Comparison of Fusion Methods for Thermo-Visual Surveillance Tracking

C. Ó Conaire, N. E. O'Connor, E. Cooke, A. F. Smeaton
Adaptive Information Cluster,
Dublin City University,
Ireland.

Abstract - ***In this paper, we evaluate the appearance tracking performance of multiple fusion schemes that combine information from standard CCTV and thermal infrared spectrum video for the tracking of surveillance objects, such as people, faces, bicycles and vehicles. We show results on numerous real world multimodal surveillance sequences, tracking challenging objects whose appearance changes rapidly. Based on these results we can determine the most promising fusion schemes.***

Keywords: Tracking, Visible Spectrum, Thermal Infrared, Fusion, Surveillance.

1 Introduction

World events have ensured that security and surveillance have received much research attention in recent years. The desire to provide robust and accurate surveillance information has led to considerable research on methods to integrate information from different sensors. This has the potential to provide more robust systems by leveraging the combined benefits of using different modalities whilst compensating for failures in individual modalities.

In our work, we investigate the advantages of capturing thermal infrared video in parallel with standard visible spectrum CCTV, for tracking objects in surveillance scenarios using appearance models. These sources are intuitively complementary, since they capture object information in emitted and reflected radiation, respectively. By utilising both complementary sources of data, we obtain improved robustness against camouflage, as foreground objects are less likely to be of a similar colour *and* temperature to the background. Using multiple sources also provides additional features to assist the tracker in cluttered tracking environments. This paper evaluates various fusion schemes and similarity metrics for multi-modal tracking using appearance models. We evaluate tracking performance on real data using manually annotated ground truth.

This paper is organised as follows: In section 2, we provide a brief background literature review to contextualise our work. Section 3 describes the hardware we used to capture our test data and how the data from both cameras was aligned. The appearance model we use for tracking is detailed in section 4, along with the similarity metrics we use. In section 5, we outline the different fusion schemes that we evaluated. We present results in section 6, comparing the tracking performance of the evaluated fusion schemes on real surveillance data, and finally give our conclusions and directions for future work in section 7.

2 Related work

In a review of video surveillance and sensor networks research [1], Cucchiara argues that the integration of video technology with sensors and other media streams will constitute the fundamental infrastructure for new generations of multimedia surveillance systems. Also reviewing surveillance research [2], Hu et al. conclude in their section on Future Developments in Surveillance that *'Surveillance using multiple different sensors seems to be a very interesting subject. The main problem is how to make use of their respective merits and fuse information from such kinds of sensors'*.

In the tracking literature, many approaches have been proposed to combine the information from multiple sources, in order to provide more accurate and robust detection and tracking. Probabilistic methods are commonly used to fuse information sources. In [3], Bayesian probability theory is used to fuse the tracking information available from a suite of cues to track a person in 3D space. A Bayesian tracking framework using particle filters is described in [4] for fusing colour cues with stereo or motion information. A Bayesian multi-object tracker is described in [5] that fuses binary information from foreground detection with colour tracking cues. Linear combinations of sources have also been widely used to fuse information from multiple sources. In [6], information from image segmentation is fused with chamfer matching scores to robustly detect people in cluttered images. Lim and Kriegman [7] use a linear combination of shape and appearance to track people in an indoor environment. Both [6] and [7] use fixed weighting for the data sources. In [8], the weightings for each tracking cue (colour and edge histograms) are adaptively updated using the Bhattacharyya coefficients. Fumera and Roli [9] consider linear combinations of classifiers and conduct a theoretical analysis, as well as performing experiments on real data sets. Their conclusions were that weighted average combinations usually only provide a marginal improvement over simple averaging, even with optimal weights. Recently, in [10], Ensemble Tracking was introduced as a general tracking framework to combine information from multiple sources. An *ensemble* of linear classifiers are trained online in a least-squares manner, to distinguish between object and background pixel features. This allows any type of pixel data to be added to assist the tracking. Kruppa and Schiele [11] fuse information from multiple object models by determining a configuration that maximises the mutual information between the models. In [12], Torresan et al. describe a surveillance system that fuses standard visible spectrum and thermal infrared video to detect and track pedestrians. They link foreground regions in consecutive frames and do not model the appearances of tracked objects, therefore their method requires many complex ad-hoc rules to account for the splitting and merging of foreground regions.

