Detecting Moving Objects in Airborne Forward Looking Infra-Red Sequences

Alexander Strehl and J. K. Aggarwal [*]
Computer and Vision Research Center
The University of Texas at Austin
Department of Electrical and Computer Engineering
Austin, TX 78712-1084, U.S.A.
{strehl,aggarwaljk}@mail.utexas.edu

Abstract

In this paper we propose a system that detects independently moving objects (IMOs) in forward looking infra-red (FLIR) image sequences taken from an airborne, moving platform. Ego-motion effects are removed through a robust multi-scale affine image registration process. Consequently, areas with residual motion indicate object activity. These areas are detected, refined and selected using a Bayes' classifier. The remaining regions are clustered into pairs. Each pair represents an object's front and rear end. Using motion and scene knowledge we estimate object pose and establish a region-of-interest (ROI) for each pair. Edge elements within each ROI are used to segment the convex cover containing the IMO. We show detailed results on real, complex, cluttered and noisy sequences. Moreover, we outline the integration of our robust system into a comprehensive automatic target recognition (ATR) and action classification system.

1 Introduction

1.1 Motivation

Forward looking infra-red (FLIR) images are frequently used in automatic target recognition (ATR) applications. ATR is a generic term used for a variety of semi-automated and automated operations ranging from cuing a human observer to potential targets to fire-and-forget. Many researchers have investigated various approaches to detection, recognition and pose estimation of targets from *static* FLIR images. A comprehensive recent review by Ratches, Walters, Buser and Guenther on techniques for image-based ATR systems can be found in [17].

A variety of techniques to *detect* targets in static images have been proposed. Early work often was data-driven and used ad hoc methods such as thresholding based on the contrast of an object compared to the local background or pixel statistics. Later algorithms used knowledge-based systems and template matching approaches. More recent research focuses on model-based approaches and multi-sensor fusion [16, 18, 5]. While common ATR systems can track objects based on a series of single-frame detections, motion has been neglected as a cue to target detection and pose estimation. Motion information can be a very strong aid for finding targets in images. It can be motivated biologically to use image motion as a low-level segmentation aid rather than a post-processing result from single frame image analysis [23]. Consequently, including *dynamic* scene information to a static ATR system adds an independent criterion that can significantly increase detection rates and decrease false alarms.

Today, many techniques exist for the motion analysis of visual imagery [6, 2, 14, 13, 20]. Irani and Anandan differentiate scenes and the appropriate algorithms along a 2D to 3D continuum [13]. In 2D analysis the scene can be approximated by a flat surface and the camera is undergoing mainly rotations and zooms. 3D scenes are characterized by significant depth variations in the scene and a translating camera. Motion models have to be appropriate for the processing environment.

In this paper, we present a motion-based object detection system tailored for FLIR sequences. Our FLIR sequences are taken from a moving platform and depict scenery as well as independently moving objects (IMOs). This case represents the most general (and most difficult) scenario of motion processing. Observer-motion (ego-motion) and object motions induce *multiple* coupled motions into the FLIR images. In our approach, we compensate for the observer-motion, which makes the background stationary. After re-

[*] This research was supported in part by the Army Research Office under contracts DAAH04-95-I-0494 and DAAG55-98-1-0230, and by the Texas Higher Education Coordating Board, Advanced Research Project 97-ARP-275.

moving the effects of ego-motion, residual motions must be due to moving objects. We use these residual motion areas to detect and segment the targets and estimate their pose.

1.2 FLIR Versus Visible

To detect IMOs in FLIR image sequences, the sensor properties have to be taken into account. We face additional challenges caused by the following important differences to visual sequences:

- FLIR imagery smoothes out object edges and corners. This leads to a reduction of distinct features.
- The generation and maintenance of kinetic energy usually heat up a moving object (e.g., friction, engine combustion). Consequently, moving objects often appear brighter than the background.
- FLIR images are noisy and have less contrast. Moreover, they often contain artifacts such as dirt on the lens, brightness which fades out at the end of the scanline, or local sensor failure at certain pixel locations.
- FLIR sequences are not easily available (especially not from controlled experiments) and have a lower resolution. The sequences available to us are 128x128 pixels as compared to 512x512 pixels and more of standard visual cameras.
- FLIR sequences are often taken under difficult circumstances and may have abrupt discontinuities in motion.

Extracting 3D structure and motion information from an image is an under-constrained problem that can only be solved in special cases (such as by translational ego-motion with sufficient scene texture). Moreover, the aperture may limit our view to image regions with structure that is insufficient to estimate motion. Accurate motion estimation needs distinct features – corners are good, line segments can be sufficient, but homogeneous areas are useless. This is known as the generalized aperture problem [21]. The limitations and difficulties of FLIR imagery must be taken into account. They demand more robust (less noise sensitive) techniques than visual sequences. Consequently, we preferred a 2D robust motion model to highly noise sensitive 3D approaches. Moreover, FLIR processing is often performed in a time-critical setting. This requires algorithms to be very efficient and/or to be capable of highly parallel execution.

