An Attention Based Method For Motion Detection And Estimation

Shijie Zhang[†] and Fred Stentiford[‡]

Department of Electronic and Electrical Engineering
University College London, Adastral Park Campus, Ross Building
Martlesham Heath, Ipswich, IP5 3RE, UK
{[†] j.zhang | [‡] f.stentiford}@adastral.ucl.ac.uk

Abstract: The demand for automated motion detection and object tracking systems has promoted considerable research activity in the field of computer vision. A novel approach to motion detection and estimation based on visual attention is proposed in the paper. Two different thresholding techniques are applied and comparisons are made with Black's motion estimation technique [1] based on the measure of overall derived tracking angle. The method is illustrated on various video data and results show that the new method can extract both motion and shape information.

Keywords: visual attention, motion detection, motion estimation, object tracking

1. Introduction and Background

The demand for automated motion detection and object tracking systems has promoted considerable research activity in the field of computer vision [2-6]. Bouthemy [2] proposed a novel probabilistic parameter-free method for detecting independently moving objects using the Helmholz principle. Optical flow fields were estimated without making assumptions on motion presence and allowed for possible illumination changes. The method imposes a requirement on the minimum size for the detected region and detection errors arise with small and low contrasted objects. Black and Jepson [3] proposed a method for optical flow estimation based on the motion of planar regions plus local deformations. The approach used brightness information for motion interpretation by using segmented regions of piecewise smooth brightness to hypothesize planar regions in the scene. The proposed method has problems dealing with small and fast moving objects. It is also computational expensive. Black and Anandan [1] then proposed a framework based on robust estimation that addressed violations of both brightness constancy and spatial smoothness assumptions caused by multiple motions. It was applied to two common techniques for optical flow estimation: the area-based regression method and the gradient-based method. To cope with motions larger than a single pixel, a coarse-to-fine strategy was employed in which a pyramid of spatially filtered and sub-sampled images was constructed. Separate motions were recovered using estimated affine motions,

however, the method is relatively slow. Viola and Jones [4] presented a pedestrian detection system that integrated both image intensity (appearance) and motion information, which was the first approach that combined motion and appearance in a single model. The system works relatively fast and operates on low resolution images under difficult conditions such as rain and snow, but it does not detect occluded or partial human figures. In [5] a method for motion detection based on a modified image subtraction approach was proposed to determine the contour point strings of moving objects. The proposed algorithm works well in real time and is stable for illumination changes. However, it is weak in areas where a contour appears in the background which corresponds to a part of the moving object in the input image. Also some of the contours in temporarily non-moving regions are neglected in memory so that small broken contours may appear. In [6] the least squares method was used for change detection. The proposed approach is efficient and successful on image sequences with low SNR and is robust to illumination changes. The biggest shortfall is that it can only cope with single object movements because of the averaging nature of least square method.

Burt [7] first proposed using visual attention for tracking and high level interpretation. Tracking starts with the estimation of background motion based on the correlation of two successive frames, performed at reduced resolution for computational efficiency. A pyramid is constructed for each frame and correlation is repeated to form a refined estimate of the motion. Rather than correlate each frame as a whole, it is divided into regions and displacement vectors are computed separately. The vectors are averaged to obtain the estimate of background motion discarding those that were most different from the majority due to the local object movement. The method uses a simple model to estimate only the translational background motion. Tsotsos [8] proposed a feed-forward motion processing hierarchy based on neurobiology of primate motion processes. A selective tuning (ST) model for visual attention was demonstrated on this structure to localize and label motion patterns. Motion features extracted are grouped using a top-down attentional selection mechanism. The ST method provides a distributed definition of saliency different from other models based on Koch and Ullman [9].

The use of visual attention (VA) methods [10-12] to define the foreground and background information in a static image for scene analysis has motivated this investigation. We propose in this paper that similar mechanisms may be applied to the detection of saliency in motion and thereby derive an estimate for that motion.

2. Motion VA Algorithm

Methods for identifying areas of static saliency are described in [11] and are based upon the premise that regions which are largely different to most of the other parts of the image will be salient and will be present in the foreground. Such discrimination between foreground and background can be dependent upon features such as colour, shape, texture, or a combination. This concept has been extended into the time domain and is applied to

sequences of video frames to detect salient motion. The approach is not dependent on a specific segmentation process but only upon the detection of anomalous movements. The method estimates the shift by obtaining the distribution of displacements of corresponding salient features.