Occlusion handling is an important component in practical tracking algorithms for surveillance. In [13], all objects in the camera's field of view are tracked; appearance models and linear velocity prediction are used to cater for situations where objects occlude one another. In [14], occlusion is handled using robust statistics and occlusion is declared when over 15% of pixels are determined to be outliers. We do not specifically tackle the occlusion problem here, as we feel it is out of scope of this paper that focuses on evaluating fusion methods for tracking.

In [14], Zhou et al. introduce an adaptive model for robust appearance tracking. Using image brightness values in their appearance model, results are shown on tracking the rear end of a car, a frontal face and an aerial view of a tank. The objects they tracked do not alter significantly in appearance, although the pose does change. The appearance model we adopt in this paper is inspired by [14], but with significant differences; including the ability to track non-rectangular patches, integrate information from background modelling algorithms and perform rapid initialisation. Additionally, we introduce an alternative similarity metric to match the model to the image.

3 Data capture and alignment

3.1 Hardware

To obtain our raw video data, we use a Raytheon ControlIR 2000B thermal imaging video camera that is sensitive to wavelengths of 7μm-14μm, along with a Panasonic WV-CP470 colour video camera. The two cameras are synchronised (using *gen-lock*) to ensure that they capture frames simultaneously. Both channels of analogue video output are captured and digitised by a Falcon Quattro multi-channel framegrabber. Figure 1(a) shows the configuration of the visible and thermal cameras. A pane of thermally-reflective glass was used to act as a beam-splitter.

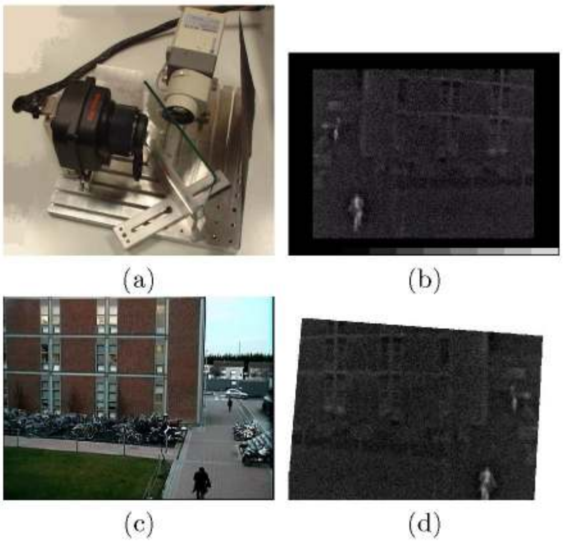


Figure 1: (a) Visible/Infrared camera rig, (b) Original infrared image, (c) Visible image, (d) Aligned infrared image

3.2 Frame alignment

To align pixels in the thermal and visible spectrum, we determine the optimum planar homography [15] and apply an image warping to all thermal infrared frames. This homography is determined by manually selecting many corresponding points in both modalities and computing the homography with least-squared error. There is no correlation between visible spectrum brightness and thermal infrared brightness values, so many of the automatic mutual-information based alignment methods [16][17] would not be appropriate. An automatic alignment technique that can be used for images of very different modalities (such as thermal and visible images) is proposed in [18], and relies on the correlation between edge orientations in the modalities. Currently, no automatic method is required, as only one warping needs to be computed for an entire sequence, but this could be a direction for future work. Figure 1(d) shows an example of an aligned infrared video frame.

