Feedback scheme for thermal-visible video registration, sensor fusion, and people tracking

Atousa Torabi, Guillaume Massé, and Guillaume-Alexandre Bilodeau
École Polytechnique de Montréal
Department of Computer Engineering
Montréal, QC, Canada
Email:{atousa.torabi, guillaume.masse, guillaume-alexandre.bilodeau}@polymtl.ca

Abstract

In this work, we propose a feedback scheme for simultaneous thermal-visible video registration, sensor fusion, and tracking for online video surveillance applications. The video registration is based on a RANSAC trajectory-to-trajectory matching that estimates an affine transformation matrix that maximizes the corresponding trajectory points and overlapping of foreground thermal and visible pixels. Sensor fusion uses the aligned images to compute sum-rule blobs for thermal and visible images and constructs the thermal-visible blobs. Finally, the multiple object tracking gets blobs constructed in sensor fusion as the input and outputs the trajectories of moving humans in the scene. We tested our method on long-term indoor and outdoor video sequences and demonstrate the effectiveness of our feedback design in obtaining better quality for both image registration and tracking.

1. Introduction

Nowadays, there is increasing interests in thermal-visible video surveillance systems in computer vision community [13, 4]. This domain of research has urged studying sensor fusion algorithms for methods related to thermal-visible surveillance videos applications such as background subtraction [3] and object tracking [9]. The main difficulty associated to bi-modal (i.e. thermal-visible) image/video registration is detecting and matching common features in images captured by two different sensors that reflect different information of the scene. In literatures, several works have been accumulated for multi-modal image registration. Krotosky and Trivedi [8] give comparative analyses of multimodal image registration methods in literatures. Irani and Anandan proposed an image registration by computing local correlation values of the features extracted from Gaussian pyramid of visible and thermal images and performing a global alignment using an iterative Newton's method [7]. In Coiras et al. paper, image registration is done by estimating an affine transformation that maximizes the global edge-formed triangle matching [2]. In Han and Bhanu work, a hierarchical genetic algorithm-based method is applied for matching the human silhouette in thermal and visible images using two pairs of corresponding points of a human walking in a straight line at a fixed distance from the camera [5]. However, in these works [7, 2, 5], the registration problem is defined as a low-level image-to-image feature-based matching problem. Since in video surveillance applications, our data is sequences of images, it is advantageous to use the spatio-temporal information of the scene and perform a sequence-to-sequence matching rather than the low level image-to-image matching. In fact, the spatio-temporal information of the scene, such as moving object trajectory points, allows doing registration in the situation where the image-to-image feature matching is considerably challenging such as multimodal image registration. In Caspi et al. paper, a feature-based sequence-to-sequence matching technique is proposed based on matching object trajectory points [1]. However, trajectory-based matching involves another problem which is computing the object trajectories of moving objects in the scene for a pair of video sequences. In our previous work [10], we proposed a trajectory-based sequence-to-sequence video registration, where the object trajectories were computed separately offline for thermal and color video sequences using a multiple object tracking. In this paper, we propose an online simultaneous thermal-visible video registration, sensor fusion and tracking for two synchronized streams of long-range videos recorded by collocated visible and thermal cameras under different zooms. We mainly focus on the proposed feedback scheme and the collaboration of three modules of our system (image registration, sensor fusion, and tracking) to demonstrate the effectiveness of the whole system.

Contribution. We have improved the trajectory-based

978-1-4244-7028-0/10/$26.00 ©2010 IEEE

matching method that Caspi proposed [1], by applying a new criterion for matching, which is the overlap foreground pixels of two images. This registration criterion improves performance of matching when few trajectories exist in the scene. Also the iterative feedbacks between the modules of our system considerably augment the performance of whole system in two ways. 1) thermal-visible sensor fusion improves input data of tracking in thermal and visible videos, which results in more accurate object trajectories. The sensor fusion, based on a feedback about the quality of image registration, computes the blobs in a way to reduce errors caused by imperfect image registration in some frames that may also cause inaccurate trajectory computations. 2) More accurate trajectories computed by tracking results in more accurate image registration, which is based on matching the trajectory points.

