Visible and infrared image registration using trajectories and composite foreground images

G.A. Bilodeau*[,a], A. Torabi[a], F. Morin[a]

[a]*LITIV, Department of Computer and Software Engineering,*
École Polytechnique de Montréal,
P.O. Box 6079, Station Centre-ville, Montréal
(Québec), Canada, H3C 3A7

Abstract

The registration of images from multiple types of sensors (particularly infrared sensors and visible color sensors) is a step toward achieving multi-sensor fusion. This paper proposes a registration method using a novel error function. Registration of infrared and visible color images is performed by using the trajectories of moving objects obtained using background subtraction and simple tracking. The trajectory points are matched using a RANSAC-based algorithm and a novel registration criterion, which is based on the overlap of foreground pixels in composite foreground images. This criterion allows performing registration when there are few trajectories and gives more stable results. Our method was tested and its performance quantified using nine scenarios. It outperforms a related method only based on trajectory points in cases where there are few moving objects.

Key words: Infrared, Registration, multi-sensors, trajectories, foreground

*Corresponding author
Email addresses: guillaume-alexandre.bilodeau@polymtl.ca (G.A. Bilodeau), atousa.torabi@polymtl.ca (A. Torabi), francois-2.morin@polymtl.ca (F. Morin)

images

1. Introduction

Traditionally, the computer vision community has focused principally on processing images captured with visible range sensors (400-1000 nm) during day and in indoor environments. Because of their cost, infrared sensors (.9-13.5 μm) were used only in special area like medicine (breast cancer detection) and military (night vision). Infrared sensors are now becoming valuable assets in an advance surveillance system as they supply information that a visible sensor cannot provide in poor lighting, smoke, and fog. It is often useful to pair an infrared camera with a visible camera. They both perform well in different situations. For example, a visible camera fails when it is dark, but is good during the day. An infrared camera fails during very hot summer days (objects are undistinguishable), but otherwise is efficient in the dark. The two cameras are valuable in a video surveillance setup.

To benefit from both sensors, the fusion of information is often used. In many fusion techniques, registration is needed to find automatically the transformation matrix between two images or between two videos. In this paper, we are interested in registering two videos acquired from two cameras (one visible and one infrared) installed in a stereoscopic configuration and observing a common scene with an overlapping field of view. We assume that objects are in the same plane (i.e. that the scene is planar), that is, objects' distances from the camera are much larger than the camera baseline. We are particularly interested in the case for which the two cameras produce useful data, for example, during the day. In this situation, two images are obtained, and thus the registration can be found to be used later on to improve monitoring of a scene in various in various lighting and weather conditions.

In this paper, we are dealing with the issue of evaluating a transformation matrix based on a registration using trajectories [1, 2, 3]. Previous works use Euclidean distance between trajectory points as an error function for evaluation, but it requires more trajectories to constrain the possible transformations. Furthermore, in our case, we do not wish to estimate the registration of a recorded video. We aim to register two cameras online as an automatic procedure that can be applied periodically over normal system operation. Thus, it is useful to find an evaluation criterion for finding the transformation matrix that does not require a large number of trajectories.

Our proposed registration method for infrared and visible color videos is based on the trajectories of moving objects obtained using background subtraction and simple tracking. The videos from the two cameras are assumed to be synchronized. The trajectory points are matched using a RANSAC-based algorithm [4]. The novel registration criterion used in this algorithm is the number of overlapping foreground pixels in composite foreground images. Note that we do not address the problem of fusion in this paper. We tested our method with the affine model, although it is applicable to any 2D homography. Results are compared to ground-truth, and we show that using composite foreground images improve registration accuracy compared to using only distance on trajectory points. Furthermore, it gives more stable results. Because our proposed method aligns silhouettes instead of trajectory points, it is appropriate as automatic registration for methods performing image fusion using blob contours or edges [5, 6], where the alignment of these structures is assumed and required.

Related works are discussed in section 2. Section 3 presents our methodology and section 4 presents registration results and an accuracy evaluation with ground-truth. Finally, section 5 concludes the paper.

2. Related works

Most works related to this paper are about visible/infrared image registration. For registering infrared and visible images, correspondences between images must be found. There are two types of approaches: 1) intensity-based methods, and 2) Feature-based methods [7]. These approaches have been developed for visible stereo pairs, but several researchers have tried applying them to mixed infrared/visible pairs. Their application to this camera configuration is not straight forward since infrared and visible images are the manifestation of two different phenomena. Visible cameras measure reflected light on objects, while infrared camera measure principally infrared radiations emitted by objects. A texture or an edge in a visible image is often missing in the infrared image because texture seldom influences the heat emitted by an object.

2.1. Approaches based on intensity

Approaches based on intensity rely on either cross-correlation, on Fourier method, or on the mutual information theory. Cross-correlation (CC) matching is computed for pairs of windows in two images. A pair of windows that maximizes the cross-correlation is considered a correspondence [8]. This method has some drawbacks which are the flatness of similarity measure in textureless areas and the high computational complexity.

The second matching approach is the Fourier method. The Fourier representation of edge images is exploited and correlations are found in the frequency domain [9]. It has robustness to noisy images and images that are captured under variable lighting conditions. It is relatively fast compared to correlation-like methods.

Finally, in mutual information theory methods [10, 11, 12, 13, 14], images from both sensors are transformed and overlapped in such a way to maximize the dependence between the two images. The mutual information is used as a quality metric to evaluate correspondence for a given transformation. However, for two given infrared/visible image pairs, the mutual information might be good only on a small portion of the images. Thus, mutual information can be used only on a selected region of an image [13], on region with similar edge density [11], or on a detected foreground [12]. Edges can be used to obtain a transformation estimate before applying mutual information [14].