1.3 Organization

Figure 1 shows a graphical overview of our proposed system. In the first module, we compensate for shortcomings in image quality such as low contrast, artifacts and noise (Section 2). Thereafter, we perform robust multi-scale affine image registration to eliminate effects from the motion of the camera platform (Section 3). Then, candidate regions for object parts are obtained by analyzing the residual misalignment. Using properties of the scene and the sensor, we remove unlikely regions and identify pairs of front and rear parts. Together with edge elements, we obtain a convex cover for the IMOs' locations in the image (Section 4). Section 5 demonstrates experimental results and Section 6 summarizes the proposed system and suggests integration into a comprehensive framework.

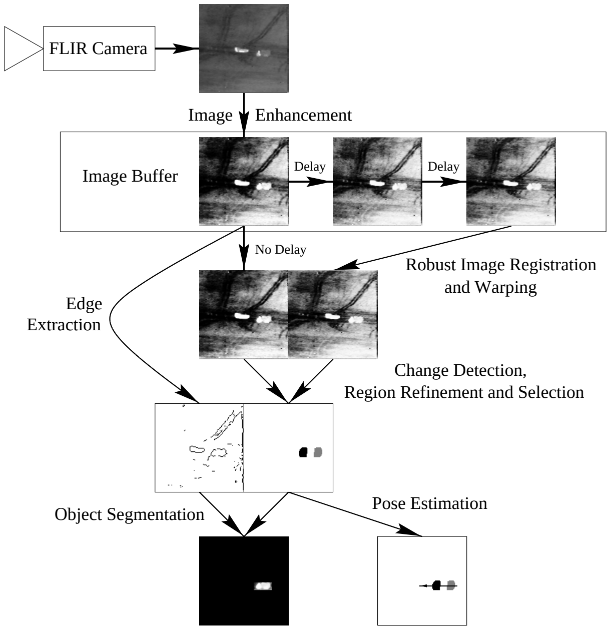

Figure 1. Overview of our independently moving object (IMO) detection system.

2 Image Enhancement

Since motion processing is noise sensitive and FLIR images are of a low quality, we enhance the images to reduce misleading shortcomings before proceeding with the motion-based object detection and analysis. In images recorded from a moving platform, artifacts appear as candidates for independently moving objects since they do not move coherently with the scene leading to false alarms. To prevent this, the incoming frames are filtered before further processing. Locations with artifacts rarely undergo small modifications (such as Gaussian noise), but often have completely erroneous gray-level values (such as salt-and-pepper noise). Consequently, a morphologic filter (such as an order statistic filter) is more suitable than a linear filter (such as a mean filter). We decided to use a median filter. It successfully removes small artifacts and image noise while preserving relevant edge information.

FLIR images are based on the thermal electro-magnetic spectrum. Differences within a scene's background are rather small compared to differences between background and objects. This leads, in general, to a very low contrast in most of the image area. In order to enhance the background feature points that will be needed later to properly compensate for image motion, we normalize the contrast for an incoming frame by histogram equalization. This technique remaps the gray-level values (order preserving) in the image such that the cumulative histogram has an approximately linear slope. In the next section we will discuss how the effects of camera motion are removed from the enhanced sequence.

3 Robust Multi-scale Affine Registration

Moving objects induce motion in an image sequence. Since their image motion is independent (and different) from the image motion caused by the camera's movement, they are referred to as independently moving objects (IMOs). In the case of airborne imagery, the objects are moving on the ground and appear rather small. Consequently, the background of the scene will cover most of the image. The dominant motion explains most of the apparent motion in an image. The background in the image undergoes displacement caused by the observer's movement (or ego-motion) and, hence, constitutes the dominant motion. When ego-motion prevails over most of the image, IMOs can also be understood as objects whose motion violates the dominant motion model. In order to detect such objects, we remove the effects of the prevailing (dominant) motion from the sequence. This leaves only the effects of secondary and smaller motions (the independent motions).

Due to the high noise and the eventually large displacements, we have to use the entire image and can not rely on a windowed approach to compensate for motion. 3D (or more precisely 2.5D) models require a depth map from the scene. This depth map can be either given or estimated from the sequence, if sufficient translational ego-motion is present [1, 13]. While 3D models have a small model error (bias), they are prone to high estimation error (variance) due to their high number of degrees-of-freedom (one unknown depth parameter for every location in the image plus rigid-motion parameters). The 2D affine model with its six degrees-of-freedom provides a good balance for the bias-variance-tradeoff, especially when considering the FLIR shortcomings and the noise sensitivity of motion estimation. An estimator is robust if outliers can not arbitrarily worsen the estimate. By applying robust statistics [12] to motion estimation [6], the dominant motion estimate can be made invariant to small model-violations such as IMOs or minor depth discontinuities (parallax). The selection of the motion model is crucial to the success of compensating for camera motion.