2. Overview of methods

The input data of the system is the synchronized thermal and visible video streams acquired by a thermal and a visible camera that are collocated with intersecting field of view (FOV). As pre-processing, we applied the background subtraction in [11] to separate foreground pixels in each image. In this part, a fair amount of false negative and false positive foreground pixels is acceptable. Therefore, any other reasonable background subtraction method may be used. Fig. 2 shows the flowchart of our algorithm that contains two stages: 1) Initialization and 2) the main loop for image registration, blob fusion, and tracking. Initialization is done at beginning of the videos, where for some frames, tracking is performed separately for the thermal and the visible video frames without any data fusion, until we obtain enough object trajectory points in the scene for estimating a good transformation matrix needed for thermal-visible sensor fusion. The second part of the algorithm is a loop on pairs of thermal and visible video frames, where image registration, sensor fusion, and tracking are performed. Image registration estimates an affine transformation matrix that is used to transform one image to the coordinates of the second one. Sensor fusion matches color and thermal pixels of blobs using the transformation matrix computed in image registration and combines thermal and color information. This considerably improves the quality of input data for tracking. Also, the matching quality of blobs is evaluated to decide whether if estimating new transformation matrix should be done or skipped in the next frame. At last, tracking is performed for thermal and visible videos using information obtained from sensor fusion. At each frame t, tracking algorithm computes the trajectories of objects in thermal and visible videos up to the current frame for image registration computation at next frame $t+1$. Our proposed algorithm is described thoroughly in next sections.

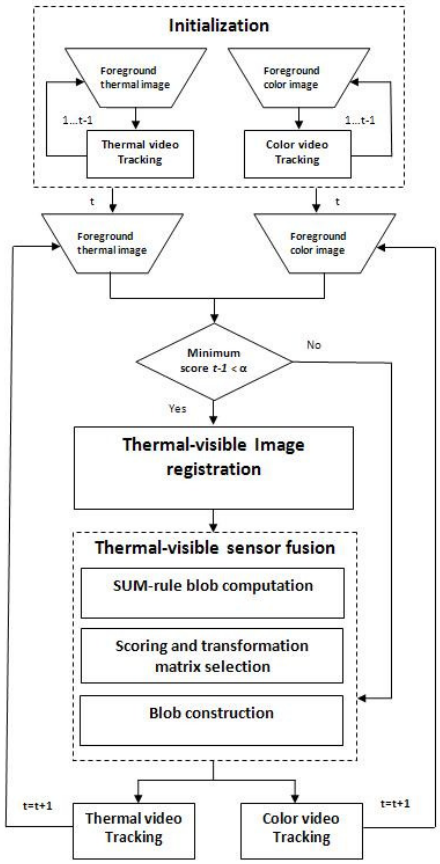


Figure 1. Flowchart of our system

3. Initialization

At the beginning of the videos, a few trajectory points are needed to compute a first good estimate of the transformation matrix used for sensor fusion. For a fixed number of frames, tracking is performed separately in thermal and visible videos. Then, image registration is done and overlapping error (equation 3) is computed . Registration is repeated until reaching a frame for which the overlapping error is less than a fixed threshold to ensure the good quality image alignment necessary for sensor fusion.

4. Image registration

In our camera setup, we consider thermal and visible cameras in different zoom. Cameras might have relative translation along x-axis and y-axis and rotation around the z-axis. In our work, it is assumed that objects lie approximately on the same plane in the scene. This assumption is valid since videos are recorded for far-range scenes. Therefore, for image coordinate transformation, we applied an affine homography matrix H that is relatively fast to calculate and is sufficiently accurate for video surveillance applications. H is calculated using matching trajectory points

Repeat N_1 times

1. Pick a pair of trajectories at random
2. Estimate the preliminary affine matrix H:
   Repeat N_2 times
   - Pick three pairs of points at random
   - Calculate H
   - Add inliers pairs of points:
     a. Evaluate Euclidean distance error
     b. Re-calculate H
3. Evaluate overlapping error
4. Add inliers pairs of trajectories:
   Repeat for all possible pairs
   - Add inliers pairs of points
   - Evaluate overlapping error

Figure 2. RANSAC-based algorithm for trajectory point matching

selected by a RANSAC-based algorithm from the pair of images of the left and the right view (thermal and color images). In fact, each object trajectory point is the position of the top-most/centroid point of the blobs' silhouettes in the corresponding video frame. In each frame *t*, the trajectories of the objects in the scene are updated by tracking for both visible and thermal videos and are used for image registration in the following frame *t+1*.

