition. Tracking and selection is achieved using skin colour information and optical flow magnitude. The evaluation is done in an everyday indoor setting using an off the shelf webcam, without any specialized equipment i.e. specialized lighting, data gloves or markers. The evaluation results show the robustness of the proposed descriptor and potential for application of the tracking mechanism in an unconstrained multiuser scenario.

1 Introduction

Computer vision based gesture recognition has attracted interest from researchers and practitioners in computer vision [24] and human computer interaction [18]. Gesture recognition has a wide range of applications from desktop applications [1], gaming [2], surveillance [20] to classroom teaching [3]. The research in single user gesture recognition is well established [4, 5, 23]. However, the potential of gestural interfaces in applications that involve more than one user is yet to be exploited.

There have been some important studies and advancements that show the promise of multiuser applications. Advanced controllers like Kinect have redefined the gaming experience. A state of the art system [6] allows two users to manipulate Google maps on large screens using hand gestures. Two prototypes have been developed [13] that enable the manipulation of virtual objects, projected on a purpose built table, by gestures. Another specialized table is used in [2] for a multiuser computer game. These systems, although state of the art, restrict the users in terms of freedom of movement and natural interaction. For example you need to be in the proximity of the specialized glass table to manipulate the virtual objects and to manipulate Google maps you need special gloves. Techniques that work in an unconstrained environment will engender a wider range of applications e.g.

intelligent spaces through ambient intelligence, collaborative problem solving etc. In this paper, we present feature extraction and tracking techniques that are aimed at both single and multiuser application scenarios with minimum constraints. The paper is organized as:

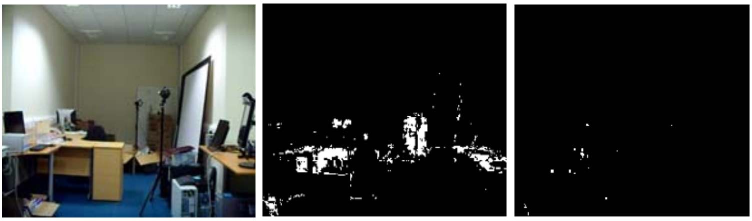
Section 2: Briefly describes the background subtraction technique [14] used to segment skin regions from the rest of the image.

Section 3: Describes a novel feature descriptor based on the Radon transform, and compares it with another state of the art Radon based technique [8]. The descriptor is evaluated for real time application. The evaluation is done using a novel framework.

Section 4: For multiuser interaction, we present a mechanism capable of tracking arbitrarily changing numbers of skin regions and selecting region(s) of interest for feature extraction and gesture recognition. This is achieved using a metric based on skin colour and Lucas Kanade [7] optical flow algorithm. Multiuser interaction is a work in progress. However, initial evaluation demonstrates potential for an advanced, multiuser application in an unconstrained setting.

2 Background Subtraction

In this paper as we are dealing with human users, so skin colour can be used for background subtraction. We rely on a technique that employs joint-thresholding using normalized red, green chromaticity space and optical flow magnitude [14]. Figure 1 shows the advantage of using this technique. The skin colour based background subtraction results in large false positive regions (1.b). These regions cannot be removed unless skin colour is combined with additional information i.e. optical flow magnitude (1.c).



(a) (b) (c)

Figure 1: Comparison of background segmentation results, a) original b) skin colour based segmentation c) skin colour + optical flow magnitude

3 Feature Extraction

The feature descriptor needs to cope with expected variation in conditions. As our methodology is based on (skin) colour, the proposed feature descriptor is expected to cope with the challenges of varying lighting conditions and cluttered background. These problems can lead to poor segmentation thus affecting the recognition rates. Our feature descriptor is based on the Radon transform of the segmented hand contour.

