

# Block-Matching-Based Motion Field Generation Utilizing Directional Edge Displacement

Hitoshi Hayakawa  
Department of Electronic Engineering  
The University of Tokyo  
Tokyo, Japan  
hayakawa@if.t.u-tokyo.ac.jp

Tadashi Shibata  
Department of Frontier Informatics  
The University of Tokyo  
Tokyo, Japan  
shibata@ee.t.u-tokyo.ac.jp

**Abstract**—A motion field generation algorithm using block matching of edge-flag histograms has been developed aiming at its application to motion recognition systems. Use of edge flags instead of pixel intensities has made the algorithm robust against illumination changes. In order to detect local motions of interest effectively, a new adaptive frame interval adjustment scheme has been introduced in which only the edge flags due to local motions present in the frame are accumulated and utilized in block matching. These edge flags are projected onto x and y axes to generate histograms and the motion in x and y directions are determined by histogram matching. As a result, the computational cost for best match search has been substantially reduced. A vector representation of the motion field is also proposed and has been applied to preliminary motion recognition experiments using Hidden Markov Models (HMMs). The advantage of the proposed method over the simple optical flow has been demonstrated.

## I. INTRODUCTION

Motion recognition plays important roles in a number of applications such as remote control by gestures, man-machine interaction, interpretation of sign languages, surveillance cameras and so forth. Motion recognition is composed of two stages: motion feature extraction and feature recognition. In most motion recognition systems, the emphases are placed on the latter stage, namely, on how to construct models for target motions and recognize them. In the feature recognition stage, motion specific models as well as more general ones such as Hidden Markov Models (HMMs) and Bayes Filters are employed [1,2,3]. On the other hand, in the motion feature extraction stage, there is still room for developing new schemes. Therefore, it is important to explore new one because the result of feature extraction directly influences the recognition performance. The motion feature extraction is typically performed by tracking feature points in the frame and analyzing their trajectories [4,5,6]. Feature points are marked beforehand using color information, object contours or pixel activities. In tracking methods, difficulties are encountered in consistent

tracking, i.e. how to identify one feature point in the present frame as the same feature point in the previous frame.

In other approaches, motion features are directly extracted from image sequences. The luminance change at each pixel site is obtained and represented as two-dimensional maps and pattern analysis is carried out on such maps [7,8]. Since the map contains only the information of temporal pixel intensity activities, motion speeds and directions are not included in the motion feature representation. This certainly limits the performance of motion recognition. To solve this problem, motion fields are often utilized. The motion field is a map of local velocities each representing the motion vector determined at every pixel site. In [9] and [10], the statistical features obtained from a motion field are utilized to represent the motion in the frame.

Usually, motion field is generated using gradient methods such as normal optical flow, in which the motion vector at each pixel is calculated using pixel intensities of small area, i.e. intensities at neighboring pixels in the same frame and those at the same pixel site in the next frame, under some constraints [11,12,13]. Although this method is very simple and the computational cost is low, the accuracy is limited because of the small area information. On the other hand, block matching is also used to generate motion fields. Block matching estimates the motion of each block in the frame by scanning with the image in the block the search area in the next frame and finding the best matched location. Block matching is more accurate than gradient methods but the problem is its high computational cost [14,15]. Moreover both the gradient method and the block matching are weak against illumination changes because they both rely on pixel intensities.