2.2. Approaches based on features

Edges can be extracted using discrete wavelet transform [15] or Gabor filters [16]. A threshold is used to select some edges that are considered more discriminative, and then they are used to compute the transformation matrix by pairing them in different combination using a RANSAC [4] type method. The distance between edge points, or invariant moments and contour orientation, are used as the criterion to evaluate a transformation. Another alternative is to extract the edges with an edge detector, and then group the edges into segments [17]. Segments are further grouped into triangles. Affine transformation matrices are then computed to align pairs of triangles until a quality criterion is reached. Similarly, in other works [18, 19], edges are extracted using the Canny edge detector. The quality of the transformation is evaluated using the Hausdorff distance between edge points. Finally, edges and foreground detection may be used [20]. In this case, only the edges on moving objects are used for computing the registration.

Feature point-based methods are founded on the principle that only a few points with known correspondence are required to compute a registration. The challenge is to find points that appear both in infrared and visible images. Corners can be used in some situations, for example, to register facade [21]. In this case, registration is found by minimizing the Hausdorff distance between corresponding corners. Another approach is based on foreground detection and trajectory formed from centroids of the detected blobs [22, 2]. For a given blob, its center of mass (centroid) in consecutive images forms a trajectory in the video. This method matches points on trajectory pairs to compute the transformation matrix. Thus, it uses temporal information. Many combinations of trajectories are tested with a RANSAC algorithm to minimize the transformation error which is evaluated based on the Euclidean distance between corresponding trajectory points.

2.3. Approaches based on consistent temporal behavior

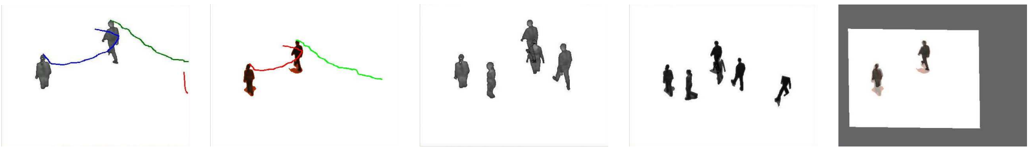
Alternatively, a third possible approach relies on consistent temporal behavior. In intensity and feature based approaches, the image alignment requires a consistent appearance in two images. However, for the alignment of two image sequences that do have not spatial overlap between their field of view, retrieving common feature is nearly impossible because the consistent appearance assumption is not valid. Caspi and Irani have proposed a method for alignment of both in time and space of two image sequences with no overlap between their field of view [23]. In their method [23], homography matrix (2D/3D) between two cameras and temporal synchronization are computed using the consistency of temporal behavior between two image sequences, which is satisfied by using two collocated cameras that are moved jointly in space. In our case, since we have overlapping fields of view, we can use image features and use a simpler method.

Our proposed method is based on the work of [1, 2, 3]. These works are based solely on the Euclidean distance between corresponding trajectory points to evaluate candidate transformation matrices. Since trajectories are essentially one-dimensional, the drawback of this registration criterion is that it needs many trajectories to restrict the possible transformations between the two videos. Few trajectories give inaccurate and unstable registration. In this work, we aim to improve this registration criterion to allow accurate registration even if only few trajectories are available.

3. Methodology

3.1. Overview of the registration method

Our proposed method is based on trajectories and regions of detected foreground objects. Figure 1 illustrates the process. The cameras are assumed to be in a stereoscopic configuration, synchronized, and stationary for the duration of the calibration procedure. We also assume that the objects are in the same plane, so we can compute the transformation matrix for all moving objects. The main steps of our method are:



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

Figure 1: Illustration of processing steps for a given frame. (a) and (b) Object trajectories in IR and visible respectively, (c) and (d) Foreground composite image in IR and visible respectively, (e) Registration result

1. Find the trajectories of moving objects using a simple tracking method (figure 1 (a) and (b));
2. Calculate the new composite foreground images (figure 1 (c) and (d));
3. Find the best point and trajectory correspondences using a RANSAC-based method, a transformation matrix, and foreground composite images;
4. Calculate the transformation matrix based on the best set of trajectory points;
5. Select the best transformation matrix (current versus the new one of step 4) (figure 1 (e)).

In the following, the video sequence of the infrared camera will be referred to as the left image or the left video. In the same way, the right image or the right video will correspond to the visible color camera.

3.2. Background subtraction and tracking

In this work, the focus is on the registration method. Thus, background subtraction and tracking are done using simple methods. Background subtraction is performed using the algorithm proposed in [24], which detects

Table 1: Effect of different phenomenon on background subtraction in visible and in infrared

Phenomenon	Is visible affected?	Is IR affected?
Motion in background	Yes (depends on color of moving/moved object)	Yes (depends on temperature of moving/moved object)
Change in lighting	Yes	Only if lighting change is by a significant heat source (e.g. sun)
Change of scene temperature	Only if heat source is emitting light (e.g. sun)	Yes
Cloths with color similar to background components	Yes	No
Cloths at temperature similar to background components	No	Yes
Reflections on polished metal or glass	Only under some lighting conditions or with mirror surfaces	Yes
Shadows of moving objects	Yes	Only if it causes a significant change in temperature (e.g. light emitting heat source is occluded for a long time

the foreground using the temporal average of the color (or intensity) of each pixel and a threshold. This method is fast but sensitive to lighting (or temperature) variations and it cannot handle periodically changing backgrounds. Table 1 gives the factors that influences background subtraction in visible and in infrared. Both infrared and visible are affected by different phenomenon, but rarely simultaneously, and this is, in fact, why it is useful to use these two modalities. In our case, the background, the lighting, and the scene temperature do not change significantly, because our method is tested indoor. Thus, the results are, in general, reliable enough for the following steps.

For tracking, we use the method of [25] which is based on the overlap between detected foreground regions in two consecutive frames. This algorithm is fast, but it does not handle data associations in complex interactions. The tracking results are good enough to test our proposed registration method, because it is not concerned with object identities. It only needs trajectories of moving foreground regions as input.