3.1 Radon Transform

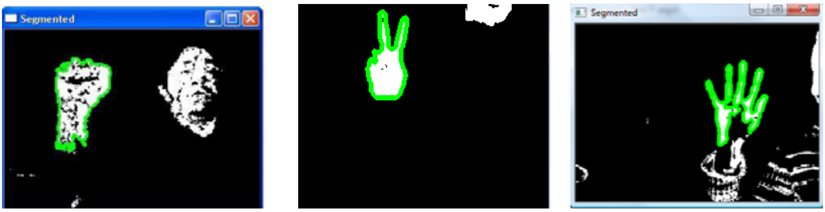
The Radon transform is an established technique in medical imaging especially for computed tomography [19]. The transform has a very important property of rotation invariance [36]. The Radon transform of an intensity image is given by projections or line integrals at certain angles. For an image *f(x,y)*, the Radon transform at an angle θ, is given as:

$$R(\theta, r) = \iint f(x, y)\delta(x\cos\theta + y\cos\theta - r)dxdy \qquad (1)$$

Where, δ is the delta function, r is the perpendicular distance from the origin to the projection line (or beam) and θ is the angle at which the transform is computed. The Radon transform for the entire image is a collection of such transforms computed at various angles. The transform has been used in various computer vision applications [39, 40].

3.2 Proposed Radon Transform Based Descriptor

The feature extraction stage comprises of two steps. After background subtraction, segmented regions are processed using the connected component labelling, from these regions we select a region of interest or gesture making hand (section 4.2) and its external contour is extracted through OpenCV implementation of [21].



(a) (b) (c)

Figure 2: The extracted contour for various gestures

Once we obtain the contour, its Radon transform is computed at eight angles (0, 22, 45, 67, 90, 112, 135 and 157 degrees) and over 51 beams or projection lines. The values are summed and normalised over all orientations, giving 51 features to represent the contour. A smaller number of beams can be used for smaller images. Increasing the number of projections does not improve the representation of gestures.

3.3 Quantitative Comparaison with Song et al.'s Descriptor [8]

Song et al.[8] presented a feature descriptor based on the Radon transform for gesture recognition. Feature extraction involves splitting the segmented hand into various regions and the transform values are summed within those regions. The difference in the sum of these regions basically distinguishes one gesture from the other. The study reported high gesture recognition results for static gestures. However, during our experiments we noticed that in most cases the segmentation of the hand from the background is not perfect. The segmented skin region contains many false negatives or black regions as shown in figure 2. This is particularly true for real time applications where the lighting conditions can vary. We implemented the descriptor presented by Song and compared it with our descriptor for a four gesture vocabulary (figure 3). The test set comprised of around 2000 images\frames taken from various publicly available datasets [9, 10]. The classification was done using SVM (RBF Kernel). Both descriptors performed well within an average accuracy of ~98%.

The results of our descriptor were clearly better for live video stream through a webcam. Four participants were asked to make each gesture at least 50 times, with a combination of left and right handed gestures. The average accuracy of both descriptors was less than that for static images but our descriptor proves much more robust with an average accuracy of more than 95%; for the Song descriptor it was around 85%. There was a significant overlap between gestures B and C.

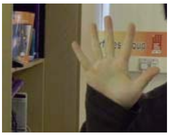
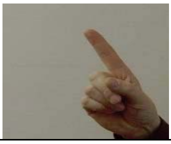
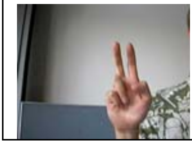
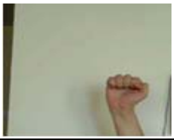
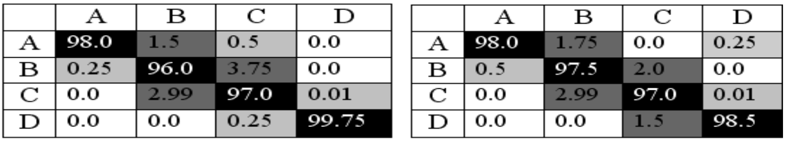
Gesture Class	Class Label	Gesture Class	Class Label
	A		C
	B		D