In order to reduce computation candidate regions of motion are first detected by generating the intensity difference frame from adjacent frames and applying a threshold.

$$I_x = \{|r_2 - r_1| + |g_2 - g_1| + |b_2 - b_1|\}/3 . \quad (1)$$

Where parameters (r_1, g_1, b_1) & (r_2, g_2, b_2) represent the rgb colour values for pixel $\boldsymbol{x}$ in frames 1 and 2. The intensity I_x is calculated by taking the average of the differences of rgb values between the two frames.

The candidate regions R_1 in frame 1 are then identified where $I_x > T$ and T is a threshold determined by intensity.

Let a pixel $\boldsymbol{x} = (x, y)$ in R_t correspond to colour components $\boldsymbol{a} = (r, g, b)$.

Let $\boldsymbol{F(x)} = \boldsymbol{a}$. and let $\boldsymbol{x_0}$ be in R_t in frame t.

Consider a neighbourhood G of $\boldsymbol{x_0}$ within a window of radius ε where

$$\{\boldsymbol{x_i'} \in G \quad iff \quad |x_0 - x'| \le \varepsilon\}. \quad (2)$$

Select an n-tuple of m random points S_x in G where

$$S_x = \{\boldsymbol{x_1'}, \boldsymbol{x_2'}, ..., \boldsymbol{x_m'}\}. \quad (3)$$

We also only consider n-tuples which are constrained to contain pixels that mismatch each other. This means that n-tuples will be selected in image regions possessing high or certainly non-zero VA scores, such as on edges or other salient features.

In this case there will be at least one pixel in the n-tuple that differs by more than δ in one or more of its rgb values with one or more of the other pixels in the n-tuple i.e.

$$|F_k(\boldsymbol{x_i'}) - F_k(\boldsymbol{x_j'})| > \delta_k \quad for\ some\ i, j, k . \quad (4)$$

Define the radius of the region within which n-tuple comparisons will be made as V (the *view radius*)

Randomly select another location $\boldsymbol{y_0}$ in the adjacent frame R_{t+1} within a radius V of $\boldsymbol{x_0}$.

Define the n-tuple

$$S_y = \{\boldsymbol{y_1'}, \boldsymbol{y_2'}, ..., \boldsymbol{y_m'}\} \text{ where } \boldsymbol{x_0} - \boldsymbol{x_i'} = \boldsymbol{y_0} - \boldsymbol{y_i'} . \quad (5)$$

$$\text{and } |\boldsymbol{y_0} - \boldsymbol{x_0}| \le V .$$

S_y is a translated version of S_x .The n-tuple centred on $\boldsymbol{x_0}$ is said to match that at $\boldsymbol{y_0}$ (S_x matches S_y) if all the colour components of corresponding pixels are within a threshold.

$$|F_k(\boldsymbol{x_i'}) - F_k(\boldsymbol{y_i'})| \le \delta_k \quad k = r, g, b \quad i = 1,2,...,m. \quad (6)$$

N attempts are made to find matches and the corresponding displacements are recorded as follows:

For the *j*th of $N_1 < N$ matches define the corresponding displacement between $\boldsymbol{x_0}$ and $\boldsymbol{y_0}$ as $\boldsymbol{\sigma_j^{t+1}} = (\sigma_p, \sigma_q)$ where

$$\sigma_p = \left| x_{0p} - y_{0p} \right|, \quad \sigma_q = \left| x_{0q} - y_{0q} \right|. \tag{7}$$

and the cumulative displacements Δ and match counts Γ as

$$\left.\begin{aligned} \Delta(\boldsymbol{x_0}) &= \Delta(\boldsymbol{x_0}) + \boldsymbol{\sigma_j^{t+1}} \\ \Gamma(\boldsymbol{x_0}) &= \Gamma(\boldsymbol{x_0}) + 1 \end{aligned}\right\} \; j = 1, \ldots, N_l < N. \tag{8}$$

where N_1 is the total number of matching n-tuples and N is the total number of matching attempts.