In order to solve these problems, block-matching-based motion field generation algorithm utilizing directional edge displacement is proposed in this paper. In this algorithm, block matching is carried out based on directional edge maps to make it robust against illumination changes. In order to

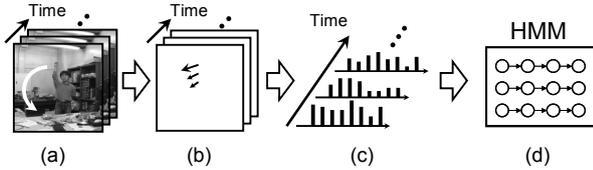


Figure 1. System organization of motion recognition system: (a) source image sequence; (b) motion fields; (c) vector expressions; (d) HMM-based sequence matching.

reduce the computational cost, the histogram matching technique has been employed in the search for the best match location. A new adaptive frame interval adjustment scheme has been introduced in the present work, which allows us to detect only motions of significance. As a result, false motions arising from background noises have been effectively eliminated. A motion-field-based feature vector representation has also been proposed to extract motion features from moving image sequences. Preliminary recognition experiments based on the proposed method were carried out using HMMs and the effectiveness of the algorithm has been tested.

## II. MOTION FIELD GENERATION ALGORITHM

Fig. 1 shows the organization of the motion recognition system which is under development in our laboratory. At first, a motion field is generated from a source image sequence. Then, the sequence of motion fields is converted to a sequence of vectors representing the motion in the image sequence. Finally, motion recognition is carried out using HMMs. Since the quality of motion field directly affects the recognition performance, accurate motion field generation is of prime importance.

The details of motion field generation are illustrated in Fig. 2, which are explained in the following.

### A. Directional Edge Map Generation

In the first place, source image sequence is converted to a sequence of horizontal and vertical binary edge maps. Directional edge detection is performed by applying horizontal and vertical filtering kernels to an input image as shown in Fig. 3(b). Binarization of the filtering result is carried out by utilizing the threshold adaptively determined. The threshold is determined so that a certain number of edges are left in the frame. Using higher threshold decreases the number of edges left in the frame. Thus, by specifying the ratio of the total edge count to the total number of pixels in the frame, the threshold is determined. This is based on the fact that human pays more attention to more visible edges (or larger gaps of intensities) in the scene. The directional edge maps thus generated are shown in Fig. 3.

### B. Adaptive Frame Interval Adjustment

The motion estimation based on binary edge maps is very sensitive to the choice of frame intervals. If the interval between two frames is too small and the motions are less than one pixel, they cannot be detected. If the interval is too

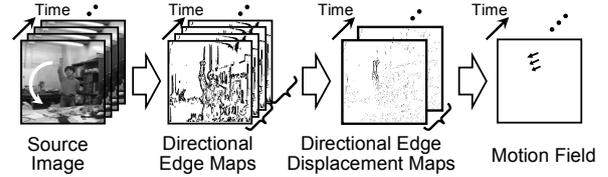


Figure 2. Detailed procedure of motion field generation.

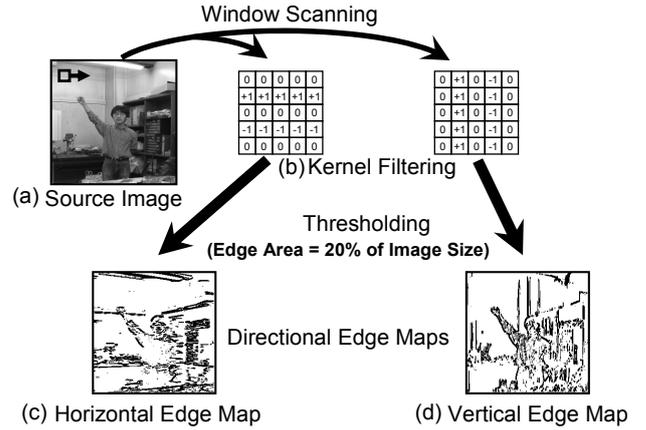


Figure 3. Generation of edge maps: (a) source image; (b) filtering kernels; (c) horizontal edge map; (d) vertical edge map at the edge area percentage of 20%.

large and the motion is beyond the range of search window ( $\pm$  eight pixels), it leads to an error. Therefore, suitable intervals must be determined adaptively according to the motion of the target object. For this end, edge-based adaptive interval adjustment scheme is proposed in the present paper. The procedure is explained in Fig. 4.