Figure 3: The 4-gesture vocabulary used for descriptors' comparison



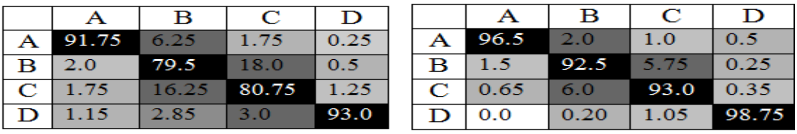
	A	B	C	D
A	98.0	1.5	0.5	0.0
B	0.25	96.0	3.75	0.0
C	0.0	2.99	97.0	0.01
D	0.0	0.0	0.25	99.75

(a)

	A	B	C	D
A	98.0	1.75	0.0	0.25
B	0.5	97.5	2.0	0.0
C	0.0	2.99	97.0	0.01
D	0.0	0.0	1.5	98.5

(b)

Figure 4: The confusion matrix for a) Song et. al[8] descriptor, b) our descriptor



	A	B	C	D
A	91.75	6.25	1.75	0.25
B	2.0	79.5	18.0	0.5
C	1.75	16.25	80.75	1.25
D	1.15	2.85	3.0	93.0

(a)

	A	B	C	D
A	96.5	2.0	1.0	0.5
B	1.5	92.5	5.75	0.25
C	0.65	6.0	93.0	0.35
D	0.0	0.20	1.05	98.75

(b)

Figure 5: The confusion matrix for real time data, a) Song descriptor, b) our descriptor

This decrease in accuracy is due to 'imperfect' segmentation. Large false negative or black regions result in low or negligible Radon transform values in those regions, making it difficult to achieve reasonable separation between identical gestures classes. On the other hand the proposed descriptor will work as long as we get an approximation of the contour. An extreme example is shown in figure 2(c) due to a sudden change in lighting conditions. Despite poor segmentation this particular example was correctly classified.

We have chosen the Song et. al.[8] descriptor for comparison as it is related to our work and produces high recognition rates for 'hand' gestures. Some computer vision studies use the term 'gesture' to describe the upper or whole body movements; this includes some recently reported research [26, 27]. Various techniques have been proposed [28, 29, 30, 31, 32, 33] that rely on skin colour for feature extraction, gesture recognition or tracking etc. Most of these studies have shown high recognition rates (well above 94%) and in some cases real time performance. However, these techniques rely heavily on good segmentation of the gesture making hand for feature extraction and recognition. As we have seen, poor segmentation can adversely affect the recognition accuracy. The large number of false negatives will make it virtually impossible to extract features reported in these studies e.g. feature descriptor in [32] extracts features for recognition around the centre of the palm. Poor segmentation is likely to affect the accuracy of this descriptor.

3.4 Extended Evaluation on Real Time Data

The gesture vocabulary was extended from four to seven gestures for this stage of evaluation as shown in figure 6. We intentionally included potentially confusing gestures in our vocabulary e.g. B and E.

We developed a single-user application for descriptor evaluation. It involves selecting and moving objects to a target location. The layout of the application is shown in figure 7. Smaller boxes on the left are to be moved to the larger target boxes on the right. The labels of the target boxes indicate the smaller boxes that are to be moved. The size of an incoming frame is 320x240 pixels. Each of the six boxes is selected by a separate gesture and moved using the 'hold' gesture. The number of target locations is limited to three due to the size of the image frame.

The success of a gesture recognition technique is usually the demonstration of a working prototype or manipulation of virtual objects. However, detailed error or performance statistics are not reported. We propose a detailed evaluation framework using criteria outlined below:

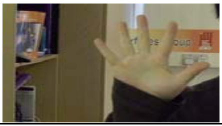
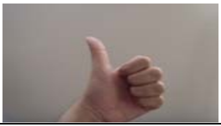
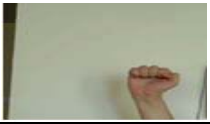
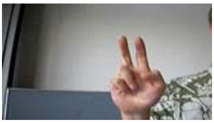
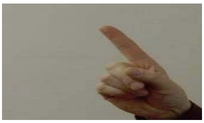
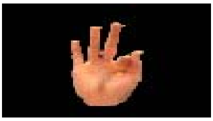
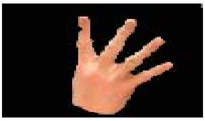
Gesture Class	Class Label	Gesture Class	Class Label	Gesture Class	Class Label
	A		D		G – Hold Gesture
	B		E		
	C		F		