There are several ways to estimate and compensate for the dominant observer motion. Feature-based motion estimation [24, 7, 20] seems inappropriate because very few features are present. These tend to be IMOs and would consequently disturb the ego-motion estimation. Abrupt strokes to the camera make spatio-temporal filtering approaches [22] ineffective, too. The best method appears to be a registration technique that uses the entire image and is able to handle large displacements while being robust against the violations by object motion. Since the moving objects are very small in airborne images (maximally 10% of image area), we can assume that camera motion is the dominant motion in the scene. For our system, we use the entire image in a robust multi-scale affine image registration [4]. This aligns a frame $\mathbf{I}$ to a reference frame $\mathbf{I}'$, assuming an affine transformation of the homogeneous coordinates [10] as described in equation 1.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \mathbf{M} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \qquad (1)$$

$$\mathbf{M} = \begin{pmatrix} \theta_1 & \theta_2 & \theta_3 \\ \theta_4 & \theta_5 & \theta_6 \\ 0 & 0 & 1 \end{pmatrix} \qquad (2)$$

We always use the most recent (current) frame as the reference frame. The motion transformation $\mathbf{M}$ is estimated in four stages [4], as described in the following subsections.

3.1 Pyramid Construction

A Laplacian image resolution hierarchy is created to allow processing on various spatial frequencies levels. In a Laplacian pyramid, the image is decomposed into one low-resolution low-pass filtered image and multiple higher-resolution layers encoding the higher frequencies [8]. We start motion estimation at the lowest resolution level and expand and refine the results layer by layer until the original resolution is reached.

3.2 Motion Estimation

Most motion estimation paradigms are based on image intensity conservation. Intensity conservation assumes that during a sufficiently small time τ, no intensity pattern in the image gets lost. However, it may get displaced by u and v in x and y direction as expressed by equation 3, which was proposed by Horn and Schunk in [11].

$$\mathbf{I}^t(x, y) = \mathbf{I}^{t+\tau}(x + u^t(x, y), y + v^t(x, y)) \qquad (3)$$

$\mathbf{I} = \mathbf{I}^t$ and $\mathbf{I}' = \mathbf{I}^{t+\tau}$ represent the image intensity as a function of (x, y) at time t and $t + \tau$, respectively. In each layer of the Laplacian pyramid, motion is estimated. We use an iterative estimator that minimizes the sum-of-squared differences (SSD) between the reference frame $\mathbf{I}'$ and the registered frame $\hat{\mathbf{I}} = \mathbf{M} \cdot \mathbf{I}$.

The initial motion guess is ‘no motion’ and, hence, the transformation matrix equals the unity matrix at iteration 0 ($\mathbf{M}_0 = \mathbf{1}$). The SSD is an error measure based on the intensity conservation assumption [11] and defined at iteration n as follows:

$$\mathrm{SSD}_n = \sum_{x,y} (\mathbf{I}' - \hat{\mathbf{I}}_n)^2 \tag{4}$$

Using the Gauss-Newton method to minimize the SSD error in respect to the motion parameters, we obtain an incremental parameter update δ_n for $\hat{\theta}_n$ as given by equations 5, 6 and 7.

$$\begin{aligned} \delta_n = \; & -\left(\textstyle\sum_{x,y} \mathbf{P}^{\mathbf{T}}(\nabla\mathbf{I})(\nabla\mathbf{I})^{\mathbf{T}}\mathbf{P}\right)^{-1} \cdot \\ & \cdot \left(\textstyle\sum_{x,y} \mathbf{P}^{\mathbf{T}}(\nabla\mathbf{I})(\mathbf{\Delta I}_n)\right) \end{aligned} \tag{5}$$

$$\hat{\theta}_n = \begin{pmatrix} \hat{\theta}_3 & \hat{\theta}_1 & \hat{\theta}_2 & \hat{\theta}_6 & \hat{\theta}_4 & \hat{\theta}_5 \end{pmatrix}^{\mathbf{T}} \tag{6}$$

$$\mathbf{P} = \begin{pmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{pmatrix} \tag{7}$$

The residual error $\mathbf{\Delta I}_n$ is computed as the pixel-wise difference between the reference frame and the registered frame $\mathbf{\Delta I}_n = \mathbf{I}' - \hat{\mathbf{I}}_n$. The image gradient $\nabla\mathbf{I}$ is approximated by filtering the image with the Sobel kernel [19] for the horizontal and vertical direction (Figure 2).