4 Appearance model

4.1 Model description

The appearance model we use in this paper is inspired by [14] but we note here the significant differences. Firstly, we use a single multi-dimensional Gaussian for each pixel, and not a mixture of Gaussians. Secondly, we introduce a per-pixel importance weighting, to track non-rectangular patches and to integrate information from background subtraction (or motion detection) algorithms. We also use update equations based on expected sufficient statistics until enough samples are obtained to switch to the exponentially 'forgetting' equations, which allows faster adaptation at initialisation time. Additionally, we introduce an alternative similarity metric to match the model to the image.

The appearance of the object being tracked is modelled as a rectangular grid of d pixels, with each pixel modelled as having a Gaussian distribution. We denote $\mu(j) = \{\mu_1(j), \mu_2(j), ..., \mu_k(j)\}$ as the mean vector value of pixel j at time t, where k is the number of features. For example, $\mu_1(j)$ could correspond to the pixel's mean brightness value and $\mu_2(j)$ could correspond to the pixel's mean edge magnitude. We further denote $\sigma(j) = \{\sigma_1(j), \sigma_2(j), ..., \sigma_k(j)\}$ as the diagonal covariance matrix. This assumes the pixel features are independent. This assumption may not hold for certain features, such as a pixel's RGB colour components, but this can be overcome by switching to a less correlated colourspace (such as the CIE Lab colourspace) or by increasing the computational complexity and using a full covariance matrix. Additionally, we add a weighting factor for each pixel, $I(j)$, where $0 \leq I(j) \leq 1$. This weighting factor allows pixels that belong to the background to be removed from the similarity computation, while emphasising trackable features. We denote our model as $\theta = \{\mu, \sigma, I\}$.

4.2 Similarity measures

For a particular vector of pixel features, x, we can compute the probability that it matches the Gaussian model of model pixel j using:

$$P(x|\theta(j)) = \prod_{i=1}^{k} N(x_i; \mu_i(j), \sigma_i(j)^2) \tag{1}$$

where $N(x; \mu, \sigma^2)$ is a one dimensional normal density:

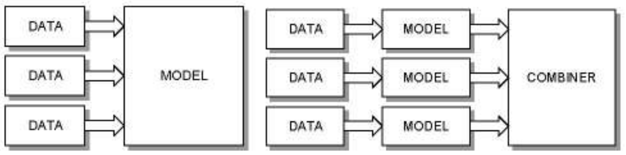
$$N(x; \mu, \sigma^2) = (2\pi\sigma^2)^{1/2} exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right) \tag{2}$$

To compute the similarity between the model and a rectangular patch of d pixels (denoted $X = \{x_1, x_2, ..., x_d\}$), [14] assumed the pixels were independent of each other and obtained the matching probability as:

$$P(X|\theta) = \prod_{j=1}^{d} P(x_j|\theta(j)) \tag{3}$$

Our pixel weighting factor can be incorporated by allowing it to represent the probability that the pixel belongs to the object (and not the background). The similarity measure then becomes:

$$S_1(X|\theta) = \prod_{j=1}^{d} (I(j)P(X_j|\theta(j)) + 1 - I(j)) \tag{4}$$



(a) GFM (b) CM

Figure 2: Fusion architectures

However, we found that this probability computation could be significantly affected by outliers, since a single bad pixel match could return a probability very close (or equal) to zero. Fortunately, the occlusion handling mechanism used in [14] provides robustness against outliers. The normal density of equation (1) is replaced by $\hat{N}(x; \mu, \sigma^2)$:

$$\hat{N}(x; \mu, \sigma^2) = (2\pi\sigma^2)^{1/2} exp\left(-\rho(\frac{x-\mu}{\sigma})\right) \tag{5}$$

where

$$\rho(x) = \begin{cases} x^2/2 & \text{if } |x| \leq c \\ c|x| - c^2/2 & \text{otherwise} \end{cases} \tag{6}$$

In our experiments, as in the original paper, we set $c = 1.435$, where c is an outlier threshold. Additionally, we define an alternative similarity measure, S_2, which is the average probability of a pixel matching its corresponding model pixel, weighted by the pixel importance values.