Figure 6: The extended gesture vocabulary for single user real time application

1. Elapsed frames during recognition: This essentially measures the number of frames that elapse during the process of recognizing a gesture and selecting the correct object on the screen. It is important that this number is small to avoid delay in interaction and especially when switching gestures. In other words, it is not sufficient to select a correct object but how long it takes to select that object is of importance.

2. Accuracy of classification: This gives us the percentage of frames that were misclassified during the manipulation\movement of screen objects. Consider a case that involves moving objects **A** and **B**; only three gestures should be recognized in this operation i.e. gestures **A**, **B** and **hold** gesture. Any other gesture is a misclassification.

3. Wrong Object Moved: Based on the results on static images it was unlikely that a misclassification causes a wrong object to be selected and moved. However, this criterion is particularly of interest under adverse lighting conditions where segmentation and recognition of a similar gesture across successive frames becomes inconsistent. In our evaluation, only 3 operations had erroneous movements (see Table 1), as the application handled most these inconsistencies.

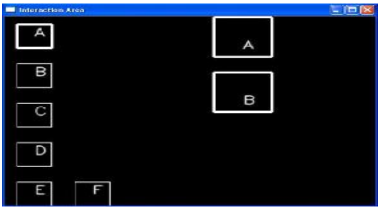


Figure 7: The layout of the interaction area. Box ‘A’ is selected

4. Time: This is the time taken to complete an operation; in this case moving particular objects form their initial position to target locations.

Criterion 3 is specific to applications that involve manipulating a virtual object. However, the remaining three criteria can be used to evaluate any gestural application. This could help in the formulation of a standardized evaluation protocol. A seven gesture vocabulary is sufficient for most gestural applications as shown in [1, 38]; reported applications work with a 2 and 3 gesture vocabulary.

3.5 Evaluation Results

Two tests were conducted for evaluation. In the first test it was required to move two objects out of six to the target locations. Similarly, for the second test three objects were required to be moved. We tested for all the possible combinations of two and three objects e.g. AB and BA were both tested. The results reported here are for five attempts i.e. total 5x150 operations, not just the best results. The testing was done over a period of three days at different times of the day. The main source of lighting was florescent, with varying contribution from natural light.

Testing Criteria	Benchmark based on Training, Offline Testing and Parameter Tuning[16]	Two-object test	Three object test
Elapsed Frames	3	3.3±1.3	3.5±1.1
Gesture Classification Accuracy	> 92%	95%	94.3%
Incorrect Object Selection/Movement	In < 10 operations	0	3
Completion Time per operation	10s for two object, 16s for three object,	9s±1.5s	15s±1.9s

Table 1: Evaluation results for the proposed descriptor on real time data

4 Tracking Mechanism

Tracking is achieved using a joint metric based on skin colour and optical flow. This metric was introduced in [35] for Bayesian tracking and evaluated on video sequences. We propose a non-Bayesian framework based on this metric that reliably tracks arbitrarily changing number of skin regions in real time. We also extend this metric by introducing a mechanism for recovering occluded skin regions. Previous studies that deal with tracking skin regions in real time have focussed on a single user. An important non-Bayesian technique presented in [11] shows robustness to fast movements, occlusions etc. A simplified version has been used for a computer game [12]. However, both these studies do not explicitly address the issue of cluttered background that can create large false positive regions adversely affecting the system performance. Combination of skin colour and optical flow handles background clutter without relying on Bayesian filtering [41].