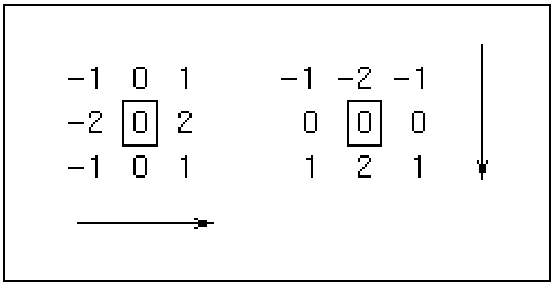

Figure 2. Sobel edge filter. Linear filter kernels for x- and y-direction.

3.3 Image Warping

The current motion estimate $\mathbf{M}_n$ at iteration n is used to warp the earlier image $\mathbf{I}$ so it matches the reference image $\mathbf{I}'$. We employ a standard warping technique using bilinear interpolation. The warped image $\hat{\mathbf{I}}_n = \mathbf{M}_n \cdot \mathbf{I}$ is used instead of the original frame $\mathbf{I}$ and the motion estimation process is repeated. Motion estimation and image warping are iterated with the updated image $\hat{\mathbf{I}}_n$ and the reference frame $\mathbf{I}'$. Iteration is terminated upon reaching a fix-point for the motion estimate ($\delta_n = 0$) or the maximum number of iterations. The selection of the maximum number of iterations depends on the expected magnitude of inter-frame motion (typically between 3 and 10 iterations).

3.4 Refinement

The estimates are refined by expanding the results within the resolution pyramid in a coarse-to-fine fashion (Figure 3). This prevents aliasing of high spatial frequency components that undergo large motions and minimizes the outlier sensitivity. It also speeds up the motion analysis since fewer iterations are required at each resolution level [4, 3].

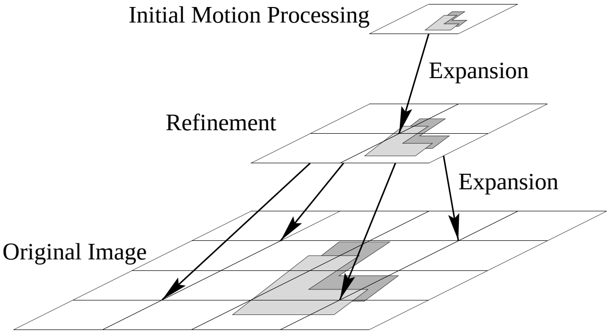

Figure 3. Illustration of coarse-to-fine motion processing.

4 Locating Moving Objects

4.1 Change Detection and Region Refinement

After the effects of camera motion have been removed, the remaining regions with significant changes may contain IMOs. To determine which regions exhibit significant change, we first compute the difference of the current image (the reference image) to a registered frame from the past (e.g., 0.2 seconds ago or 5 frames at 25 frames per second). The time difference between the frames must be long enough so the IMO moves significantly between them, (e.g., its movement is detectable in the image considering resolution and target distance). Locations exceeding a threshold difference are considered outliers to the background motion and constitute our initial change regions. These are then processed with morphological operations such as erosion (each pixel adopts the lowest value in its neighborhood) and dilation (each pixel adopts the highest value in its neighborhood). Iterative application of opening operations (erosion followed by dilation) in a 3x3 neighborhood eliminates singletons and small regions. The opening is repeated until the image no longer changes. A final dilatation operation grows the remaining regions.

$$\mathbf{I_r} = \text{dilate}(\lim_{n \to \inf}(\text{open}^n(\text{bin}(\mathbf{I'} - \mathbf{M} \cdot \mathbf{I})))) \tag{8}$$

The resulting image $\mathbf{I_r}$ contains the candidate regions for IMO parts.

4.2 Region Selection and Pose Estimation

Some candidate regions may not correspond to a moving object. For example, heavy noise, artifacts or partial sensor failure could induce such false alarm regions. To eliminate false alarms, we compute four features $s_{i,1}$ to $s_{i,4}$ for each candidate region in $\mathbf{I_r}$. The symbol X_i denotes the set of points in a particular region i. The r-th (central) momentum of X_i is denoted m_r^0 (m_r^1).

$$m_r^c = E\left[(X_i - c \cdot E[X_i])^r\right] \tag{9}$$

$$s_{i,1} = \quad m_1^0 \quad = \text{mean} \tag{10}$$

$$s_{i,2} = \quad m_2^1 \quad = \text{variance} \tag{11}$$

$$s_{i,3} = \quad E\left[\frac{m_3^1}{\sqrt{m_2^1}^3}\right] \quad = \text{skewness} \tag{12}$$

$$s_{i,4} = \quad E\left[\frac{m_4^1}{\sqrt{m_2^1}^4}\right] \quad = \text{kurtosis} \tag{13}$$

Based on these features we decide if a region will be processed further or rejected as a false alarm. Due to the severely deteriorated image quality at the right and lower borders (end of scan-line), we want to reject regions centered very close to any image margin. Moreover, size and symmetry and compactness can be used to exclude other false alarms. All these properties are captured by the four region features. We use them in a Bayesian approach to make a decision λ_i regarding the selection or rejection of a candidate region i based on its likelihood of being caused by a moving object.