$$S_2(X) = \frac{\sum_{j=1}^{d} I(j)P(X_j|\theta(j))}{\sum_{j=1}^{d} I(j)} \tag{7}$$

4.3 Appearance model update

We update the model using expected sufficient statistics update equations then switch to L-recent window version after L updates, where L is a time constant that determines the update rate. The features for each pixel are updated independently using the following equations:

$$\sigma_{t+1}^2 = \alpha\sigma_t^2 + (1-\alpha)(\mu_t - x)^2 \tag{8}$$

$$\mu_{t+1} = \alpha\mu_t + (1-\alpha)x \tag{9}$$

$$I_{t+1} = \alpha I_t + (1-\alpha)f \tag{10}$$

where x is the current feature value, f is the binary foreground value for this pixel, and

$$\alpha = 1 - \frac{1}{\min(N, L)} \tag{11}$$

where N is the number of times the model has been updated. Additionally, we add resilience to noise by imposing the restriction that f is set to zero if the pixel is not adjacent to a pixel from the model whose importance value is greater than $1/L$. For all our experiments, we set $L = 20$, as suggested in [13].

5 Fusion models

In this paper, we are concerned with evaluating the tracking performance of multiple different fusion schemes. Figure 2 shows the two fusion architectures that we utilise. In the first scheme, which we will refer to as the *general fusion model*, the fusion occurs at the pixel level, by using a single appearance model where each pixel is modelled as a k-dimensional Gaussian, where k is the number of data sources (or features). In the second scheme, which we will refer to as a *combination of models*, the fusion occurs at the model level, by using a separate appearance model (with $k = 1$) for each data source and combining their scores. We examine a number of different methods of score fusion at the model level.

5.1 General fusion model

In the general fusion model(GFM), a single appearance model (grid of d pixels) is used, where each pixel is modelled as a k-dimensional Gaussian, where k is the number of data sources. Each pixel is assigned a single importance value that weights its importance in the overall similarity metric. In this framework, fusion happens at the pixel level, which is represented by the architecture in figure 2(a).

5.2 Model combination fusion methods

To perform fusion at the model level, illustrated in figure 2(b), we use a *combination module*(CM) to fuse the similarity score from one or more appearance models. The following sections describe the various combination strategies we evaluated.

5.2.1 Simple and weighted averaging

$$S_{average}(X) = \sum_{i=1}^{M} w_i S_i(X) \tag{12}$$

where $S_i(X)$ is the similarity score between the i^{th} model and image patch X, and w_i is the weight assigned to the i^{th} model (See subsection 5.2.4 for details on how the weights are chosen). For simple averaging, the weights are fixed, with $w_i = \frac{1}{M}$ for $i = 1..M$, where M is the number of models being combined.

5.2.2 Similarity score product

$$S_{product}(X) = \prod_{i=1}^{M} S_i(X) \tag{13}$$

If the similarity scores returned by each model can be thought of as probabilities, this combined similarity returns the probability of matching the image patch X, using an assumption of independence.

5.2.3 Min and max score fusion

$$S_{min}(X) = \min_i S_i(X) \tag{14}$$

Using the minimum operator, an image patch with the greatest similarity will be such that no model believes this to be a bad match. Thus, we chose the new model position as the one that gives the 'least bad' match.

$$S_{max}(X) = \max_i S_i(X) \tag{15}$$

$$S_{wmax}(X) = \max_i w(i) S_i(X) \tag{16}$$

The maximum operator emphasises the score of a model that is very confident of finding a good match. We also use a weighted version of the maximum operator. A weighted *min* fusion scheme would not make sense, as it would use the most unreliable model for matching.