4.1 Overview of Tracking Mechanism

After processing of image data each candidate region at *t* is represented as:

$$C_i = [\alpha, \beta, (x, y), (w, h), \omega_t, \omega_{t-1}] \qquad (2)$$

Where, α is the average optical flow magnitude for the candidate region, β is the skin similarity measure that gives the percentage of pixels with normalized red, green values lying within one standard deviation of mean normalized red, green value for the region. This is based on an important property of skin colour i.e. its distribution is much more compact compared to the distribution of a non-skin object [15]. The upper left corner of the region is given by *(x, y)*. The dimension of the bounding box around the region is given by *(w,h)* i.e. width, height. The parameter ω_t indicates the joint score based on skin similarity measure and optical flow, given as:

$$\omega_t = a + (b.\beta) \qquad (3)$$

Where, b is the weighting coefficient. The value of b depends upon the percentage of pixels lying within one standard deviation as shown in Table 2. The region of interest is selected based on the joint score given by (3). Since the *ω* score is available for all the regions it is possible to select the top two or three regions.

β	b
< 30%	0.25
30% < β < 50%	0.5
50% < β < 80%	1
> 80%	2

Table 2: Weighting coefficient values

In the proposed mechanism we are looking to handle changing numbers of candidate regions, reliable selection of the region of interest, handling of occlusions and error in optical flow computation. There are three possible scenarios in terms of changing candidate regions:

a. If number of regions remains the same, the parameters are updated for all regions.
b. If there is a decrease in the number of regions, it is most likely that the regions with the lowest *ω* score at *t-1* have been dropped. However, due to error in optical flow or occlusions, higher ranked regions may be lost for a brief period. Therefore, a higher ranked region is not discarded straightaway allowing it to recover.
c. If there is an increase in the number of regions, parameters are computed for new regions and for existing regions *ω* is updated.

For a single-user, desktop applications we can rewrite (2) as:

$$C_i = [\alpha, \beta, (x, y), (w, h), \omega_t] \qquad (4)$$

The region with the highest score from up to three candidate regions (hands and face) is the ROI. Due to the nature of the application there is a rare chance of occlusion or optical flow error. The last parameter is useful in handling tricky false positive regions e.g. clothes, hair.

4.2 Region of Interest(ROI)

As mentioned above, a region of interest is selected from candidate regions based on the joint score given by (3). At time *t* candidate regions are ranked using the weighted sum as:

$$ROI_i = 0.35\omega_t + 0.65\omega_{t-1} \qquad (5)$$

The *ω* score for the last time step is given more weight due to the pattern of change in the value of optical flow magnitude. As the gesture is made there is a drop in the *ω* score. So without a robust check the gesture making hand may not be selected as a ROI.

A drop in score indicated by ω_t may not result in selection of a gesture making hand if we ignore $\omega_{t-1.}$ As the hand moves into a gesture making position it is tracked over successive time steps and is probably among higher ranked regions at *t-1.*The movement of hand is less pronounced when a gesture is made, resulting in a reduced value of *ω* at *t.* A weighted combination (5) ensures that gesture making hand is selected as a ROI. Our experiments show that gesutre making hand(s) is selected for feature extraction/gesture recognition successfully at almost all occasions. Another benefit of keeping this score is that it allows the tracker to recover candidate regions lost temporarily to occlusions or optical flow error. It is worth mentioning that the weights given in (5) are determined empirically using video sequences for OpenCV's implementation of Lucas Kanade. For a different optical flow algorithm these weights may vary, although not significantly, and more weight will always be given to $\omega_{t-1.}$ This ROI is used for feature extraction.

4.3 Recovering the Candidate Regions

Occlusions were not a frequent occurrence in our evaluation and most of the candidate regions that were lost in tracking were due to errors in optical flow computation [22]. We encounter certain 'blind spots' where optical flow information is not available or the computed magnitude is below what is required for skin\candidate regions. Therefore, it is essential that the tracker is able to recover candidate regions especially those that were previously higher ranked. To cater for optical flow errors and occlusions we do not discard the higher ranked regions straightaway. However, for the period they are invisible they are not candidates for ROI.