The result of tracking is a set of trajectories. Trajectories are formed with the top pixel coordinates (e.g. top of the head) of all the blobs (connected foreground regions) that overlap between two consecutive frames (see figure 1a and b). Let T^i_{video} be the i^{th} trajectory in a given *video* (*right* or *left*). T^i_{video} is defined in a matrix form in homogenous coordinates by

$$T^i_{video} = \begin{bmatrix} X_1 & X_2 & ... & X_n \\ Y_1 & Y_2 & ... & Y_n \\ 1 & 1 & ... & 1 \end{bmatrix}, \tag{1}$$

where n is the number of points in the trajectory. Our tracking method operates on a set of trajectories at each frame.

3.3. Registration algorithm

To register the infrared and visible videos, we find the trajectory set and point set that give the best overlap of an infrared and a visible composite foreground image. For each set of trajectories, we compute a transformation matrix. The infrared composite foreground image is transformed into the referential of the visible composite foreground image using the transformation matrix calculated. The overlap between the two images is then calculated, and the trajectory set and point set that give the best overlap is selected. This results to the final transformation matrix.

3.3.1. Transformation matrix

The goal of registration is to find the transformation matrix H to convert the coordinates of the points of the left video to the referential of the right video (i.e. the homography). That is,

$$Coord_{right} = H \times Coord_{left}, \tag{2}$$

where $Coord_{right}$ are the coordinates in the right video referential and $Coord_{left}$ are the coordinates in the left video referential. In our case, the cameras are fixed on a bar where only translations on the X and Y axes are possible (see figure 2). Rotations around the Z axis are also possible. Furthermore, the cameras can have different zooms. Because of this setup, the H matrix simplifies to an affine matrix [26]. H is then

$$H = \begin{bmatrix} a_{11} & a_{12} & t_y \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where a_{ij} are the rotation around Z and non-isotropic scaling, and t_x and t_y are the translations along X and Y, respectively. To find H, equation 2 is solved using candidate corresponding trajectory points in the left and the right videos, and the normalized Direct Linear Transform (DLT) method [26] to find the least square solution because our set of equations is over-determined.

3.3.2. Composite foreground images

Two composite foreground images are constructed; one for the left video, and another for the right video. This allows our method to perform even if

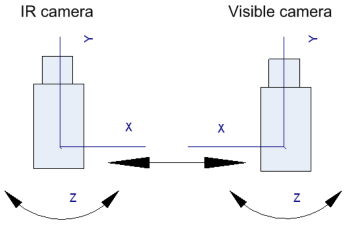
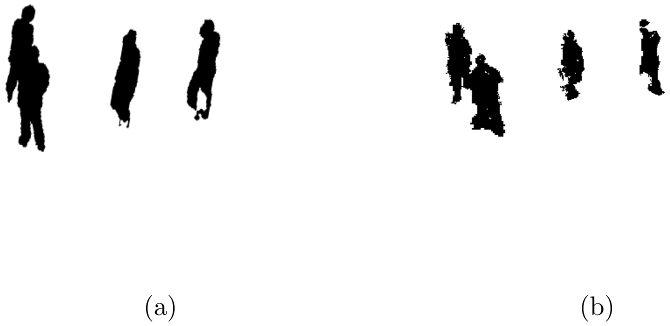


Figure 2: Camera setup

there are few trajectories, as regions restrict more the possible transformations then a few curves. Furthermore, the use of composite foreground images for evaluating the quality of a transformation matrix allows our method to produce a transformation matrix for each video frame pair. We get a good registration result as soon as we get three corresponding trajectory points in both the right and left videos.

Each composite image is composed of the superposition of F binarized foreground images (typically $F = 5$). Our goal is to obtain an image with foreground blobs localized in the four quadrants of the image. The composite image must not be composed of too much blobs because there will be a good overlap whichever the transformation matrix as large shapeless regions will be obtained. Shape is important to restrict transformations and therefore, we need few blobs, but positioned well. A composite foreground image is thus composed of the superposition of the current frame plus the last $F - 1$ foreground images such that there are blobs in each of the four quadrants of the image (if possible). Figure 3 shows composite foreground images for a single actor moving in the field of view of the two cameras. It is the superposition of four frames.

The quality of a transformation matrix is evaluated using as an error



(a) (b)

Figure 3: Composite foreground images for a single actor moving in the field of view of IR and visible cameras. (a) Foreground composite image for IR camera, (b) Foreground composite image for visible camera.

function the overlap error OE of the two composite foreground images with

$$OE = 1 - \frac{N_{left \cap right}}{N_{left \cup right}}, \tag{4}$$

where $N_{left \cap right}$ is the number of overlapping foreground pixels and $N_{left \cup right}$ is the number of foreground pixels from the union of the right image and left image.

We have chosen to build a composite foreground image to verify transformations instead of combining transformation results from many individual foreground images, because we wish to obtain a transformation matrix that can explain the image globally to obtain a unique and stable registration matrix. That is, we are assuming that all image areas include valuable information. A given individual foreground image may give a good transformation matrix only over a small image area since the known information about the whole image is small and the transformation degrees of freedom are not restricted enough (many matrices can explain the transformation of the small known area). The composite foreground images resolve the difficulty of merging disparate transformation matrices, and aim at obtaining information across the image to restrict the possible transformations.

3.3.3. Finding the best set of trajectory points

For registration, we find the correspondence between the different trajectories. A RANSAC-based algorithm is used. It has the following steps:

```
Repeat 1. - 4. until error is sufficiently small
  1. Pick a trajectory pair at random
     Repeat (a) -(e) until error is sufficiently small
       (a) Pick three pairs of points at random in the selected trajectory pair
       (b) Calculate H (equation 2)
       (c) Add participating point pairs
       (d) Recalculate H (equation 2)
       (e) Evaluate overlap error using Euclidean distance
  2. Add participating trajectory pairs
  3. Recalculate H (equation 2)
  4. Evaluate overlap error using equation 4
```

First, a set of all possible corresponding trajectory pairs is constructed. A pair is formed from two trajectories, one from the left video and the other from the right video. Each iteration, a trajectory pair is picked at random using the RANSAC method and a transformation matrix is calculated using corresponding points. Since the videos are synchronized, corresponding points in a trajectory pair are points that have the same timestamp. There are often more than three possible pairs of points in a trajectory pair. Some pairs are inliers, and others are outliers because of tracking errors or top of the head position errors caused by the foreground extraction. That is, trajectories might match only partially because of tracking errors and as a result, all point pairs should not be considered. Thus, three pairs of corresponding points are picked at random using, again, the RANSAC method. The Euclidean distance between left transformed points (using H calculated with equation 2) and there corresponding points in the right video are computed. Pair of points for which the Euclidean distance is smaller than a threshold t (typically, $t = 5$ pixels) are considered as participating point pairs.