5.2.4 Dynamic Weighting

The weights shown first in equation (12) and also used in equation (16), are initialised with $w_i = \frac{1}{M}$ for $i = 1..M$. However, we also dynamically adapt the model weights based on how well the model discriminates within the search space. If a model returns many good matches, it is less specific on the best object position and its weight will be reduced. A model that returns one good match, corresponding to a sharp peak in the search space, will receive an increased weighting. More precisely, for each model, we take the maximum score returned over all search positions, and we divide it by the sum of scores returned by that model for all evaluated positions. This value, s_i for model i, is used to measure the model's specificity. We first compute normalised values, $\hat{s_i} = s_i / \sum s_p$ and then update the weights as follows:

$$w_i = \alpha w_i + (1 - \alpha)\hat{s_i} \tag{17}$$

where α is the update rate from equation (11).

5.3 Search strategies

Irrespective of the appearance model used, we adopt the same search strategy to locate the best match for the model in each frame, in all our experiments. We denote the object motion by an affine transformation [15] $A = \{a_1, a_2, a_3, a_4, t_x, t_y\}$ where $\{t_x, t_y\}$ are the translation parameters and $\{a_1, a_2, a_3, a_4\}$ are the deformation parameters. For our experiments, the objects we tracked were assumed to have negligible rotation, so we set two of the deformation parameters to zero, to only account for scale changes. In [14], particle filtering was used to search the transformation space to find the object in the current image. We adopt a deterministic approach, using coarse-to-fine gradient-ascent search of the transformation space, first adapting the translation parameters to maximise the similarity score, then adapting the scale parameters. This allows a more precise sub-pixel object detection than using randomly generated particles, which assists the tracker to remain locked onto targets when their appearance is changing.

6 Results

In this section, we evaluate the tracking performance of the fusion schemes that we described in the previous section, on six representative surveillance sequences. Details of the objects in the tracking sequences are given in table 1, whilst the notation for the trackers we used is described in table 2. The objects (along with their trajectories) are shown in figure 3.

In all our experiments, we used the background modelling algorithm described in [19], which is an improved version of the popular mixture of Gaussians model proposed by Stauffer and Grimson [20], providing a shadow removal mechanism. The infrared background is modelled separately using the same algorithm, but treating the pixels as greyscale. We use bi-linear feature interpolation for sub-pixel matching, updating and tracking. Four features ($k = 4$) are used in our tracking experiments: visible brightness value, infrared brightness value, visible edge magnitude and infrared edge magnitude. Edges were computed using the Sobel operator [21]. All trackers, except for GFM_V and GFM_I use all four features. In our general fusion model tracker, GFM_4, only one importance value is used per pixel. This requires that the binary foreground maps from each modality be fused into one map, which is used in the model update stage. We use the method described in our previous work [22], using a pixel's spatial support when there is modality disagreement.

To accurately judge the performance of the evaluated trackers, we manually annotated all of our video test sequences, by marking a bounding box, in each frame, around the object we wished to track. Using this bounding box, R_1, and the tracker's returned position, R_2, we compute the tracker's precision and recall:

$$\text{precision} = \frac{O(R_1, R_2)}{|R_2|} \tag{18}$$

$$\text{recall} = \frac{O(R_1, R_2)}{|R_1|} \tag{19}$$

where $O(a, b)$ is the overlapping area of rectangles a and b, and $|R|$ denotes the area of the rectangle R. We determine that a tracker has failed if its precision or recall drops below 0.1 for at least three consecutive frames. The performance of each tracker on all of the sequences, using both similarity metrics is detailed in table 3. The figures denote the percentage of frames in which the object was successfully tracked, for each tracker for 2 different similarity metrics. The empty entries in the table indicate that trackers CM_{avg} and CM_{wavg} were not evaluated with similarity metric, S_1. This is because the similarity values computed with S_1 are beyond the floating-point precision of standard processors (since it is derived from the product of many probabilities). Other computations, such as multiplication for CM_{prod}, can be achieved by adding logarithm values. The cases where a tracker 'fails' after a large number of frames (e.g. 97%) are due to either the object becoming very small (only a few pixels in size) or because the object enters an area where the visible and infrared images do not fully overlap (hence the infrared values are unknown).