In order to locate candidate regions after a possible optical flow error or an occlusion we search using the centroid of the lost region. We search for a blob or a connected region within a window of 10x10 pixels around the centroid of the region to be recovered. If the connected region (or part of it) is found the tracker looks for rest of the connected region. The advantage of using the centroid is that it can allow for change of size and the region can still be recovered. If a region is found, it is verified by the skin similarity parameter, allowing for small change in the value (±5%) as the region might have moved. If the 'recovered' region is less than 50% of the size of the lost original it is discarded. This technique successfully recovers regions where movement of the occluded region is not significant i.e. as long as part of the region falls within the search window, e.g. regions in gesture making position. If the region is not recovered, it is discarded, as occlusion might be permanent or the region moved significantly. For the latter case, once visible, the region is treated by the tracker as a new candidate region.

4.4 Simultaneous Recognition of More Than One Gesture

Tracking and ROI selection mechanism was evaluated using a real time application, allowing for selection of two objects simultaneously. In order to ensure a smooth, seamless interaction it is important that there is no or minimal delay in recognition of multiple gestures. Testing was done for all two object combinations. Results are averaged over three attempts. The breakdown of evaluation results is given in Table 3. The evaluation results show that the proposed scheme can be used for a multiuser application scenario.

During our test the number of tracked skin\candidate regions varied between two to eight, with up to two users actively interacting with a system and people moving in the background. Recognition of simultaneous gestures shows the potential of developing applications where users can work on modules of a collaborative task independently. The reported evaluation is in contrast to reported studies [17, 34] aimed at multiuser interaction but do not give evidence for simultaneous, multi-object manipulation. The frame size for this test was 640x480 pixels.

Criterion	Benchmark	Evaluation Result
Elapsed Frames Between Simultaneous Two Object Recognition	0-1frame	1.0±0.96
Misclassification During Two Object Recognition	in < 3 tests	2 instances of misclassification

Table 3: Evaluation results for simultaneous, multi-gesture recognition. The maximum delay is 2 frames, which is practically not noticeable.

5 Performance Issues

Although subjective evaluation is beyond the scope of the paper, the movement of objects was quite fluent and the hold gesture did not cause participants any stress. In the context of the proposed technique our main objective is to demonstrate an acceptable baseline performance on a CPU using an off the shelf Logitech Pro 5000 webcam and OpenCV. The performance outlined in Table 1 was achieved at a modest frame rate of around 10fps, although the interaction did not have a noticeable delay.

Profiling of the code showed that major delay is caused by connected component labelling. The code was optimized and instead of relying on CPU-based blob detection we switched to a GPU based library; ArrayFire[37]. This library is currently one of the most comprehensive developer tools for GPU programming especially for blob detection. The frame rate was now 25fps and the time taken to complete the operation was reduced significantly to 3.5s±0.25s for 2-object and 7.4s±1.3s for 3-object tests. Webcam allows a natural and relatively unconstrained interaction for up to two users. However, the connected component labelling feature of the library does not scale well for an image size greater than 640x480. This limitation, once addressed, will make it relatively straightforward to develop application involving more than two users. This is an important result as our tracking mechanism is fully capable of handling any number of skin regions.

6 Conclusions

The proposed descriptor is aimed at addressing some intrinsic issues with the use of skin colour in computer vision e.g. illumination variation, similar colour background etc. Evaluation on real time data shows robustness and accuracy for single and multiuser scenarios. We achieved an average real time accuracy of ~96% and 93% in adverse lighting conditions (please see supplementary material) for 7 gestures. The performance achieved using off the shelf hardware is to serve as a baseline for future work. We also present a novel evaluation framework for real time gestural applications. Simultaneous interaction of three or more users will be achieved using the GPU and more sophisticated imaging hardware. As future work, it would be useful to compare our tracker with established methods [42, 43] for tracking multiple skin regions in real time.

References

[1] Y.Y. Pang, N.A. Ismail, and P.L.S. Gilbert. A real time vision-based hand gesture Interaction. In *Proceedings of the Fourth Asia International Conference on Mathematical/Analytical Modeling and Computer Simulation*, pages 237-242, 2010.