Firstly, an edge map is generated from an image frame by setting the edge detection threshold so that the number of edge flags remaining becomes a certain percentage of the total number of pixels in the frame. (Typically 20% in a 256x256 pixel size was employed.) This edge map is called the source edge map. Then, the edge map is generated from the next-frame image with the same remaining edge percentage (20% in the typical case) and logic OR is taken between the edge map and the source edge map. The combined edge map thus produced we call the *accumulated edge map*. From the third frame image, edges are detected with the same edge percentage, which is then merged with the accumulated edge map by taking OR again. In this manner, the accumulated edge map is updated at every incoming image frame. Such a procedure is repeated until the edge count in the accumulated edge map increases to a certain value (typically 22% was employed). Then the resultant accumulated edge map is compared with the source edge map and the difference is detected, namely, Exclusive OR is taken between the accumulated map and the source map. The edge map produced in this manner is called the *directional edge displacement map* (DED map), in which only the edge flags due to the motion occurring during the

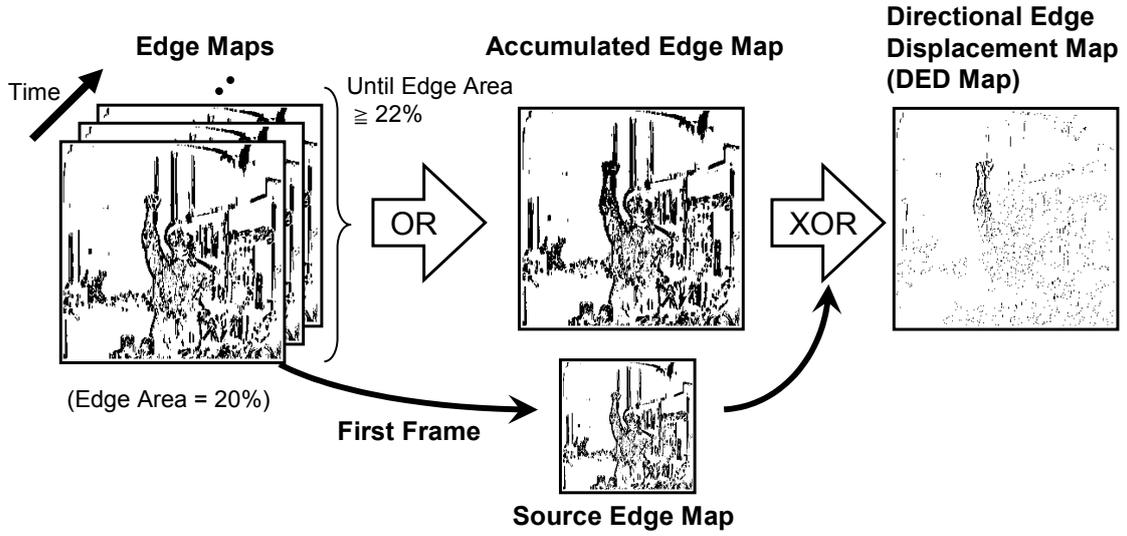


Figure 4. Adaptive frame interval adjustment and directional edge displacement map generation.