The displacement $\overline{\boldsymbol{\sigma}}_{\boldsymbol{x_0}}^{t+1}$ corresponding to pixel $\boldsymbol{x_0}$ averaged over the matching n-tuples is

$$\overline{\boldsymbol{\sigma}}_{\boldsymbol{x_0}}^{t+1} = \frac{\Delta(\boldsymbol{x}_0)}{\Gamma(\boldsymbol{x}_0)}. \tag{9}$$

A similar calculation is carried out between R_t and R_{t-1} (swapping frames) to produce $\overline{\boldsymbol{\sigma}}_{\boldsymbol{x_0}}^{t-1}$ and the estimated displacement of $\boldsymbol{x_0}$ is given by $\{\overline{\boldsymbol{\sigma}}_{\boldsymbol{x_0}}^{t+1} - \overline{\boldsymbol{\sigma}}_{\boldsymbol{x_0}}^{t-1}\}/2$. This estimate takes account of both trailing and leading edges of moving objects.

This is carried out for every pixel $\boldsymbol{x_0}$ in the candidate motion region R_t and M attempts are made to find an internally mismatching n-tuple S_x.

3. Results and Discussion

3.1 Motion Detection and Estimation

3.1.1 Shape Property

The approach allows some information about the shape of the moving object to be extracted. Figure 1 shows two red squares in each frame, with the upper one moving from left to right and the other remaining static. Two motion vector maps for both the moving square and the static square are shown alongside. The nature of the n-tuple matching process causes a displacement component to be generated that points in the direction of the curvature of the boundary of the object if there are no other salient features. Motion vectors near corners tend therefore to point towards the centre of the plain red squares in Figure 1. This "static motion" component may be subtracted to obtain vectors purely dependent upon motion or retained to assist in the characterisation of a tracked object, in which case certain parts of the object may appear to be travelling at slightly different speeds.

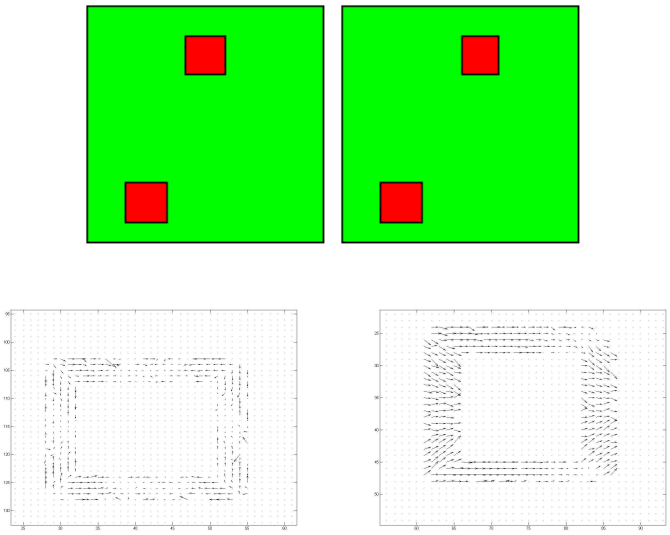

Fig. 1. Synthetic data frames and motion vectors for static and moving squares.

3.1.2 Road scenes

A pair of 352x288 frames from a traffic video was tested on with results shown in Figure 2. The intensity difference indicates the areas of candidate motion for subsequent analysis. Motion vectors were calculated as above for each pixel in the car region and plotted in Figure 3. A map for motion magnitudes in the y direction is shown in which colours represent magnitudes as indicated in the colour bar. The directions of pixel motion is also represented by colours in the bar where angles are measured anticlockwise from the vertical e.g. yellow indicates a direction towards the top left. Motion vectors are not assigned if no internally mismatching n-tuples can be found e.g. in areas of low saliency. This also means the motion of plain unchanging interiors of such a region will not be tracked. The processing took 0.23 seconds in C++.

The parameters of the experiment were M = 100, N = 100, $\varepsilon = 3$, m = 7, V = 10, $\boldsymbol{\delta}$ = (40,40,40), T = 12.6. T is set to be twice the standard deviation of the values in the matrix I_x.