[2] K. Li, Y. Du and Z. Fu. TreeHeaven: A table game using vision-based gesture recognition. In *Proceedings of the 2011 ACM symposium on the Role of Design in UbiComp Research & Practice*, pages 11-14, 2011.

[3] H. Ren and G. Xu. Human action recognition in smart classroom. In *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 417-422, 2000.

[4] Y. Wu and T. Huang. Vision-based gesture recognition: a review. In *International Gesture Workshop*, 1999.

[5] R. Hassanpour, S. Wong and A. Shahbahrami. Vision based hand gesture recognition for human computer interaction: A review. In *IADIS International Conference on Interfaces and Human Computer Interaction*, 2008.

[6] Y. Yin and D. R. Toward. Natural interaction in the real world: real-time gesture recognition. In *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction*, 2010.

[7] B. D. Lucas, and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of Imaging Understanding Workshop*, pages 121-130, 1981.

[8] Z. Song, H. Yang, Y. Zhao and F. Zheng. Hand detection and gesture recognition exploit motion times image in complicate scenarios. In *Proceedings of the 6th international conference on Advances in visual computing*, pages 628-636, 2010.

[9] T. Coogan, G. Awad, J. Han, and A. Sutherland. Real time hand gesture recognition including hand segmentation and tracking. In *Proceedings of the International Symposium of Visual Computing*, pages 495–504, 2006.

[10] P. Dreuw, T. Deselaers, D. Keysers, D, and H. Ney. Modelling image variability in appearance-based gesture recognition. In *ECCV Workshop on Statistical Methods in Multi-Image and Video Processing*, pages 7-18, 2006.

[11] A. Argyros and M. Lourakis. Real time tracking of multiple skin-colored objects with a possibly moving camera. In *Proceedings of the European Conference on Computer Vision*, pages 368-379, 2004.

[12] N. Vo, Q. Tran, T.B. Dinh and Q.M. Nguyen. An efficient human-computer interaction framework using skin color tracking and gesture recognition. In *International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future,* 2010.

[13] A. Del. Bimbo, L. Landucci , and A. Valli. Multi-user natural interaction system based on real-time hand tracking and gesture recognition. In *Proceedings of the International Conference on Pattern Recognition,* pages 55-58, 2006.

[14] M.R. Ali and T. Morris. Skin locus based skin detection for gesture recognition. In *BMVC Postgraduate Workshop,* 2010.

[15] M. Storring, J.H. Andersen, and E. Granum. Skin colour detection under changing lighting conditions. In *Proceedings of the Seventh Symposium on Intelligent Robotics Systems*, pages 187-195, 1999.

[16] C. Hsu, C. Chang, and C.J Lin. A Practical Guide to Support Vector Classification. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, 2003.
URL http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[17] G. Hackenberg, R. McCall and W. Broll, Lightweight palm and finger tracking for real-time 3D gesture control. In *Proceedings of the Virtual Reality Conference,* pages 19-26, 2011.

[18] R. Kjeldsen and J. Hartman. Design issues for vision-based computer interaction systems. In *Proceedings of the 2001 Workshop on Perceptive User Interfaces,* pages 1-8, 2001.

[19] Y. Cai, K. Shen, and J. Wang. Application of radon transform in CT image matching. In *16th World Conference on NDT*, 2004.

[20] G. Iannizzotto, F.L. Rosa, C. Costanzo, and P. Lanzafame. A multimodal perceptual user interface for video-surveillance environments. In *Proceedings of Multimodal Interfaces,* pages 45-52, 2005.

[21] F. Chang, C.J. Chen, and C.J. Lu. A linear-time component-labeling algorithm using contour tracing technique. *Computer Vision and Image Understanding*, 93(2):206-220, 2004.

[22] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. In *Proceedings of the 13th International Conference on Computer Vision*, 2007.

[23] S. Mitra and T. Acharya. Gesture recognition: A survey. *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, 37(3):311-324, 2007.

[24] J. Daugman. Face and gesture recognition: Overview. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7): 675-676, 1997.