The *a posteriori* probability that the region i is part of a target, given its feature vector $\mathbf{s}_i$ is denoted $P(T_1|\mathbf{s}_i)$. The *a posteriori* probablilities $P(T_k|\mathbf{s}_i)$ are computed using Bayes' rule and the law of total probability as shown in equation 14.

$$P(T_k|\mathbf{s}_i) = \frac{p(\mathbf{s}_i|T_k) \cdot P(T_k)}{\sum_h p(\mathbf{s}_i|T_h) \cdot P(T_h)} \tag{14}$$

The probability densities $p(\mathbf{s}_i|T_k)$ are assumed to be multivariate Gaussian densities. Their parameters μ_k and Σ_k are computed as maximum-likelihood (ML) estimates from supervised training sequences. The *a priori* probabilities $P(T_k)$ are also obtained from the training data as the relative frequencies of targets.

$$\lambda_i = \begin{cases} \arg\max_k(P(T_k|\mathbf{s}_i)) & \text{if } \max_k(P(T_k|\mathbf{s}_i)) > \beta \\ \text{reject} & \text{else} \end{cases} \tag{15}$$

For each region, we make a decision λ_i (target or false alarm) based on the *a posteriori* probabilities and the confidence threshold β according to the decision rule (equation 15). If false alarms are to be avoided, β should be increased. Conversely, if missed detections have a high cost, β should be decreased. This decision depends on the cost of a false alarm compared to a missed detection. All regions not meeting the minimum confidence requirement β are unlikely to be moving objects. Hence, these are rejected and removed for further processing. The remaining regions are the final IMO part regions. Through the growing process they now include the adjacent boundaries of the corresponding objects. One key property of infra-red sensors is that targets or their parts (especially their *hot-spots* such as engine, exhaust) appear brighter than the background. Since the sequences are recorded from airborne sensors, we are never on the same plane as the targets. This assures that the front and rear parts of regular vehicles can not be hidden due to self-occlusion. We can distinguish four cases of object motion and their resulting FLIR inter-frame intensity changes:

object is	in front of object	behind object
appearing	becomes brighter	not observable
moving visible	becomes brighter	becomes darker
disappearing	not observable	becomes darker
moving occluded	not observable	not observable

We call the front of the IMO in the direction of its movement the head (and the other end is the tail). Consequently, an IMO region where the intensity increased after registration to the later image's frame of reference will contain the head. Conversely, a darkening region in the later image's frame of reference is a region that the tail just vacated (right behind the IMO). This information not only gives us a good indication of the location of certain object parts, but also allows us to obtain a rough estimate of the object's pose in the image.

In case of multiple moving objects, we have to find matching pairs of final regions. This requires clustering the detected regions into pairs consisting of a head and a tail region each. This also helps eliminate misdetected regions (false alarms) since it is very unlikely that there is a matching region to form a valid pair. To establish pairs, we assume that the distance from one object's front to its tail is smaller than from any of its parts to the contrary part of any another object. All possible pairs (combinations of a head and a tail) are considered and ranked by the distance measure $p_{i,j}$ as given by equation 16.

$$p_{i,j} = ||s_{i,1} - s_{j,1}|| \tag{16}$$

Starting from the closest match (lowest ranking), we now successively assign two regions to each pair. Since each region can be in only one pair, this accomplishes the desired clustering. Excess head or tail regions (false alarms) remain unpaired and are dropped at this stage.

Let us assume that a matching pair of a head and tail region has been found. We can approximate the object's pose in the image by the direction α' of the straight line from the centroid of the object's head to the tail. In our notation, $\alpha' = 0$ and $\alpha' = 90$ represent the directions straight up and straight to the right, respectively. For the typical airborne surveillance application, let us assume an elevated camera with a large focal distance ($f = \inf$ or parallel projection) looking forward at an object on a planar surface as depicted in Figure 4. This scene geometry can be used to link the image pose α' to the true object pose α. The true pose α is defined here as the direction of the vehicle's heading on the ground plane in respect to the observer.