4.1. Thermal-visible image registration algorithm

Our RANSAC algorithm is a non-deterministic iterative algorithm that estimates the transformation matrix based on matching the object trajectory points from a pair of thermal and visible images. Figure 2 shows the steps of our object trajectory points matching. It is composed of two RANSAC loops, one for the pairs of trajectories with N_1 iterations, and one for the pairs of points in a selected pair of trajectories with N_2 iterations. A pair of trajectories is composed of one trajectory from the thermal video and the other one from the visible video. For example, at frame t, there might be three trajectories for thermal video (T^1_{left}, T^2_{left} and T^3_{left}) and two trajectories for visible video (T^1_{right} and T^2_{right}), then we get six pairs of trajectories that are used as the data pool for the RANSAC algorithm. Since the videos are synchronized, possible corresponding points are points that have the same timestamp. In fact, matching possible pair of points instead of all the points reduces considerably the combinatorial complexity of matching problem.

4.1.1 Number of iterations in a RANSAC loop

The RANSAC algorithm is an iterative algorithm. The number of iterations N is computed as follows,

$$N = \frac{log(1-p)}{log(1-(1-\epsilon)^s)}, \tag{1}$$

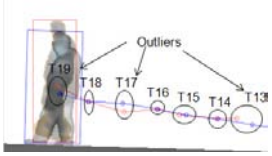


Figure 3. Matching object trajectory points from thermal and visible video. $T14$, $T15$, $T16$, $T18$, $T19$ are inliers

where, p is the confidence (in our experiments p is 0.99), and ϵ, the probability of outliers, is computed by

$$\epsilon = 1 - \frac{N_p}{N_t}, \tag{2}$$

where N_p is the number of inliers pairs of points/trajectories, and N_t is the total number of pairs of points/trajectories. In fact, the number of iterations depends to the number of inliers pairs of points/trajectories. The bigger the number of the inliers pairs, the less iterations are needed.

4.1.2 Computing the first estimate of matrix *H*

To compute H, three pairs of points are selected at random. Then, H is calculated for the three selected pairs of points. After that, all the points of the trajectory of the thermal video frame are transformed using the estimated H and the Euclidean distance between theses transformed points and their corresponding points in the visible video are computed. Pairs of points for which the Euclidean distance is smaller than a threshold T (typically, $T = 5$ pixels) are considered as inliers pairs. The best estimation of H is the one computed with the largest number of inliers pairs of points. H is re-estimated using all the inliers pair of points. Figure 3 illustrates the matching of selected pairs of trajectory points.

4.1.3 Computing foreground overlapping error

After first estimation of transformation matrix H, its quality is evaluated using an overlapping error function, OE, defined for the foreground pixels of the pair of thermal and visible video frames.

$$OE = 1 - \frac{N_{c \cap t}}{N_{c \cup t}}, \tag{3}$$

where $N_{c \cap t}$ is the number of overlapping foreground pixels of color and thermal images and $N_{c \cup t}$ is the number of foreground pixels from the union of color and thermal images. Evaluating overlapping errors allows our method to perform even if there are few trajectories in the scene.

Repeat for thermal and visible images

1. Compute sum-rule silhouettes using matrix M_n
2. Compute sum-rule silhouettes using matrix M_b
3. Compute Blobs' scores:
   Repeat for all the blobs in the reference image
   - Calculate *Score* of silhouette computed using M_n
   - Calculate *Score* of silhouette computed using M_b
4. Compute overall score $Score_n$ of reference image
5. Compute overall score $Score_b$ of reference image
6. select transformation matrix:
   IF $Score_n > Score_b$ **THEN** $M_b = M_n$
7. Compute object model

Figure 4. Our sensor fusion algorithm

4.1.4 Adding inliers pairs of trajectories

For each possible pair of trajectories, the thermal image trajectory points are transformed to visible image coordinates. For the pair of trajectories, the inliers pairs of points are selected using Euclidean distance error described in section 4.1.2. Using all inliers points, the H matrix is re-calculated. Then, the overlapping error is computed for new estimated matrix H. If the overlapping error for new estimated matrix is less than the overlapping error of the previous estimation of H, the pair of trajectories are added to inliers pair of trajectories set. This procedure is continued until all the possible pairs of trajectories are evaluated.