Table 1: Tracked Objects

Seq	Object	Frames	Description
1	Face	113	Scale increase as person moves towards camera
2	Person	224	Dark night-time sequence
3	Bicycle	440	Large changes in scale and shape as cyclist approaches and turns
4	Motorbike	256	Large scale decrease as motorbike drives away
5	Person	666	Slight occlusion by background and shape change
6	Bicycle	314	Severe occlusion by another moving person and shape change

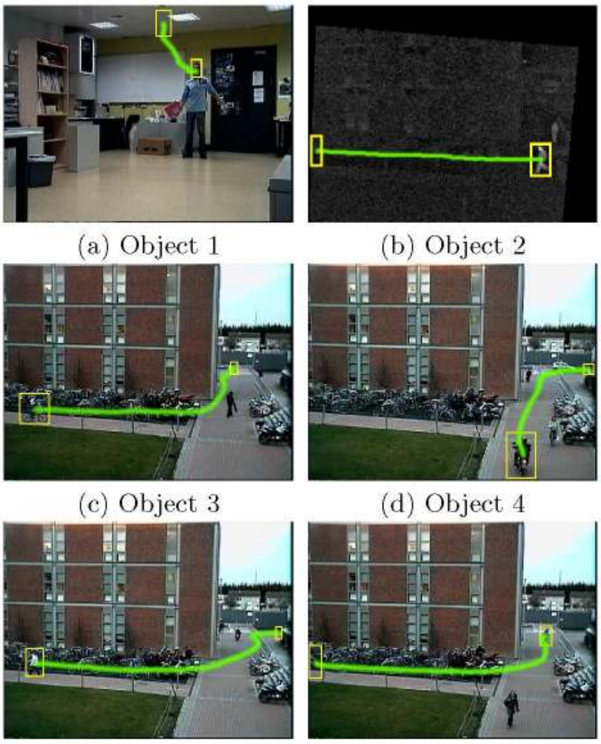
There are a number of conclusions that can be drawn from the figures in table 3. The best overall tracker was CM_{prod} which performed perfectly on all six sequences with either similarity metric. As expected, the tracker using visible spectrum only, GFM_V, failed on the night-time sequence (Seq-2). None of the sequences were particularly challenging for infrared, as the objects all had a strong signal compared to the background, therefore GFM_I fares quite well. We found that the adaptive weighting of models does not seem to add additional tracking performance (note the near identical performance of trackers CM_{avg} and CM_{wavg}, and trackers CM_{max} and CM_{wmax}). We speculate that this is because our models already account for noisy tracking by adapting the variance of the pixel models, giving lower scores for larger variances. Another explanation may be that the weightings were not strong enough, or were adapted too slowly. However, the weights generated by the proposed algorithm may have other uses, such as determining areas where a sensor is weak. It could also have uses in classification: for example, an object that has a weak infrared weight is probably of the same temperature as the background and therefore unlikely to be a person. Additionally, the method could prove useful in situations where one modality fails completely, such as when an infrared camera performs rapid automatic gain correction. Investigating this possibility remains as future work. From the table, we also see that tracker GFM_4 failed in a number of cases. This is likely due to the fact that there is a single importance weighting per pixel, rather than a per-pixel-per-modality importance. Therefore, incorrectly labelled foreground pixels caused by the *halo-effect* in infrared could label visible background as 'important'. Figure 4 shows an exam-

Table 3: Tracking Results

	Seq-1		Seq-2		Seq-3		Seq-4		Seq-5		Seq-6	
Tracker	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2
GFM_4	64	100	100	69	100	100	84	92	100	94	60	100
GFM_V	100	100	8	9	100	100	100	100	96	100	45	100
GFM_I	100	100	100	100	100	42	91	76	95	34	100	37
CM_{avg}	100	-	24	-	100	-	97	-	94	-	100	-
CM_{wavg}	100	-	24	-	100	-	97	-	100	-	100	-
CM_{prod}	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
CM_{min}	100	100	100	100	41	100	100	100	3	98	100	59
CM_{max}	100	100	8	9	100	42	91	76	95	34	100	37
CM_{wmax}	100	100	8	9	100	42	91	76	95	34	100	37