The selection of random point pairs for a given trajectory pair is repeated until the error is sufficiently small in the RANSAC algorithm. We use a confidence p of 0.99.

For the selection of the random trajectory pairs, the same principle is used, but the participating trajectory pairs are established using the composite foreground images. For a given trajectory pair, the quality of the transformation is evaluated using equation 4. A participating trajectory pair is a trajectory pair, which decreases the overlap error (value of equation 4).

To summarize our selection of the best corresponding trajectory points, we first use a RANSAC algorithm to select trajectory pairs at random using the composite foreground images to evaluate the quality of H. We use a second RANSAC algorithm to select at random, point pairs inside a given trajectory pair using the Euclidean distance as the criterion for evaluating H. This process results in selecting the best corresponding trajectories and the best corresponding points within corresponding trajectories. The selected points are then used to calculated H.

3.4. Selection of the best transformation matrix

The principle of our method is that the estimation of the affine matrix H should improve as points are added to the trajectories. Thus, earlier estimation should be replaced with a newer if it decreases the overlap error of the composite foreground images. For each frame, equation 4 is computed for the previous H and the new estimation. If overlap relative error E increases with the new estimation, the previous H is kept, if not, it is replaced with the new estimation.

4. Experiments

4.1. Experimental methodology

4.1.1. Data acquisition and setup

For all experiments, a FLIR Thermovision A40 camera (infrared) and a Sony DFW-SX910 camera (color) were used. Videos are 320x240@7,5fps. The cameras were supported by one or two tripods depending on the baseline distance. On the same tripod, the cameras were distanced by 19 cm and on two tripods by approximately 80 cm. All scenarios were filmed from the top to obtain trajectories that are not too linear[1] and in the same plane. Scenarios involve one or more actors (between 1 and 5) and different baseline (19 cm or approximately 80 cm).

4.1.2. Comparison with previous method

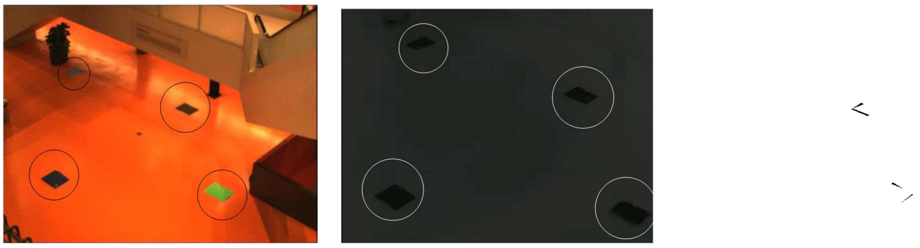
In our method, the transformation matrix for each frame is found by minimizing equation 4, that is the overlap between composite foreground

[1] To compute the affine matrix, the three points in each image must not be collinear.

images. To compare our proposed error function with previous work, we compared our method with the method of [2] in which the error function is the Euclidean distance between the right trajectory points and the transformed left trajectory points. For comparing our error function with theirs, we made a second version of our method which uses this error function instead of ours. This allows us to assess the benefit of our proposed error function. For all results, we show the performance using our proposed error function and the Euclidean error function.

4.1.3. Ground-truth and evaluation metrics

To quantify registration errors, it is necessary to measure the pixel displacement error between the left image and the transformed right image. To do this, some kind of ground-truth and a metric is needed. As ground-truth, we have introduced in the scene for all videos cold square cardboards, visible both in infrared and in color images (see figure 4). Square cardboards were cooled down outdoor at a temperature of -20^oC during a few minutes. The cardboards in infrared are visible only at the beginning of the video when the scene is empty of moving objects, so for both visible and infrared, the cardboards are part of the background model of the scene during processing. Ground-truth binary images were constructed by a human operator selecting manually the corner points of cardboards in the left and right image. Since ground-truths do not move and are part of the scene to be registered, this can be done for only one pair of images. A good transformation matrix should allow overlapping the cardboard areas with small displacement error. The displacement error can be measured either by calculating the maximum displacement of the cardboard corners points or by verifying the overlap error.



(a) (b) (c)

Figure 4: Cold square cardboards used for ground-truth. (a) Cardboards in color image, (b) Cardboards in infrared image, (c) An overlap error of the cardboards of 16.5% ($E = 0.165$).

We have chosen to calculate the overlap error as the implementation of our method has already the functionality for such a task.

The overlap error E is calculated as follows:

1. Create ground-truth by selecting the corners of the cardboards for a test video. This gives two binary images of the cardboards in the visible and infrared (CB_{left} and CB_{right}) that are used at each frame to verify the quality of registration;
2. Process each frame f of the same test video. Apply our method to find transformation matrix for frame f using proposed (composite foreground images) or Euclidean error function.
3. Register the two cardboard binary images using the transformation matrix obtained.
4. Compute the overlap error with:

$$E = 1 - \frac{P_{CB_{left} \cap CB_{right}}}{P_{CB_{left} \cup CB_{right}}}, \tag{5}$$

where $P_{CB_{left} \cap CB_{right}}$ is the number of overlapping ground-truth cardboard pixels and $P_{CB_{left} \cup CB_{right}}$ is the number of ground-truth card-

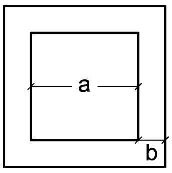


Figure 5: Typical displacement and overlap error geometry.

board pixels from the union of the cardboards binary images (CB_{left} and CB_{right}).