5. Thermal-visible sensor fusion

Thermal-visible sensor fusion combines the information of registered color and thermal foreground images. Figure 4 shows our sensor fusion algorithm. M_n represents the transformation matrix estimated by the image registration in current frame and M_b represents the current best matrix (more details in section 5.2). If image registration is not performed in the current frame, step 1 is just skipped. The following sections explain the steps of sensor fusion algorithm in details.

5.1. Sum-rule silhouette computation

To compute the sum-rule silhouette, either foreground pixel coordinates of thermal image should be transformed to visible image coordinates or vice versa. In both ways, the computed sum-rule silhouette is the same. Sum-rule method is proposed in [6], and is defined as,

(X,Y)$\in$ S: **IF** $P(S \mid t(X,Y)) + P(S \mid c(X,Y)) > \alpha_{sum}$,

where $t(X,Y)$ represents thermal image coordinates, $c(X,Y)$ represents color image coordinates after transformation, S represents sum-rule silhouette, and α_{sum} represents a threshold, and the probabilities are computed as,

$$P(S|t(X,Y)) = 1 - e^{\|t(X,Y) - \mu_t(X,Y)\|^2} \quad (4)$$

$$P(S|c(X,Y)) = 1 - e^{\|c(X,Y) - \mu_c(X,Y)\|^2} \quad (5)$$

where $\mu_t(X,Y)$ and $\mu_c(X,Y)$ are respectively the mean background value of the coordinate (X,Y) for thermal and transformed color images.

5.2. Sum-rule silhouette accuracy estimation and matrix selection

The accuracy of computed silhouette varies based on the accuracy of applied transformation matrix. The quality of a sum-rule silhouette is evaluated using a score function. A transformation matrix is selected, based on the scoring results of all the silhouettes inside one image. The score function for thermal and color image are defined as follows,

$$SF_t(i) = \frac{sum\left(B^t_{j\in\{1,...n\}} \cap S^t_i\right)}{sum\left(B^t_{j\in\{1,...n\}}\right)}, i \in \{1,...,m\} \quad (6)$$

and

$$SF_c(i) = \frac{sum\left(B^c_{j\in\{1,...n\}} \cap S^c_i\right)}{sum\left(B^c_{j\in\{1,...n\}}\right)}, , i \in \{1,...,m\} \quad (7)$$

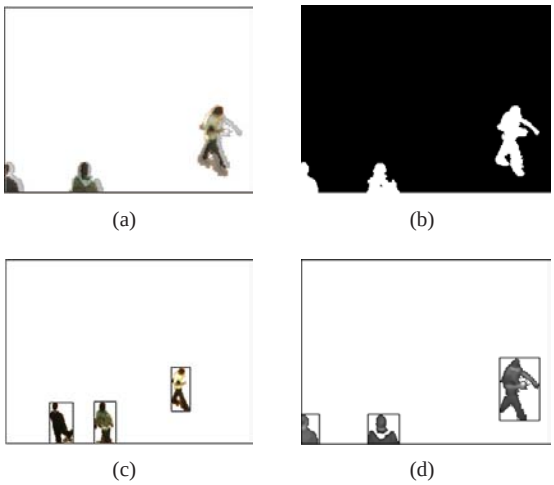
where m is the number of computed sum-rule silhouettes inside the intersecting field of view of the two cameras, S^t_i represents i^{th} sum-rule silhouette computed in the thermal image, $SF_t(i)$ represents its score, and B^t are blobs of the original thermal foreground image that intersects S^t_i. Since background subtraction is not perfect, in the original foreground image, object regions might be fragmented to smaller blobs. So blobs B^t that intersect S^t_i should be all fragments belonging to one object. If all blobs B^t are inside S^t_i, then S^t_i is perfectly aligned and its score will be 1 (the maximum value). The same applies for visible using $SF_c(i)$ function. The score of matrix M_n and M_b for one image are respectively,

$$Score_n = \left\{ \frac{\sum_{i=1}^{m} (SF_c(i) + SF_t(i))}{2 \times m} \right\}_{M_n}, \quad (8)$$

and,

$$Score_b = \left\{ \frac{\sum_{i=1}^{m} (SF_c(i) + SF_t(i))}{2 \times m} \right\}_{M_b}, \quad (9)$$

where m is the number of sum-rule silhouettes, $Score_n$ is the score of matrix M_n, since the silhouettes are aligned



(a) (b) (c) (d)

Figure 5. (a) Aligned visible on thermal image, (b) Computed Sum-rule blobs on thermal image, (c) Detected blobs in visible image, and (d) Detected blobs in thermal image

using M_n, and $Score_b$ is the score of matrix M_b the best tranformation matrix. For an image, if the score $Score_n$ of new estimated matrix is higher than the score $Score_b$ of the best matrix, M_n replaces M_b.