Table 2: Tracker Descriptions

Name	Description
GFM_4	General fusion model using all four features
GFM_V	General fusion model using visible brightness only
GFM_I	General fusion model using infrared brightness only
CM_{avg}	Model combination using score average
CM_{wavg}	Model combination using weighted score average
CM_{prod}	Model combination using score product
CM_{min}	Model combination using minimum score
CM_{max}	Model combination using maximum score
CM_{wmax}	Model combination using weighted maximum score


(a) Object 1

(b) Object 2

(c) Object 3

(d) Object 4

(e) Object 5

(f) Object 6

Figure 3: Tracked Objects listed in table 1

ple of the GFM_4 appearance model of the person in Seq-5, showing large variance in the leg pixels, due to the walking motion. It is difficult to evaluate the merits of the two similarity metrics we used, as neither shows a remarkable advantage over the other. However, the metric we proposed is more computationally efficient, since it does not requires the calculation of logarithm values. In all sequences, except Seq-6 (using S_1), the CM_{max} and CM_{wmax} trackers seem to perform only as well as the weakest of either GFM_V or GFM_I. This suggests that maximum score selection is a poor scheme for fusing data from appearance models for tracking. The CM_{min} tracker fares quite well overall, but fails immediately at the start of Seq-5, when it locks onto the background due to the scores of the visible edge features. Both CM_{max} and CM_{min} choose one score, and exclude all others. Experimental results validate this as an unwise approach.

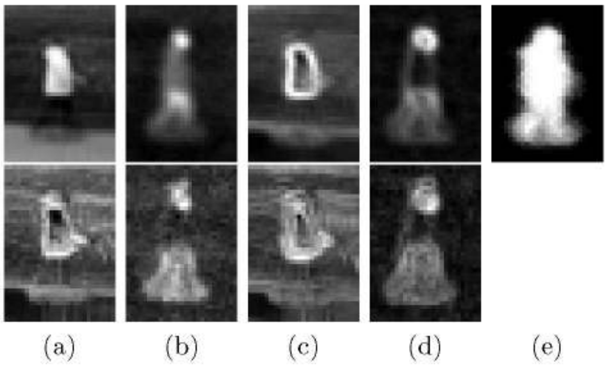


Figure 4: General fusion model of walking person from Seq-5: (a)-(d) Mean and variance of the four features we use for tracking, (e) Pixel importance weighting

7 Conclusions and future work

In this paper, we evaluated the appearance model tracking performance of multiple different fusion schemes, using manually annotated multi-modal surveillance video. The appearance model we used was based on an existing tracker but with notable improvements, including the ability to track non-rectangular patches, integrate information from background modelling algorithms and perform rapid initialisation. Additionally, we introduced an alternative similarity metric to match the appearance model to image patches.

There remain many avenues of future investigation. These include discovering the exact nature of the failures in individual cases, as well as performing more comprehensive testing using different video sequences, especially using data that is challenging for infrared alone. A theoretical analysis of the *product fusion* would be very useful, as it might further justify the tracking performance it achieved on our surveillance video.

One popular framework for data fusion that was not considered in the paper is the Transferable Belief Model [23]. It provides mechanisms to cater for doubt and uncertainty, which could be useful when there are multiple prominent peaks in the matching search space, or when one modality is missing (where the images from both modalities do not overlap). Another similarity metric to evaluate in future work is to compute the median pixel matching probability. This would be more robust to outliers and shape changes, but with a greater computational burden. Another area of future work will examine whether it would be beneficial to switch between similarity metrics, as our results seem to indicate that they can be complementary.