Note that the cardboards are not part of the foreground in the videos, and thus they are not used to find the transformation matrix with our proposed method. Our proposed error function minimizes the overlap error of composite foreground images, not the overlap of the cardboards. As such, our method is not favoured even thought the evaluation metric has the same formulation as the proposed error function. Also, we do the transformation matrix computations continuously for evaluation purposes, but could be stop after a given number of frames.

Relative mean displacement error b is related to our relative overlap error E approximately in the following way (see figure 5):

$$E = 1 - \frac{a^2}{(a+2b)^2} \tag{6}$$

$$E(a+2b)^2 = (a+2b)^2 - a^2 \tag{7}$$

$$a = \sqrt{(a+2b)^2 - E(a+2b)^2} \tag{8}$$

$$(a+2b) - 2b = \sqrt{(a+2b)^2 - E(a+2b)^2} \tag{9}$$

Since $a + 2b = 1$,

$$1 - 2b = \sqrt{1 - E} \tag{10}$$

And thus,

$$b = \frac{1 - \sqrt{1 - E}}{2} \tag{11}$$

From equation 11, it means that for an overlap error $E = 0.1(10\%)$, we obtain a relative mean displacement error of b=0.026. Thus, for an average cardboard size of 25 pixels, the absolute mean displacement error is about 0.6 pixel ($b \times 25$). For $E = 0.2$, it is about 1.3 pixels, for $E = 0.3$, it is about 2 pixels, and for $E = 0.4$ it is about 2.8 pixels.

Furthermore, we have compared the automatic registration results with manual registration. The ground-truth affine transformation matrix was calculated from the corners points of the cardboards using equation 2. The error E (equation 5) is also calculated for the cardboards to be used as a reference. This is what we consider the ground-truth error. It is not necessarily zero, because the points are selected manually (see figure 4(c)). Furthermore, to verify the overlap of the cardboards and to compute E, we have to cut them out manually from the color and infrared images and in general, they are not cut out perfectly as the boundaries are not always sharp.

Since RANSAC is not a deterministic algorithm, to have statistically sound results, each experiment was repeated 30 times [27]. Our results are statistics over these repetitions, and they are the mean, minimum, median and standard deviation (σ) of E for the 30 repetitions at each frame. To synthesize the results, we show in the tables, the mean of three statistical measures (mean, minimum and σ) at each frame for the complete videos. That is for a frame, we calculate the mean overlap error E (out of 30 repetitions), and for a test scenario, we do the mean of the means of each frame. The mean of the means ($\mu_{\overline{E}}$) is given by

$$\mu_{\overline{E}} = \frac{\displaystyle\sum_{j=1}^{NF} \frac{\sum_{i=1}^{30} E_i^j}{30}}{NF}, \tag{12}$$

where E_i^j is the overlap error of a repetition i of RANSAC for the $j$ith frame, and NF is the number of frames in the test video. We did the same for the minimum and σ (mean of the minimums (μ_{min}) and mean of σ (μ_σ)).

4.2. Results and discussion

Table 2 gives the mean of the minimums (μ_{min}) for each scenario. It shows that the use of the composite foreground images allows us to obtain results that are consistently closer to the ground-truth compared to the use of only Euclidean distance between trajectory points. Thus, the use of blobs as a quality criterion to evaluate a transformation matrix stabilizes the results. Figure 6 shows the complete results for scenario 3. This figure shows that the mean and σ change a lot at every frames when using only the Euclidean error function. Furthermore, we get better results faster (within 130 frames for 30% error compared to around 170 frames). The foreground composite images restrict significantly more the number of possible transformations. Thus, we believe the use of this strategy is an important contribution to a feature point-based method.

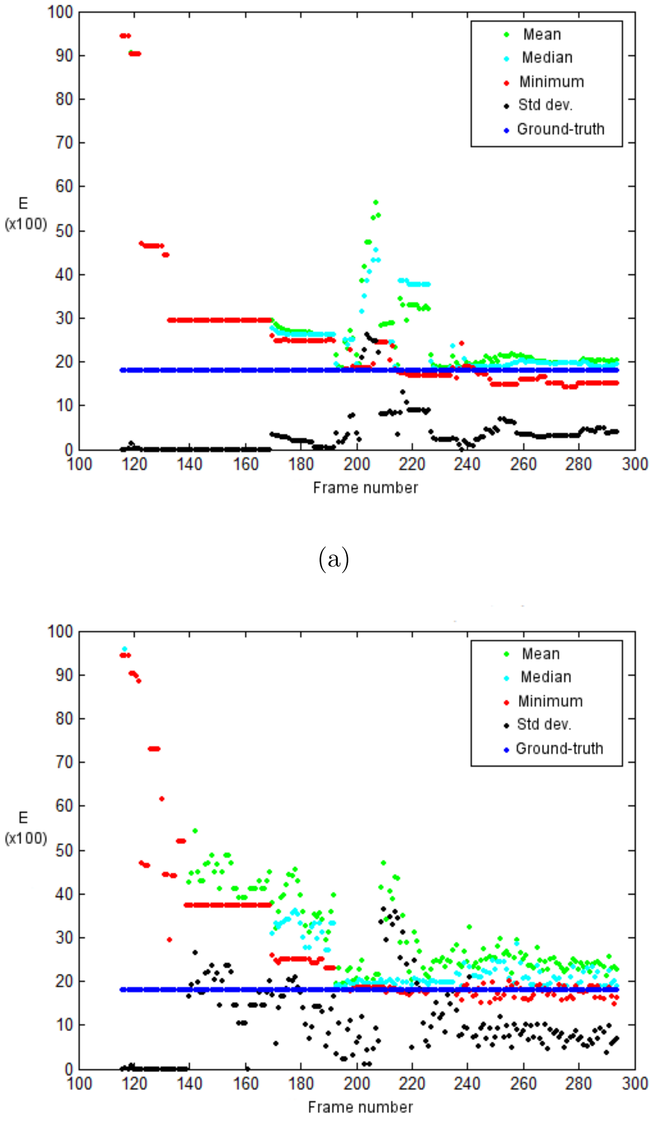
The reader can notice an increase in σ between frames 200 and 220 (figure 6). In this interval, there are many background subtraction errors, and the transformation between the two videos becomes harder to establish. In fact, in this interval, the foreground is not well detected, and there are less foreground pixels in the visible composite image which makes the overlap