5.3. Blob construction

Blobs are the input data of tracking. Sensor fusion improves the quality of the input data of tracking by computing sum-rule silhouette that handles the background subtraction shortcomings, using a single sensor, such as blob fragmentation. Furthermore, sensor fusion provides the color and thermal information of the blob pixels that are used as features for tracking. For blob construction, if the score of a sum-rule silhouette (equation 6 or 7) is more than a fixed threshold, sum-rule silhouette will be considered as a detected blob in the reference image. Otherwise, if the sum-rule silhouette score is not sufficiently high, the original blob's fragments computed by background subtraction in reference image that have intersection with computed sum-rule silhouette will be clustered as one blob. In this way, the fragmentation problem is handled. Figure 5 shows the image alignment is not perfect, therefore the detected blobs in thermal image are clustered blob fragments of thermal image.

6. object model and Tracking

The object model used in our tracking is the color-thermal histogram of input blobs. This histogram has 54 bins for HSV color of the pixels and 16 bins for thermal intensities of the pixels. For tracking, we used the multiple hypothesis tracking method in [12]. In this method, the blobs in consecutive frames are matched based on their overlapping bounding box and in situation in which there is an ambiguity of blob's identity such as merging/splitting, a graph-based multiple hypothesis method is applied to label blobs. This method is suitable for online applications, since it labels blobs at every frame and computes the trajectories up to the current frame. This method [12] includes fragmentation handling, but in our method, sensor fusion using thermal and visible video sequences replaces it.

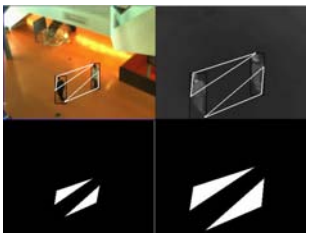


Figure 6. Top: manually selected polygons in IR and visible images (Frame 90, Seq.1), Bottom: ground-truth images

7. Results and discussion

For validating our method, we used one video from OTCBVS (dataset 03, seq. 5)[3] and three homemade videos (LITIV dataset). Figure 8 shows qualitative results of image registration, sensor fusion, and tracking. Our system performs in the intersection of field of views of thermal and visible cameras. It is shown in second row of figure 8 columns (f) and (g) that if one object does not exist in FOV of both cameras, it is not tracked since the sensor fusion needs the data coming from both sensors.

7.1. Image registration

We assessed the performance of the image registration part of our system, named with sensor fusion (WSF), by comparing it with modified version of this image registration (the same method) that is not a part of a whole system and performs independently using the same parameters. In the modified version, an affine matrix is estimated iteratively at each frame using trajectories generated from a thermal video tracker and a visible video tracker, without sensor fusion (WOSF). For quantitative comparison of the two image registration approaches, we constructed two ground-truth (GT) binary foreground images using manually picked corresponding points forming polygons in thermal and visible images. Figure 6 shows manually selected polygons and GT thermal and visible binary images. To validate our method, the overlapping error (equation 3) is computed for GT images, applying the transformation matrices computed by both image registration methods. Figure 7 shows the overlapping error (equation 3) of WSF and WOSF for video seq.1 from LITIV dataset (our homemade video). This plot shows that WSF estimates a good transfor-

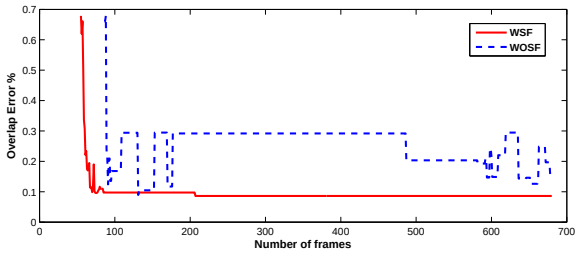


Figure 7. WSF: overlapping error of our image registration and WOSF: overlapping error of modified version of our image registration.

mation matrix earlier than WOSF. It also shows the transformation matrix accuracy of WSF is more stable in time compared to WOSF. In this comparison, the better performance of our image registration (WSF) demonstrates the effectiveness of our system with feedback, and it is because of two reasons: 1) the trajectory points computed by tracking with data fusion are more accurate compared to trajectories computed by separate tracking in thermal and visible videos. Therefore, the transformation matrix estimated using more accurate trajectory points is more precise. 2) Our matrix selection based on the fusion scores (section 5.2) performs better than the overlapping criterion used in image registration process, which results in a more stable and accurate matrix.