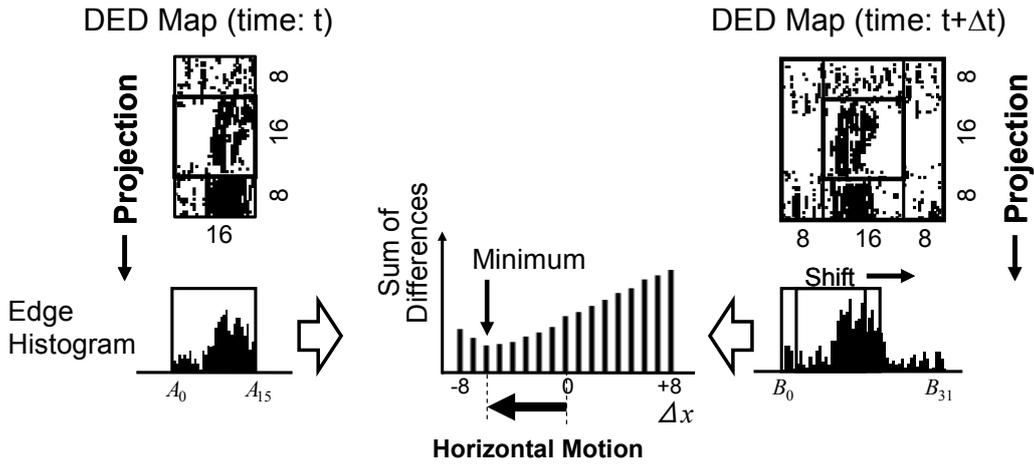


Figure 5. Horizontal motion detection by shifting and matching of edge histograms generated by projecting edge flags in DED map.

image sequence are accumulated and remaining. This very edge map is utilized in the motion field generation.

The same procedure is restarted from the following frame image, and repeated until the next DED map is obtained. From these two DED maps, motion field is generated as explained in the next subsection. Since the number of edge flags remaining in the DED map is set to a constant value (2% in the typical case), the time interval between the two DED maps are automatically adjusted as approximately corresponding to the equal amount of motion occurring in the image sequence.

### C. Motion Field Generation

Motion estimation using two DED maps are performed by block matching of edge images where computational cost was drastically reduced by using projected histograms

instead of using two dimensional edge maps. The procedure is illustrated in Fig. 5. In the following explanation, horizontal motion detection is described as an example. Vertical motion detection is carried out in the same manner. A block of 16x16 pixels in the first map at  $t$  is searched in the area of 32x32 pixels in the second map at  $t+\Delta t$ . The block in the map is vertically extended to 32x16 pixels and projected onto the horizontal axis, thus generating a 16-element histogram (edge histogram) by counting the number of edges in each column. Then the edge histogram at time  $t$  is compared with the 32-element edge histogram which is generated by projection of 32x32 pixels in the second map (time:  $t+\Delta t$ ). The sum of differences between two histograms is calculated for each shift of the histogram at  $t$  in the target histogram at  $t+\Delta t$  and the minimum in the sum of differences is searched. This yields the best match location. The

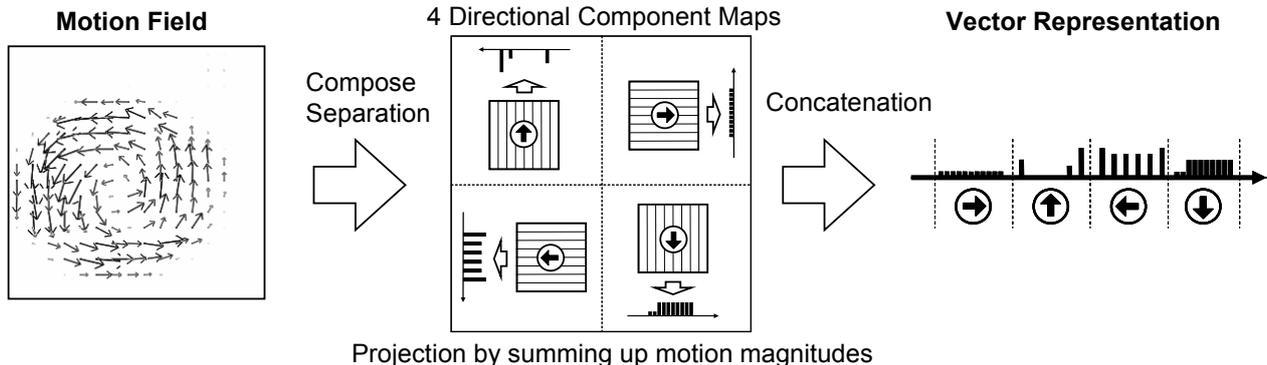


Figure 6. Feature vector representation of motion field: four directional motion components are summed up and represented as four concatenated histograms.

horizontal motion is determined using both vertical and horizontal edge maps.