Fig. 2. Two adjacent frames and their intensity difference

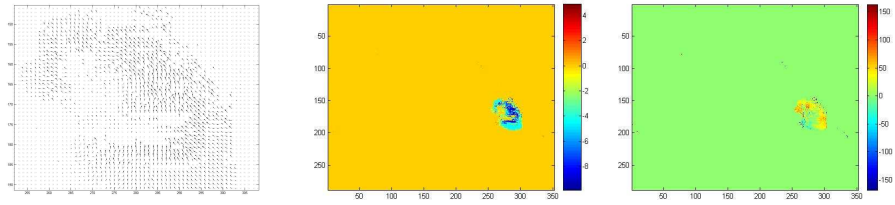

Fig. 3. Motion vector map, y direction magnitude map, and angle map corresponding to frames in Fig. 2.

Two pairs of frames from the same traffic video were tested on with results shown in Figure 4 and Figure 5. Both figures include a magnitude map and an angle map for 3-car and 4-car scenario respectively.

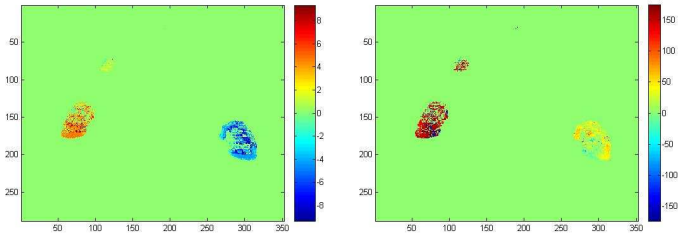

Fig. 4. Y direction magnitude map and angle map (3-car)

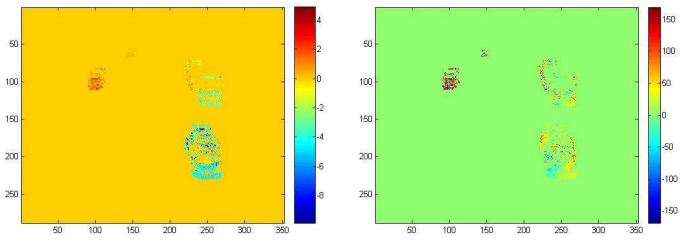

Fig. 5. Y direction magnitude map and angle map (4-car)

3.2 Object Tracking

The method was compared with Black's motion estimation technique based on the weighted object tracking angle θ defined as follows

$$\theta = \frac{\sum \left(MI^2 \times AI \right)}{\sum MI^2} \tag{10}$$

where *MI* is magnitude of the motion of a pixel, and *AI* is the angle of the direction of motion of the pixel. A squared weighting was used so that motion vectors with higher values have a bigger influence on the weighted angle as they are likely to be more reliable.

Colour images are used in contrast to the greyscale images used by Black because they provide more information for the comparison of regions. A new threshold δ for pixel matching was also compared based on the joint Euclidean distance of RGB colour channels rather than treating the channels separately as in (6).

In the case of pixel mismatching, there will be at least one pixel in the n-tuple that differs by more than δ with one or more of the other pixels in the n-tuple according to

$$\sqrt{\sum_k \left(F_k(\boldsymbol{x}_i') - F_k(\boldsymbol{x}_j') \right)^2} > \delta \quad \textit{for some } i, j \tag{11}$$

The n-tuple centred on $\boldsymbol{x_0}$ is said to match that at $\boldsymbol{y_0}$ (S_x matches S_y) if the Euclidean distance between corresponding n-tuple pixels are all less than δ

$$\sqrt{\sum_k \left(F_k(\boldsymbol{x}_i') - F_k(\boldsymbol{y}_i') \right)^2} \leq \delta \quad \forall\, i \tag{12}$$

Figure 6 illustrates a comparison between the weighted tracking angles derived from Black's software and those derived from the motion VA algorithm using the separate and joint metrics, . The parameters of the experiment were M = 100, N = 1000, $\varepsilon = 3$, V = 10,

$\boldsymbol{\delta}$ = (40,40,40), T = 12.6, and the calculations were repeated for different numbers of n-tuple pixels (m) on the same frame from 2 to 49 (fully filled 7x7 n-tuple). The ground truth angle of the car was measured to be $36° \pm 5°$. As is shown in the figure, both angle results produced by the motion VA algorithm give closer estimates than Black ($= 24.6°$). In both cases the weighted angle increases as extra pixels are added into the n-tuple with the separate colour channel metric performing better. Increased accuracy and precision are achieved at the expense of addition computation which increases with m. The improvements diminish beyond 15 n-tuple pixels.