Also in table 1,we show the average Euclidean distance error of the corners of the polygons in thermal and visible GT foreground images that are aligned with estimated matrices using our system (rows (A)) and WOSF (rows (B)). As shown, in both X and Y direction, the Euclidean distance error of our system is less. In our test, videos from OTCBVS are considered as two unregistered sequence of images. Thus, we applied our image registration method. The Euclidean error related to these videos shows that our system succeeds in estimating a transformation matrix near to identity matrix.

7.2. Tracking

In table 2, we quantitatively compared tracking results of our system and a modified version of our system that always use sum-rule silhouettes as tracking input data (blob construction step is not performed). This means the blobs constructed in the sensor fusion always are the sum-rule silhouettes without considering if they are perfect or imperfect based on the quality of image registration. In this comparison, we assessed the effectiveness of our blob detection step that uses the feedback about the quality of image registration. In fact, imprecise blob as tracking input results in inaccurate object trajectory point. Consequently, it causes computing inaccurate transformation matrices in next frames. Therefore, our blob construction step has a critical role in

$Seq.$	NF	NM	SF	NP	AE_X	AE_Y
1(A)	680	626	54	7	0.68	2.17
1(B)	680	626	54	7	6.03	11.00
2(A)	698	541	157	3	4.14	3.37
2(B)	698	541	157	3	4.55	3.99
3(A)	1238	1038	200	5	3.84	5.74
3(B)	1238	1038	200	5	4.09	8.45
4(A)	2031	1851	180	2	1.29	1.57
4(B)	2031	1851	180	2	1.90	1.4

Table 1. Seq.1-3, videos from LITIV dataset and Seq. 4 OTCBVS videos[3]. (A) rows, our system results. (B) rows, WOSF results. NF: number of video frames, NM: number of matrices which is equal to number of video frames that transformation matrix is estimated, SF: Starting frame which varies based on fixed frame number criterion of initialization part (section 3), AE_X: Average Euclidean error of Polygons' corners of ground-truth thermal image and aligned visible image on thermal along X direction, AE_Y: Average Euclidean error of Polygons' corners of ground-truth thermal image and aligned visible image on thermal along Y direction.

$Seq.$	NF	NP	$-P_{ir-vi}$	$+P_{ir-vi}$	AE_{ir-vi}
1 (A)	680	7	0-0	0-0	3.57-2.12
1 (B)	680	7	0-0	3-1	5.05-2.89
2 (A)	698	3	0-0	0-1	2.32-3.57
2 (B)	698	3	0-0	3-0	2.54-4.46
3 (A)	1238	5	0-0	0-0	2.72-2.83
3 (B)	1238	5	0-0	0-2	3.47-3.93
4 (A)	2031	2	0-0	1-0	2.51-1.38
4 (B)	2031	2	0-0	5-0	5.28-4.91

Table 2. Seq.1-3, videos from LITIV dataset and Seq. 4 OTCBVS videos[3].(A) rows, our thermal-visible tracking results. (B) rows, thermal-visible tracking results without feedback between registration and sensor fusion modules. NF: frame number, NP: number of tracked people, $+FP_{ir-vi}$: falsely tracked number of people in thermal and visible sequences, $-FP_{ir-vi}$: falsely missed tracking number of people in thermal and visible sequences, and AE_{ir-vi}: Average trajectory Euclidean distance error compared with manually generated GT trajectories.

quality of the whole system. Table 2 shows the quantitative results of this comparison. It is shown that trajectory points' average Euclidean distance error AE_{ir-vi} of our system is smaller than AE_{ir-vi} of modified version of our system (without blob construction step). We get smaller errors in our system, because in some frames in which two images are not perfectly aligned, the tracking input blobs are clustered fragment blobs of the original image instead of sum-rule silhouettes. Since, in general, the trajectory points related to clustered fragment blobs are more precise than the trajectory points related to inaccurate sum-rule silhouette, the average Euclidean distance error of the trajectory points computed in our system is smaller. Table 2 also shows that