In the motion estimation, unreliable block matching results should be ignored. One is the source block that contains too little edges. Therefore the results of edge-histograms whose average element values are smaller than one are discarded. Another unreliable result is the case where there exist multiple minima having closer values. So the matching result is discarded unless the minimum in the sum of differences is smaller than a half the average of sums of differences for all shifts. The motion estimation in the horizontal direction is performed using both horizontal and vertical edge maps. If one of the two results is zero or discarded as invalid, the other result is adopted as the final result. If the two results are different, then the results are ignored as invalid.

By repeating the processing at predetermined pixel locations, a motion field is generated from two DED maps generated at different time stamps.

#### D. Vector Representation of Motion Field

An example of generating a feature vector from a motion field is illustrated in Fig. 6. Motion field is separated into four directional motion component maps (positive and negative components in horizontal and vertical motions, namely right and left motions and up and down motions). These component maps are projected onto the axes perpendicular to the directions of respective motions, and four histograms are generated. Thus the elements in a vector represent the spatial distribution of four directional motions with somehow reduced spatial resolution in the direction of motion. Such a vector representation of the motion field is named as PPMD (Projected Principal-Motion Distribution) following the directional edge based still image representation known as PPEd (Projected Principal-Edge Distribution) [16].

### III. EXPERIMENTAL RESULTS AND DISCUSSION

Preliminary experiments of motion recognition were carried out based on the algorithm developed in the present work. The sequence of motion field generated from the source image sequence was converted to a sequence of vectors. Then these vector sequences were inputted into HMM recognition system as shown in Fig. 1.

Samples were composed of four cyclic motions, “horizontal”, “circle”, “diagonal” and “triangle” as shown in Fig. 7. The sequence of mirror images (horizontally reflected images) were also included as confusing test samples. Every motion was performed 4~5 times by each of three different individuals, thus 28 samples were collected for each motion. Therefore 112 image sequences (each contains 506 to 669 256x256-pixel grayscale frames) were used for the

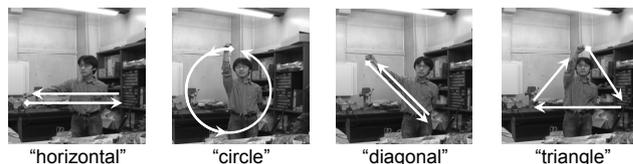


Figure 7. Four kinds of cyclic motions and their reflected images are used as sample sequences.

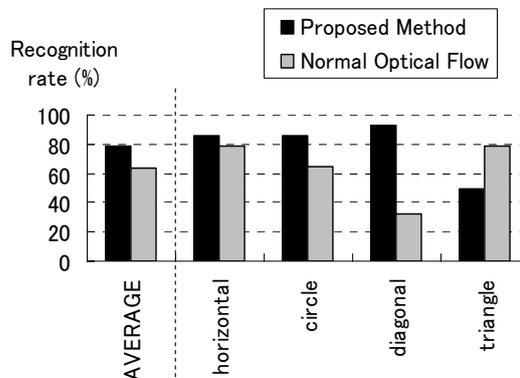


Figure 8. Recognition rate of motion recognition experiments.

experiments. A 6-state 1-dimensional left-right model with three Gaussian mixture was employed in the HMM. The samples were converted to sequences of 16-element vectors using two methods: the proposed method in this paper and the one employing normal optical flow for motion field generation. For each kind of motion, samples from one person were used as test data and the samples from the other two persons were utilized as learning data. The recognition test was repeated for all combinations.