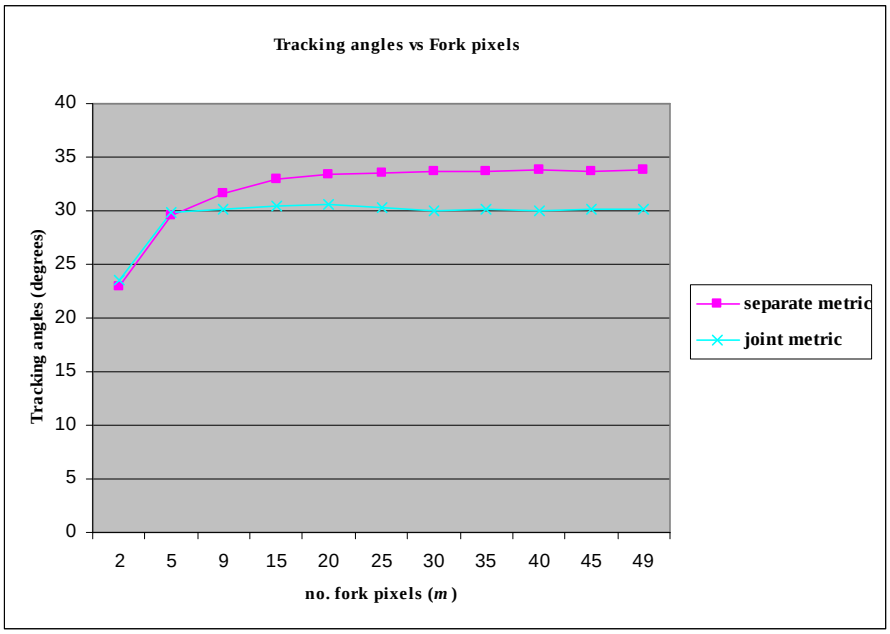

Fig. 6. Weighted tracking angles for Motion VA algorithm against numbers of n-tuple pixels and separate (6) and joint (12) distance metrics. Black's estimate is = $24.6°$.

N-tuple radii of ε = 2,3,4,5,6 (n-tuple sizes of 5x5 to 13x13) were then used with other parameters fixed to compare the performance on the angle estimates for motion VA algorithm using the separate channel metric. The results illustrated in Figure 7 shows that a slightly better estimate can be obtained with bigger n-tuple radii but there is little improvement above 15 pixels. Also the results converge as the number of n-tuple pixels increase.

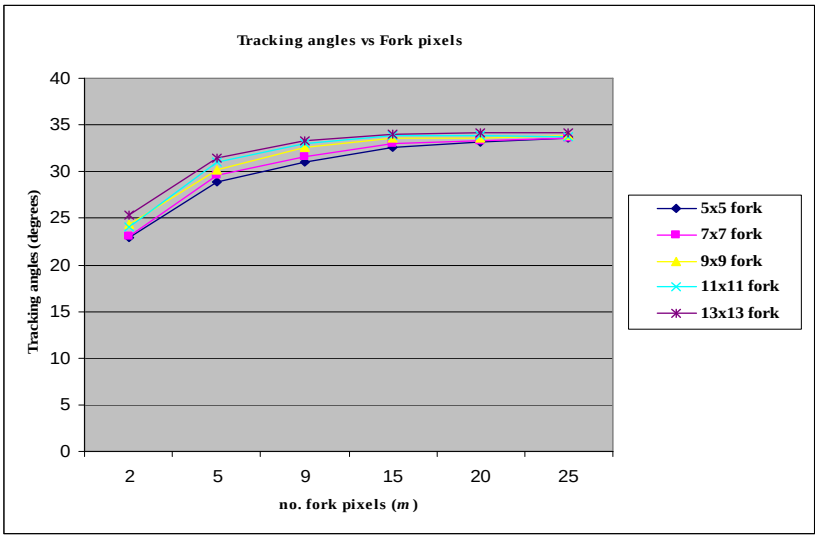

Fig. 7. Weighted tracking angles for Motion VA algorithm using the separate channel metric (6)