(a) (b) (c) (d) (e) (f) (g)

Figure 8. Our results of Seq.1 at frames 99, 182, 300, and 652. (a) Registration of visible on thermal image, (b) Sum-rule silhouette aligned on visible image (c) Sum-rule silhouette aligned on thermal image, (d) and (f) Tracking result of visible image, (e) and (g) Tracking result of thermal image

the number of falsely detected people $+P_{ir-vi}$ in our system is smaller. The falsely detected people error usually occurs in video because of blob fragmentation error caused by background subtraction. For example, in thermal image of Seq. 4, several blob fragmentation occurs using background subtraction. In our results, we have 1 falsely detected people and in our modified system, we have 5 falsely detected people. This is because in our system, most fragmentation is handled in sensor fusion resulting in more accurate image registration compared to the modified version of our system.

8. Conclusion and future works

In this paper, we proposed online simultaneous thermal and color video registration, sensor fusion, and object tracking with an iterative feedback design. In our result, we have shown that the sensor fusion improves the tracking, which results in accurate object trajectories. We have also shown that using the accurate object trajectories computed in tracking module in frame $t-1$ improves trajectory-based video registration of the next frame t. In our system, the imperfect results of background subtraction (blob miss detection and false detection) is improved by sensor fusion. Then, fusion data is used as input data for tracking. However, image registration still uses results of background subtraction that is performed separately in thermal and visible video without sensor fusion. Our system can be improved by aligning thermal and visible video frames at frame t using the transformation matrix computed in frame $t-1$. After, a thermal-visible background subtraction method such as the one proposed in [9] can be applied. In this way, the quality of the whole system will be improved.

9. Acknowledgments

We would like to thank the Canadian Foundation for Innovation (CFI), Natural Sciences and Engineering Research Council of Canada (NSERC), and the Fonds québécois de la recherche sur la nature et les technologies (FQRNT) for their support with grants and a scholarship respectively.

References

[1] Y. Caspi, D. Simakov, and M. Irani. Feature-based sequence-to-sequence matching. *Int. J. Comput. Vision*, 68:53–64, 2006. 1, 2

[2] E. Coiras, J. Santamaria, and C. Miravet. Segment-based registration technique for visual-infrared images. *Optical Engineering*, 39:282–289, 2000. 1

[3] J. W. Davis and V. Sharma. Fusion-based background-subtraction using contour saliency. In *CVPR '05: IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 11–19, 2005. 1, 5, 6

[4] R. Hammoud. *Augmented Vision Perception in Infrared: Algorithms and Applied Systems*. Springer Publishing Company, Incorporated, 2009. 1

[5] J. Han and B. Bhanu. Detecting moving humans using color and infrared video. In *International Conference on Multisensor Fusion*, 2003. 1

[6] J. Han and B. Bhanu. Fusion of color and infrared video for moving human detection. *Pattern Recogn.*, 40:1771–1784, 2007. 4

[7] M. Irani and P. Anandan. Robust multi-sensor image alignment. In *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, 1998. 1

[8] S. J. Krotosky and M. M. Trivedi. Mutual information based registration of multimodal stereo videos for person tracking. *Comput. Vis. Image Underst.*, 106(2-3):270–287, 2007. 1

[9] A. Leykin and R. Hammoud. Robust multi-pedestrian tracking in thermal-visible surveillance videos. In *CVPRW '06: Conference on Computer Vision and Pattern Recognition Workshop*, pages 136–144, 2006. 1, 7

[10] F. Morin, A. Torabi, and G.-A. Bilodeau. Automatic registration of color and infrared videos using trajectories obtained from a multiple object tracking algorithm. pages 311–318, 2008. 1

[11] B. Shoushtarian and H. E. Bez. A practical adaptive approach for dynamic background subtraction using an invariant colour model and object tracking. *Pattern Recogn. Lett.*, 26(1):5–26, 2005. 2

[12] A. Torabi and G.-A. Bilodeau. A multiple hypothesis tracking method with fragmentation handling. In *Computer and Robot Vision, 2009. CRV '09*, pages 8–15, 2009. 5

[13] Z. Zhu and T. Huang. Multimodal surveillance: an introduction. pages 1 –6, 2007. 1