Table 2: Results for nine scenarios (mean of the minimums (μ_{min}) at each frame)

Scenario	Ground-truth	Proposed error function	Euclidean error function
1	0.210	**0.267**	0.323
2	0.205	0.403	**0.395**
3	0.178	**0.251**	0.280
4	0.183	**0.184**	0.300
5	0.165	**0.407**	0.541
6	0.119	**0.221**	0.253
7	0.112	0.152	**0.139**
8	0.274	**0.264**	0.334
9	0.123	**0.085**	0.112

more ambiguous. Since the selection of the transformation matrix relies on OE (see 3.4), we may select a less accurate matrix (w.r.t. ground-truth and E) than the previous one. To solve this, we could make statistics on the best transformation matrices found at every frame, and select a matrix that minimizes OE for many frame. We could also build the transformation matrix from the matrices of many frames.

Sometimes our algorithm gets results with error E smaller than the ground-truth. It is possible and desirable because the cardboards are not cut out perfectly and the ground-truth is selected by hand, so there is a margin of error and we expect automatic registration to be more precise than a manual one. It is possible that we did not select accurately the corners of the cardboards. Notice that the Euclidean error method can also give results



(a)

(b)

Figure 6: Results for scenario 3. (a) Proposed error function, (b) Euclidean error function.

better than the ground-truth. For the mean of the minimums (μ_{min}), our method is outperform slightly two times by the Euclidean error method.

If we now consider table 3, our method gives in general mean transformation matrices that are closer to the ground-truth. It is not the case for scenario 2 and 7. For the second scenario, the results are about the same, but for scenario 7, our method is significantly outperformed. Figure 7 details the result. Our method is still more stable compared to the Euclidean error method, but the transformation error stays large. This test scenario involves five actors and as a result much more trajectories. In this case, the Euclidean error method can find better results. In fact, this method is design to perform well with many trajectories, while our method aims at solving the case where they are fewer trajectories. For this scenario, the composite foreground images become very crowded, and a bad transformation (w.r.t. ground-truth) can give a good overlap. To solve this, we should verify that the foreground blobs in the composite foreground images do not cover an area that is too large, and in such case, consider fewer frames to superpose in the composite foreground images.

The two methods do not work as well when the baseline is larger (scenario 5), because there is less overlap between the two images. Thus, there are more ambiguities and fewer points to pair.

Finally, table 4 gives the mean of the standard deviations (μ_σ) for each frames. Except for scenario 7, for the reasons previously explained, our method outperforms the Euclidean error method. If not, the results are about the same.

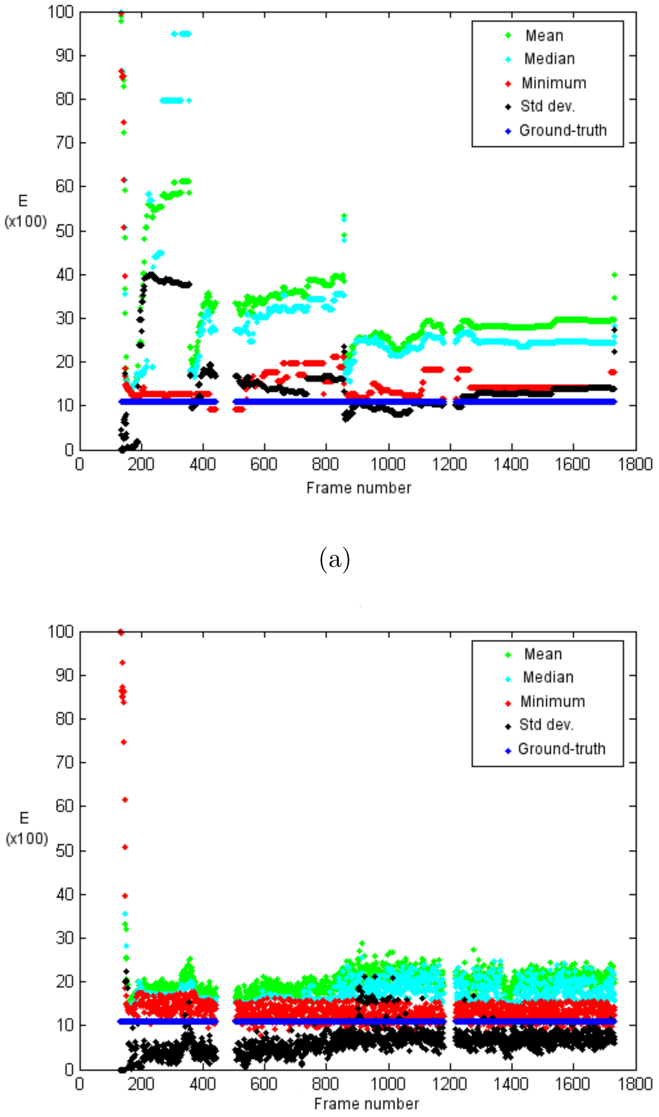
Figure 8 shows actual registration results obtained for frame 48 of scenario

Table 3: Results for nine scenarios (mean of the mean ($\mu_{\overline{E}}$) at each frame)

Scenario	Ground-truth	Proposed error function	Euclidean error function
1	0.210	**0.329**	0.601
2	0.205	0.419	**0.416**
3	0.178	**0.294**	0.351
4	0.183	**0.270**	0.350
5	0.165	**0.522**	0.700
6	0.119	**0.284**	0.302
7	0.112	0.328	**0.206**
8	0.274	**0.348**	0.404
9	0.123	**0.177**	0.201

Table 4: Results for nine scenarios (mean of σ (μ_{σ}) at each frame)

Scenario	Ground-truth	Proposed error function	Euclidean error function
1	0	**0.064**	0.116
2	0	0.021	**0.012**
3	0	**0.035**	0.109
4	0	0.065	**0.055**
5	0	**0.060**	0.072
6	0	**0.057**	0.058
7	0	0.152	**0.062**
8	0	0.082	**0.075**
9	0	**0.085**	0.113



(a)

(b)

Figure 7: Results for scenario 7. (a) Proposed error function, (b) Euclidean error function.