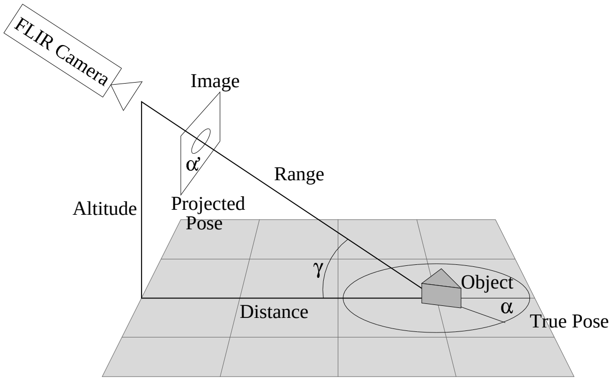

Figure 4. Scene geometry for planar surface and elevated observer.

$$\tan(\alpha) = \tan(\alpha') \cdot \sin(\gamma) \tag{17}$$

$$\text{distance} \cdot \tan(\gamma) = \text{altitude} \tag{18}$$

These equations can be rewritten to obtain a universal closed-form solution for α using the function atan, which is a generalized $\arctan$ function that computes an angle from a vertical and a horizontal component. Approximate knowledge of the camera's elevation above the ground plane (altitude) and its distance to the object on the ground allows us to compute α as follows:

$$\alpha = \text{atan}(\sin(\alpha') \cdot \sin(\text{atan}(\text{altitude}, \text{distance})), \cos(\alpha')) \tag{19}$$

Figure 5 shows a contour plot of the true object pose as a function of the distance-to-altitude ratio and the projected object pose in degrees. The true pose angles $\alpha = 0, \alpha = 90$, $\alpha = 180$, $\alpha = -90$ correspond to the vehicle pointing outbound, to the right, inbound, and to the left in respect to the observer. For distance/altitude $= 0$ the observer is exactly above the object and, hence, perceives the true pose ($\alpha' = \alpha$). With increasing distance at constant altitude the motion component in z-direction becomes less visible. In the limit, only strict left (-90 degrees) and right (90 degrees) movement can be perceived. This graph also shows that in high distance/altitude scenarios, small image pose estimation errors around 90 and -90 degrees result in large true pose estimation errors. From a long distance, it is hard to visually estimate if an object is moving in- or outbound.

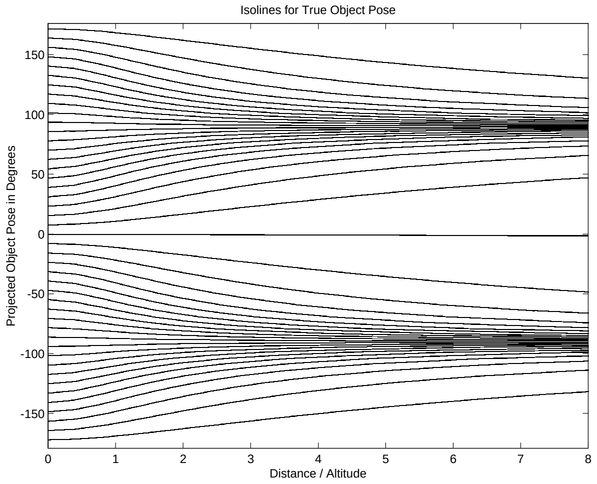

Figure 5. Contour plot of the true object pose as a function of the distance-to-altitude ratio and the projected object pose α'. Lines show locations of equal true object pose α.

4.3 Edge Extraction and Segmentation

As we have just seen, the IMO regions indicate the front or rear part of the moving object However, not all parts of the object are included into these two kinds of regions. Motion of homogeneously intense areas, for example, can not be observed. How can we find the entire object from the IMO part pairs? We have to resort to another feature domain, since pure motion information is not sufficient to solve this problem. Gray-level edges in the image can provide an indication of an object's boundaries. In the case of partial occlusion, we interpret the obstruction-object border as the object boundary. Independently from the motion detection, we extract the edges [9] from the reference frame based on the Sobel approximation of the derivative [19]. This can be done in parallel with the change detection. At this point we assume that the objects project to convex regions in the image with (eventually only partially) visible object boundaries in the direction of motion. Even though the convexity assumption may not hold for all objects, its violation leads to the detection of the convex part, which is usually sufficient. Since the IMO regions were grown, they now include the object boundaries. The edge in the head region corresponds to the IMO's front end, and the edge in the tail region to the rear end. Consequently, locations fulfilling both constraints, lying within an IMO region and being classified as an edge location, are the boundary locations of the object. Using the convexity assumption, the convex cover of the boundary regions constitutes the desired region-of-interest (ROI) containing the independently moving object (IMO).

5 Results

Figure 6 shows two FLIR frames (top row) and detected IMOs (bottom row) and Figure 7 depicts a spatio-temporal view of the entire sequence. During frames 1 to 30, a truck (the IMO) approaches the tank that sits in the center of the image. The elevated camera gradually comes closer. At frame 34, the camera was struck, resulting in an abrupt spatio-temporal discontinuity of the data. The camera fixates back at frame 38, but until the last frame 79, the sequence is unstable, with large inter-frame displacements. During this interval, the truck stops briefly and changes direction, driving toward the observer. The sequence demonstrates a mixture of continuous translational and abrupt, unsteady rotational camera motion.