The recognition rate for each kind of motion is illustrated in Fig. 8. On average, 78.5% recognition rate was achieved by the proposed method while the results using normal optical flow were 63.4%. The recognition rate was degraded due to the differences in individuals, especially differences in the arm length and the center position of motion. When three samples were randomly selected from the same motion group and utilized as test data, and the rest samples were utilized for learning, 100% recognition rate was achieved by proposed method while 98.2% using normal optical flow on average. To investigate the difference in the results between motion field generation methods, the quality of motion field sequence was evaluated as in the following.

For this purpose, an average trajectory of moving object was reconstructed. All the motion vectors in a frame were summed up to produce one pair of average horizontal and vertical motion components ( $v_x$ ,  $v_y$ ). Then, all motion components from a sequence were integrated and represented as a trace on an x-y plane as shown in Fig. 9. Here the length of the trace was normalized to 1.0. Fig. 9 draws two traces of the same sample “circle” using motion field sequences generated using either the proposed method or the optical flow. The end point of normal optical flow is far away from the starting point. This means that the motion field was not correctly detected. It is difficult to recognize

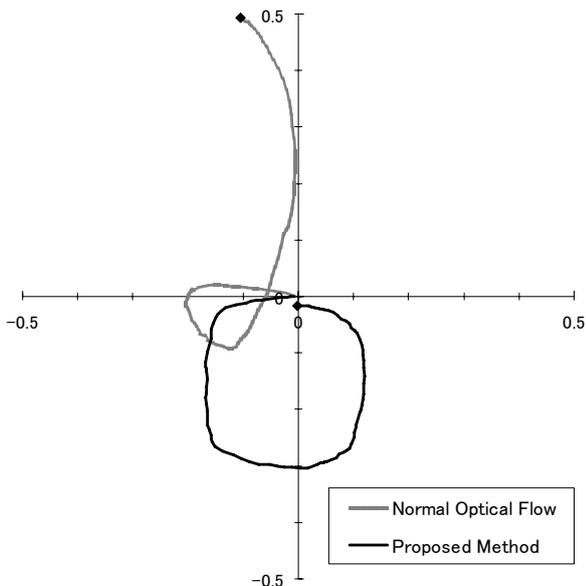


Figure 9. Reconstructed average trajectories from motion field sequence of “circle” motion.

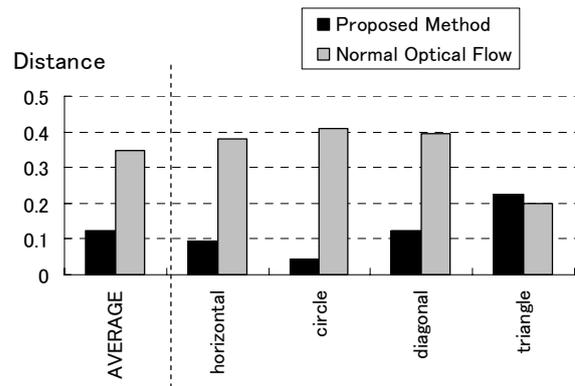


Figure 10. Distance from starting point to end point in reconstructed average trajectory for four kinds of cyclic motions.

that the trace draws a circle. On the contrary, the trace using the proposed method is obviously drawing a circle. The distance between the starting point and ending points was calculated for all samples and the results are shown in Fig. 10. The long distance observed for sample “triangle” was caused by a large variation in the motion depending on individuals. Figs. 8 and 10 indicate strong relationship between the recognition rate and the quality of motion field.

Finally, computational cost was examined. The operation time to generate one frame of motion field (motion vector generation at every pixel site of a 256x256-pixel image) took 17.8 seconds using proposed method while 72 milliseconds using normal optical flow on 2.0GHz dual core AMD Opteron processor. This means proposed method is about 250 times computationally more expensive as compared with normal optical flow. By employing adaptive frame interval adjustment method, 5.8 frames were skipped on average. As a result, the proposed method takes 43 times longer time than normal optical flow for total operation.