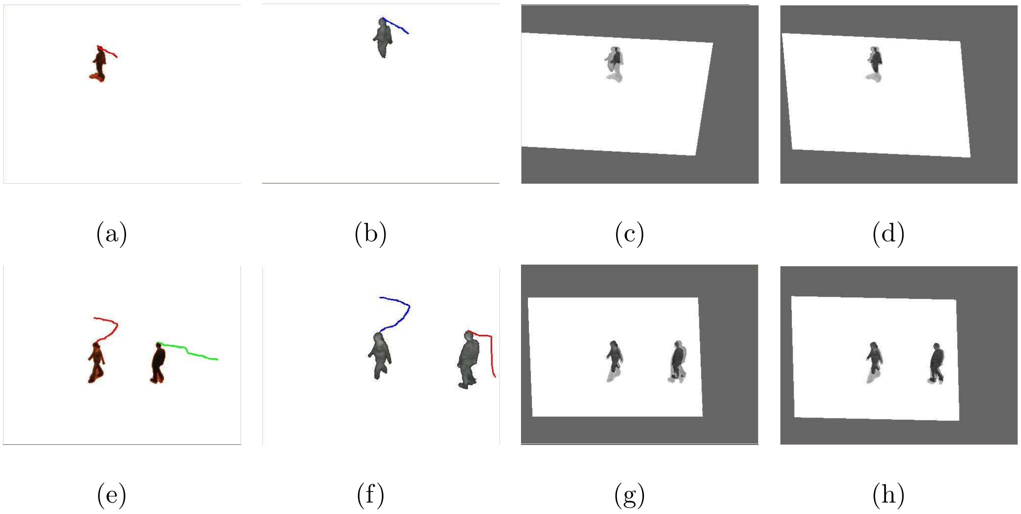


Figure 8: Registration results for two pairs of frames. (a) to (d), frame 48 from scenario 1: (a) Visible image and computed trajectory, (b) IR image and computed trajectory, (c) Registration of (a) and (b) with Euclidean error function, (d) Registration of (a) and (b) with our proposed error function. (e) to (h), frame 98 from scenario 9: (e) Visible image and computed trajectories, (f) IR image and computed trajectories, (g) Registration of (e) and (f) with Euclidean error function, (h) Registration of (e) and (f) with our proposed error function.

1 and for frame 98 of scenario 9. In the first case (Figure 8 a,b,c,d), because of the small number of trajectory points, the transformation matrix found using Euclidean error function does not give a good registration. Conversely, using our proposed error function allows a better registration because the region overlap constrains more the set of possible transformations. In the second case (Figure 8 e,f,g,h), the registration is again better with our proposed error function, because even if more trajectory points are available, their positioning might not be exact, and thus transformations are less constrained than with a region.

4.3. General discussion

Globally, the use of the composite foreground images allows our method to perform better when there are fewer trajectories compared to the use of Euclidean distance on trajectory points. Furthermore, it stabilizes the transformation matrix obtained and the RANSAC algorithm gives more consistently the same result. This is because we are registering information across the whole image, and thus the possible transformations are more restricted. However, when there are more actors, our composite foreground images might get too crowded. It this case, it should be composed of less frames (i.e. F should be smaller than 5).

The core of our method, our proposed overlap error function, is not restricted to affine transformation, as any registration, whichever the 2D homography matrix used, should overlap images almost perfectly. Thus, the goal of having a good level of overlap everywhere in an image is valid for any 2D homography matrix. Our method could be applied for projective 2D homography by just changing matrix calculation. If the matrix is not changed, the overlap will not be as good, but since in the composite foreground images there should be objects all across their areas, the overlap criterion will still restrict as much as possible the possible transformations as desired.

Synchronization may affect the results because synchronization error will cause different or displaced shapes to be overlapped. In this work, we assume that the cameras are synchronized. Actually, since they are synchronized by software, images from both cameras are obtained sequentially, and the frame rate is relatively low, images are in fact not perfectly synchronized (see for example figure 1a) and b), the spacing between the legs of the right actor is different). The desynchronization of the legs does not affect too much the result because it corresponds to small areas. However, the positional error for large desynchronization will cause more problems. Fortunately, the videos may be synchronized by the method proposed by [2]. Applying this method would have reduced the errors we have calculated in our experiments for the two methods, but with similar conclusions.

Object detection also affects the results. The top of the head position might change and the areas to overlap will have different shapes. By using composite foreground images, local background subtraction errors will not cause significant overlap errors since the total area to overlap is large. Background subtraction errors will cause difficulties only if they affect a large part of the image during many frames, because our foreground composite images include frames at different moment in time. The trajectories may be significantly affected too by background subtraction errors. Some trajectory points will be incorrectly positioned. They will be rejected based on our error function since they will not produce good overlap of foreground composite images. In fact, trajectories and object regions will be affected independently in general since we use the top of the head for trajectories. For example, if the head of a human is missing, there will be an error in the trajectory but the body region to overlap will be large and almost complete. Whereas if only the head is detected, the trajectory will be good, but there will be a large overlap error for this human.

5. Conclusion

In this paper, we presented a method and a novel criterion to register infrared and color (visible) videos. It is a feature point-based method that uses top pixel coordinates found after foreground detection and tracking to build trajectories in both visible and infrared videos. Then, the trajectory points are used to find the transformation matrix that is obtained using a RANSAC algorithm and composite foreground images as quality criterion.