Figure 8 illustrates the success of the frame-wise registration to stabilize the sequence. Various spatio-temporal slices through the entire sequence are shown before and after stabilization. In each slice the time progresses towards the right and the upright axis is the free spatial axis. The stabilization removed the small and short-term effects of the wobbling camera (the jagged lines in Figure 8(a) become smooth in (b)), as well as the continuous effect of the camera coming closer (the diverging lines in Figure 8(a) become parallel in (b)). It is interesting to note the merging of the bright traces of the sitting tank and the moving truck in Figures 8(c) and (d). In Figures 8(a) and (b), the IMO 'enters' the vertical slice late in the sequence and appears as the lower chip of the bright trace.

The middle row of Figure 6 shows the detected head (black) and regions behind the object's tail (gray). The objects' estimated direction of movement α' is indicated by the arrows. The final object segmentation obtained from edge and motion information is shown in the bottom row of Figure 6. In frames 15 and 72 the IMO is located accurately and successfully segmented from the stationary components of the scene. While our system reliably detects the IMO for most frames in the sequence, it fails in frame 34 when

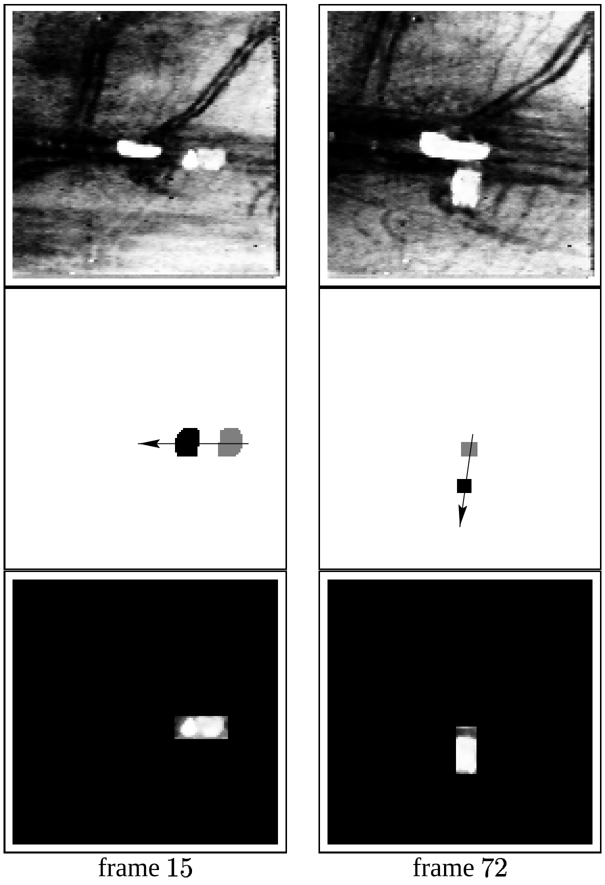
frame 15 frame 72

Figure 6. Detection and pose estimation results obtained with our system. FLIR frames 15 and 72 (top row) and the final object part pairs with pose arrows (middle row). The bottom row shows the corresponding ROIs.

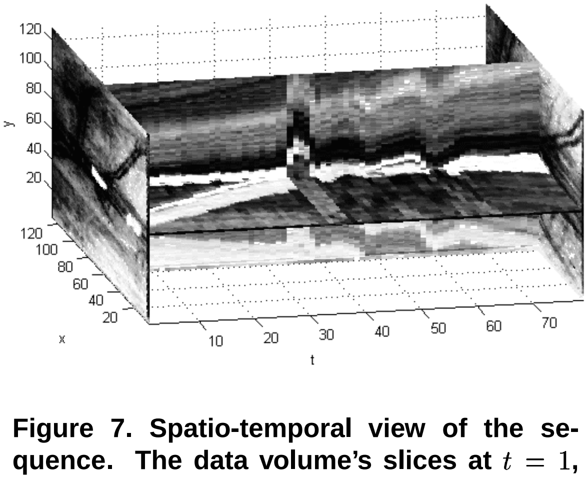

Figure 7. Spatio-temporal view of the sequence. The data volume's slices at $t = 1$, $t = 79$, $x = 60$, and $y = 60$ are shown.