3.3 Football Data

The algorithm was also illustrated on football data. Figure 8 shows a pair of 352x288 frames used to estimate the motion vector map for one player in the scene. The weighted tracking angle θ was calculated to be $98.7°$ as compared to the actual angle of $100°$. The parameters were M = 100, N = 10000, $\varepsilon = 3$, m = 2, V = 40, δ = (40,40,40), T = 30. The processing took 17 seconds in C++. The number of n-tuple pixels (m) was set to 2 small to maximise the number of matches. V was increased to accommodate the larger movement between frames. A lower limit for N was determined by an empirical formula which increases with both the radius of the n-tuple ε and the view radius V given by

$$N \geq [2 \times (\varepsilon + V)]^2 \ . \qquad (13)$$

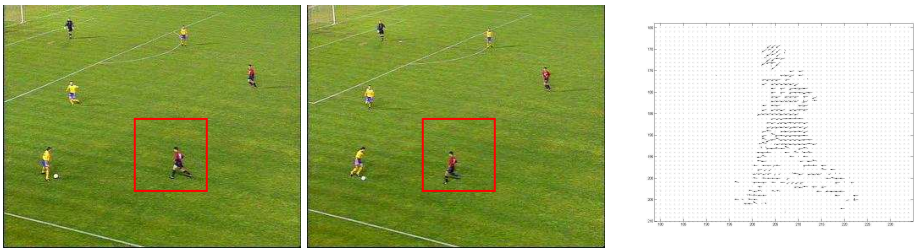

Fig. 8. Football frames and motion vector map

4. Conclusions

An attention based mechanism has been proposed for motion detection and estimation. The approach measures motion by comparing salient regions between video frames. It is also able to extract certain shape information as part of the comparison process.
The method was illustrated on various video data and different thresholding criteria and was shown to obtain a better estimate of motion direction than Black's technique. The stability of the results is dependent on the volume of processing, but the elements of the computation are simple and amenable to parallel implementation. The method does not require a training stage or prior knowledge of the objects to be tracked.
Future work will be carried out on a greater diversity of data with particular emphasis on addressing background motion and changes in illumination.

Acknowledgement: The project is sponsored by European Commission Framework Programme 6 Network of Excellence MUSCLE (Multimedia Understanding through Semantics, Computation and Learning) [13].

References

1. Black, M.J., Anandan, P.: The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. CVIU, Vol. 63, Issue 1 (1996) 75-104
2. Veit, T., Cao, F., Bouthemy, P.: Probabilistic parameter-free motion detection. CVPR, Vol. 1 (2004) 715-721
3. Black, M.J., Jepson, A.D.: Estimating optical flow in segmented images using variable-order parametric models with local deformations. IEEE Trans. on PAMI, Vol. 18, Issue 10 (1996) 972-986
4. Viola, P., Jones, M.J., Snow, D.: Detecting pedestrians using patterns of motion and appearance. ICCV, Vol. 2 (2003) 734-741
5. Kellner, M., Hanning, T.: Motion detection based on contour strings. ICIP, Vol. 4 (2004) 2599-2602
6. Xu, M., Niu, R., Varshney, P.K.: Detection and tracking of moving objects in image sequences with varying illumination. ICIP, Vol. 4 (2004) 2595-2598
7. Burt, P.J.: Attention mechanisms for vision in a dynamic world. ICPR, Vol. 2 (1988) 977-987
8. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Trans. on PAMI, Vol. 20, Issue 11 (1998) 1254-1259
9. Tsotsos, J.K., Liu, Y., Martinez-Trujillo, J.C., Pomplun, M., Simine, E., Zhou, K.: Attending to visual motion. CVIU 100 (2005) 3-40
10. Koch, C., Ullman, S.: Shifts in selective visual attention: towards the underlying neural circuitry. Hum. Neurobiol. 4 (1985) 219-227
11. Stentiford, F.W.M.: An estimator for visual attention through competitive novelty with application to image compression. Picture Coding Symposium, Seoul (2001) 101-104
12. Stentiford, F.W.M.: Attention based similarity. Pattern Recognition 40 (2007) 771-783
13. Multimedia Understanding through Semantics, Computation and Learning, 2005. EC 6th Framework Programme, FP6-507752, http://www.muscle-noe.org/