In the proposed method, about 96% of operation time was consumed by generating edge histograms, shifting and matching. To accelerate the operation, edge histogram matching processor has been developed also in our group [17]. Since this processor is capable of generating one motion vector per cycle at 100MHz clock, the operation time to generate one frame of motion field takes only 0.65 milliseconds (or about 770 motion field/second considering edge detection operation) which is sufficiently fast to carry out motion recognition.

#### IV. SUMMARY

Block-matching-based motion field generation algorithm utilizing directional edge displacement has been developed. Adaptive frame interval adjustment scheme introduced in this work has enabled a reliable motion field generation. Preliminary recognition experiments confirm the effectiveness of the method. The vector representation is now being applied to more complicated motions aiming at developing motion recognition systems.

#### ACKNOWLEDGMENT

The authors would like to thank Mio Nishiyama of The University of Tokyo for her contribution in building HMMs.

#### REFERENCES

- [1] H. Fei and I. Reid, "Joint Bayes filter: a hybrid tracker for non-rigid hand motion recognition," European Conference on Computer Vision 2004, Vol. I, pp. 495-507, 2004.
- [2] T. E. Starner and A. Pentland, "Visual recognition of American sign language using hidden Markov models," in Proceedings of International Workshop on Automatic Face and Gesture Recognition, pp. 189-194, June 1995.
- [3] J. Yamato, J. Ohya and K. Ishii, "Recognizing human action in time-sequential images using hidden Markov model," in Proceedings of Computer Vision and Pattern Recognition 1992, pp. 379-385, June 1992.
- [4] C. Shan, Y. Wei, X. Qiu and T. Tan, "Gesture recognition using temporal template based trajectories," in Proceedings of International Conference on Pattern Recognition 2004, Vol. 3, pp. 954-957, August 2004.
- [5] C. R. Wren, A. Azarbayejani, T. Darrell and A. P. Pentland, "Pfinder: real-time tracking of the human body," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, Iss. 7, pp. 780-785, July 1997.
- [6] P. Kahn, "Integrating moving edge information along a 2D trajectory indensely sampled imagery," in Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 702-709, June 1988.
- [7] A. Niyogi and E. H. Adelson, "Analyzing and recognizing walking figures in XYT," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition 1994, pp. 469-474, June 1994.
- [8] A. Bobick and J. Davis, "Real-time recognition of activity using temporal templates," in Proceedings of Applications of Computer Vision 1996, pp. 39-42, December 1996.
- [9] R. Polana and R. C. Nelson, "Recognition of motion from temporal texture," in Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition 1992, pp. 129-134, June 1992.
- [10] M. J. Black and Y. Yacoob, "Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion," in Proceedings of International Conference on Computer Vision 1995, pp. 374-381, June 1995.
- [11] B. K. P. Horn and B. G. Schunck, "Determining optical flow," Artificial Intelligence, Vol. 17-1-3, pp. 185-203, August 1981.
- [12] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in Proceedings of the 7th International Joint Conference on Artificial Intelligence, pp. 674-679, August 1981.
- [13] J. H. Duncan and T. C. Chou, "On the detection of motion and the computation of optical flow," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, Iss. 3, pp. 346-352, March 1992.
- [14] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," IEEE Transactions on Image Processing Vol.9, Iss.2, pp. 287-290, February 2000.
- [15] L. M. Po and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 6, Iss. 3, pp. 313-317, June 1996.
- [16] M. Yagi and T. Shibata, "An image representation algorithm compatible with neural-associative-processor-based hardware recognition systems," IEEE Transactions on Neural Networks, vol. 14, no. 5, pp.1144-1161, September 2003.
- [17] K. Fujita, K. Ito and T. Shibata, "A feature-based optical flow processor architecture featuring single-motion-vector /cycle generation," accepted for presentation at International Symposium on System-on-Chip 2007, November 2007.