The results obtained show that the use of composite foreground images as a registration criterion give results that are more stable compared to a criterion on trajectories. Furthermore, it allows working with video in which there are few trajectories. In general, the results are close to the ground-truth. Because our proposed method aligns silhouettes instead of trajectory points, it is appropriate for method performing image fusion using blob contours or edges.

Future works are to improve the construction of the foreground composite image to ensure that it is not too crowded, which reduces its benefit, and to test the method with a more sophisticated tracking method to obtain better trajectory points. We should also design a method to assess the quality of the foreground detection. Registration could be performed only when foreground detection is acceptable. In addition, the selection of the best transformation matrix for the complete video should be improved as noted in the discussion.

We also aim at integrating this method with tracking to design a day/night tracking system that combines information from both infrared and visible sensors using a feedback between tracking and registration.

6. acknowledgements

We would like to thank the Canadian Foundation for Innovation (CFI) and the Fonds québécois de la recherche sur la nature et les technologies (FQRNT) for their support with grants.

References

[1] S. C, K. Tieu, Automated multi-camera planar tracking correspondence modeling, Vol. 1, 2003, pp. I–259–I–266 vol.1.

[2] Y. Caspi, D. Simakov, M. Irani, Feature-based sequence-to-sequence matching, International Journal of Computer Vision 68 (1) (2006) 53–64.

[3] L. Lee, R. Romano, G. Stein, Monitoring activities from multiple video streams: establishing a common coordinate frame, Pattern Analysis and Machine Intelligence, IEEE Transactions on 22 (8) (2000) 758–767.

[4] M. A. Fischler, R. C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Communications of the ACM 24 (6) (1981) 381–395.

[5] P. J. Burt, R. J. Kolczynski, Enhanced image capture through fusion, in: Computer Vision, 1993. Proceedings., Fourth International Conference on, 1993, pp. 173–182.

[6] J. W. Davis, V. Sharma, Background-subtraction using contour-based fusion of thermal and visible imagery, Comput. Vis. Image Underst. 106 (2-3) (2007) 162–182.

[7] B. Zitova, J. Flusser, Image registration methods: a survey, Image and Vision Computing 21 (2003) 977–1000.

[8] A. Roche, G. Malandain, X. Pennec, The correlation ratio as a new similarity measure for multimodal image registration, in: Medical Image Computing and Computer-Assisted Interventation (MICCAI98), Vol. 1496, 1998, pp. 1115–1124.

[9] P. Anuta, Spatial registration of multispectral and multitemporal digital imagery using fast fourier transform techniques, Geoscience Electronics, IEEE Transactions on 8 (4) (1970) 353–368.

[10] P. Viola, W. M. Wells, III, Alignment by maximization of mutual information, Int. J. Comput. Vision 24 (2) (1997) 137–154.

[11] S. K. Kyoung, H. L. Jae, B. R. Jong, Robust multi-sensor image registration by enhancing statistical correlation, in: Information Fusion, 2005 8th International Conference on, Vol. 1, 2005, pp. 380–386.

[12] S. J. Krotosky, M. M. Trivedi, Mutual information based registration of multimodal stereo videos for person tracking, Computer Vision Image Understanding 106 (2-3) (2007) 270–287.

[13] H. Chen, P. K. Varshney, M.-A. Slamani, On registration of regions of interest (roi) in video sequences, in: AVSS '03: Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, IEEE Computer Society, Washington, DC, USA, 2003, pp. 313–318.

[14] Z. Liu, R. Laganiere, Registration of ir and eo video sequences based on frame difference, 2007, pp. 459–464.

[15] X. Huang, Z. Chen, A wavelet-based multisensor image registration algorithm, in: Signal Processing, 2002 6th International Conference on, Vol. 1, 2002, pp. 773–776.

[16] M. I. Elbakary, M. K. Sundareshan, Multi-modal image registration using local frequency representation and computer-aided design (cad) models, Image and Vision Computing 25 (5) (2007) 663–670.

[17] E. Coiras, J. Santamaria, C. Miravet, Segment-based registration technique for visual-infrared images, Optical Engineering 39 (2000) 282–289.

[18] S. G. Kong, J. Heo, F. Boughorbel, Y. Zheng, B. R. Abidi, A. Koschan, M. Yi, M. A. Abidi, Multiscale fusion of visible and thermal ir images for illumination-invariant face recognition, International Journal of Computer Vision 71 (2) (2007) 215–233.

[19] O. Charoentam, V. Patanavijit, S. Jitapunkul, A robust region-based multiscale image fusion scheme for mis-registration problem of thermal and visible images, in: ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition, IEEE Computer Society, Washington, DC, USA, 2006, pp. 669–672.

[20] J. W. Joo, J. W. Choi, D. L. Cho, Robust registration in two heterogeneous sequence images on moving objects, in: Information Fusion, 2003. Proceedings of the Sixth International Conference of, Vol. 1, 2003, pp. 277–282.

[21] T. Hrkac, Z. Kalafatic, J. Krapac, Infrared-visual image registration

based on corners and hausdorff distance, in: Image Analysis, 2007, pp. 383–392.

[22] J. Han, B. Bhanu, Fusion of color and infrared video for moving human detection, Pattern Recognition 40 (6) (2007) 1771–1784.

[23] Y. Caspi, M. Irani, Aligning non-overlapping sequences, Int. J. Comput. Vision 48 (2002) 39–51.

[24] B. Shoushtarian, H. E. Bez, A practical adaptive approach for dynamic background subtraction using an invariant colour model and object tracking, Pattern Recognition Letters 26 (1) (2005) 5–26.

[25] L. M. Fuentes, S. A. Velastin, People tracking in surveillance applications, Image and Vision Computing 24 (11) (2006) 1165–1171.

[26] R. Hartley, A. Zisserman, Multiple view geometry in computer vision, 2nd Edition, Cambridge University Press, Cambridge, UK, 2003.

[27] B. Ostle, Engineering statistics : the industrial experience, 1st Edition, Duxbury Press, Belmont, Montreal, 1996.