[25] Y. Y. Pang, N. A. Ismail, and P. L. S. Gilbert. A real time vision-based hand gesture interaction. In *Proceedings the Fourth Asia International Conference on Mathematical\Analytical Modelling and Computer Simulation*, pages 237-242, 2010.

[26] D. Kim, J. Lee, H.S. Yoon, J. Kim, and J. Sohn. Vision-based arm gesture recognition for a long-range human robot interaction. *The Journal of Supercomputing*, pages 1-17, 2011.

[27] M. B. Holte, T. B. Moeslund, and P. Fihl. View-invariant gesture recognition using 3d optical flow and harmonic motion context. *Computer Vision and Image Understanding*, 114:1353-1361, 2010.

[28] H. Hongo, M. Yasumoto, Y. Niwa, M. Ohya, and K. Yamamoto. Focus of attention for face and hand gesture recognition using multiple cameras. In *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 156-162, 2000.

[29] H. Yoon. Hand gesture recognition using combined features of location, angle and velocity. *Pattern Recognition*, 34(7):1491-1501, 2001.

[30] Q. Duan and W. Ma. An approach to dynamic hand gesture modeling and real-time extraction. In *Proceedings of IEEE 3rd International Conference on Communication Software and Networks*, pages 139-142, 2011.

[31] E. Nielsen, L. Laguna, L. A. Canals, and M. Hernndeztejera. Hand gesture recognition for human-machine interaction. *Image*, 12(1):91-96, Rochester NY, 2004.

[32] N.D. Binh, E. Shuichi, and T. Ejima, Real-Time Hand Tracking and Gesture Recognition System. In *Proceedings of the International Conference on Graphics, Vision and Image Processing*, pages 362-368, 2005.

[33] M. Elmezain, A. Al-Hamadi, and B. Michaelis. Real-Time Capable System for Hand Gesture Recognition Using Hidden Markov Models in Stereo Color Image Sequences. *The Journal of WSCG*, 16(1):65-72, 2008.

[34] R.D. Vatavu, L. Grisoni, and S.G. Pentiuc. Gesture Recognition based on Elastic Deformation Energies. In *Gesture-Based Human-Computer Interaction and Simulation*. Sales Dias, M., Gibet, S., Wanderley, M.M., Bastos, R. (eds.). LNCS (LNAI), vol. 5085, pp. 1–12. Springer, Heidelberg, 2009.

[35] M.R. Ali and T. Morris. Combining skin colour and optical flow for computer vision systems. In *Proceedings of the International Conference on Machine Vision, Image Processing and Pattern Analysis*, pages 1808-1813, 2011.

[36] Z. Zalevsky and D. Mendlovic. Fractional radon transform: definition. *Applied Optics*, 35(23):4628-4631, 1996.

[37] ArrayFire, URL: http://www.accelereyes.com/products/arrayfire

[38] G. Shin and J. Chun. Vision-based multimodal human computer interface based on parallel tracking of eye and hand motion. In *Proceedings of International Conference on Convergence Information Technology*, pages 2443-2448, 2007.

[39] H. Zhou, P. Miller, J. Songg. Age classification using Radon transform and entropy based scaling SVM. In Jesse Hoey, Stephen McKenna and Emanuele Trucco, In *Proceedings of the British Machine Vision Conference*, pages 28.1-28.12. BMVA Press, September 2011.

[40] K. Jafari-Khouzani and H. Soltanian-Zadeh. Radon Transform Orientation Estimation for Rotation Invariant Texture Analysis. *Pattern Analysis and Machine Intelligence*, 27(6):1004-1008, 2005.

[41] D. Ross, J. Lim, R-S. Lin and M-H Yang. Incremental learning for visual tracking. *International Journal of Computer Vision, Special Issue: Learning for Vision*, 2007.

[42] H. Zhou, Y. Yuan and C. Shi. Object tracking using SIFT features and mean shift. *Computer Vision and Image Understanding*, 113(3):345-352, 2009.

[43] D. Comaniciu, V. Ramesh, P. Meer, Real-time tracking of non-rigid objects using mean shift. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 142–149, 2000.