Acknowledgments

This material is based on works supported by Science Foundation Ireland under Grant No. 03/IN.3/I361 and sponsored by a scholarship from the Irish Research Council for Science, Engineering and Technology (IRCSET): Funded by the National Development Plan. The authors would also like to express their gratitude to Mitsubishi Electric Research Labs (MERL) for their contribution to this work.

References

[1] R. Cucchiara. Multimedia surveillance systems. In *VSSN 05: Proceedings of the third ACM international workshop on Video surveillance & sensor networks, New York, NY, USA*, pages 3–10, 2005.

[2] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man and Cybernetics*, 34(3):334–350, August 2004.

[3] G. Loy, L. Fletcher, N. Apostoloff, and A. Zelinsky. An adaptive fusion architecture for target tracking. In *IEEE International Conference on Automatic Face and Gesture Recognition (FGR)*, 2002.

[4] P. Pérez, J. Vermaak, and A. Blake. Data fusion for visual tracking with particles. *Proceedings of the IEEE*, 92(3):495–513, March 2004.

[5] K. Smith, D. Gatica-Perez, and J.-M. Odobez. Using particles to track varying numbers of interacting people. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2005.

[6] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2005.

[7] J. Lim and D. Kriegman. Tracking humans using prior and learned representations of shape and appearance. In *IEEE International Conference on Automatic Face and Gesture Recognition (FGR)*, pages 869–874, May 2004.

[8] K. She, G. Bebis, H. Gu, and R. Miller. Vehicle tracking using on-line fusion of color and shape features. In *IEEE International Conference on Intelligent Transportation Systems*, October 2004.

[9] G. Fumera and F. Roli. A theoretical and experimental analysis of linear combiners for multiple classifier systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6):942–956, June 2005.

[10] S. Avidan. Ensemble tracking. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2005.

[11] H. Kruppa and B. Schiele. Hierarchical combination of object models using mutual information. In *BMVC*, 2001.

[12] H. Torresan, B. Turgeon, C. Ibarra, P. Hébert, and X. Maldague. Advanced surveillance system: Combining video and thermal imagery for pedestrian detection. In *Proc. of SPIE, Thermosense XXVI*, volume 5405 of *SPIE*, pages 506–515, April 2004.

[13] A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti, and R. Bolle. Appearance models for occlusion handling. In *2nd IEEE Int. Workshop on PETS, Kauai, Hawaii, USA*, Dec 2001.

[14] S. Zhou, R. Chellappa, and B. Moghaddam. Appearance tracking using adaptive models in a particle filter. In *Proc. of 6th Asian Conference on Computer Vision (ACCV)*, Jan 2004.

[15] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition, 2003.

[16] J.P.W. Pluim, J.B.A. Maintz, and M.A. Viergever. Mutual-information-based registration of medical images: a survey. *IEEE Transactions on Medical Imaging*, 22(8):986–1004, Aug 2003.

[17] P. A. Viola. *Alignment by Maximization of Mutual Information*. Phd thesis, Massachusetts Institute of Technology, Massachusetts (MA), USA, June 1995.

[18] M. Irani and P. Anandan. Robust multi-sensor image alignment. In *International Conference on Computer Vision*, pages 959–966, 1998.

[19] P. KaewTraKulPong and R. Bowden. An improved adaptive background mixture model for real-time tracking with shadow detection. In *2nd European Workshop on Advanced Video-based Surveillance Systems, Kingston upon Thames*, 2001.

[20] C. Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings of CVPR99*, pages II:246–252, 1999.

[21] W. E. Snyder and H. Qi. *Machine Vision*. Cambridge University Press, Jan 2004.

[22] C. Ó Conaire, N. O'Connor, E. Cooke, and A. Smeaton. Detection thresholding using mutual information. In *VISAPP: International Conference on Computer Vision Theory and Applications, Setúbal, Portugal (to be published)*, Feb 2006.

[23] P. Smets. The combination of evidence in the transferable belief model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):447–458, 1990.