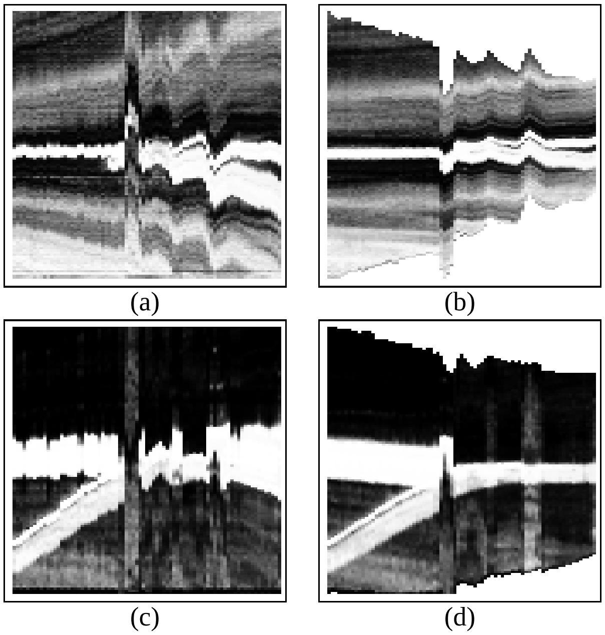

Figure 8. Effectiveness of ego-motion compensation. Vertical slices at $x = 60$ (upper row) and horizontal slices at $y = 60$ (lower row) before (left column) and after stabilization (right column).

the camera is struck heavily. This induces an abrupt and large displacement of the entire scene that can not be compensated with the registration module. Consequently, many scene features appear as candidate parts and no objects are detected.

Figure 9 shows several intermediate processing results for frame 15. In Figure 9(a) the original pixel-wise difference $\mathbf{\Delta I}_0$ of the current reference frame 15 and the previous frame 8 is shown. The difference depicted ranges from black (strong decrease) over gray (no change) to white (strong increase). After multi-scale registration, the observer-motion is removed and the errors in the difference image (Figure 9(b)) are due to IMOs. The initial regions for IMO parts (Figure 9(c)) are refined through morphological operations to obtain the candidate IMO part regions (Figure 9(e)). Candidate regions are selected (which in this case removes the false alarm regions at the margin) and paired. Edges (Figure 9(d)) within valid pair regions constitute the IMO boundaries as shown in Figure 9(f). From an overall perspective we obtain excellent results, especially when considering the quality of the FLIR sequence. The vehicle is detected and segmented successfully and accurately during most frames of the sequence. Figure 10 shows the obtained pose estimates based exclusively on image motion. The estimated pose changes from approximately -85 to -175 degrees. This correctly represents the trucks' left turn action.

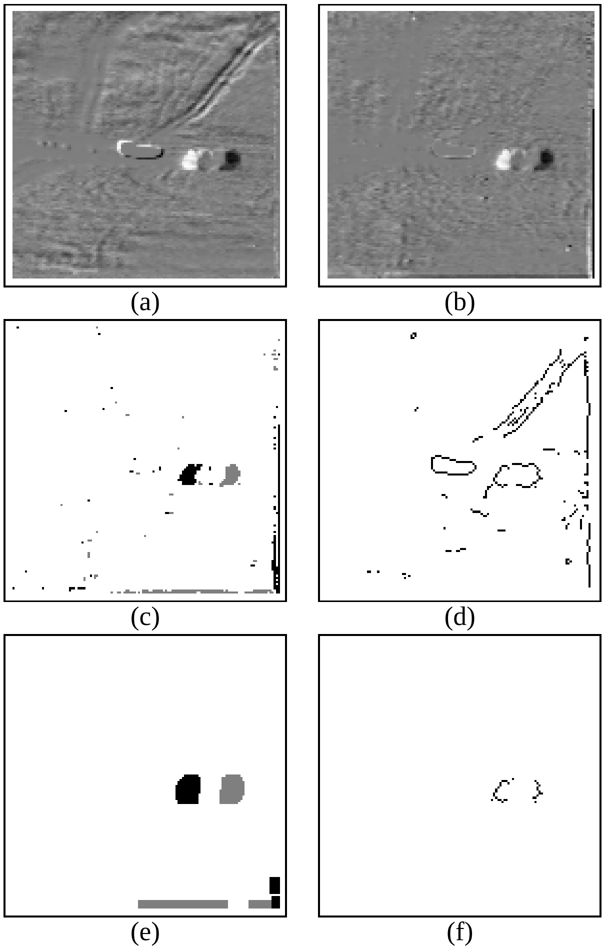

Figure 9. Steps in the processing at frame 15. (a) Original difference of frames 8 and 15. (b) After affine multi-scale registration. (c) Initial IMO parts. (d) Edge map. (e) Candidate IMO parts. (f) Final IMO boundary parts.

Results for two other complex and difficult sequences are shown in Figure 11. The top row shows a reference FLIR frame. The difference to the previous frame before registration is depicted in the middle row to illustrate overall motion effects. In the bottom row the candidate part pairs are shown. Final part pairs are overlaid with a pose indication arrow. The frame shown on the left (Figure 11(a)) contains two moving objects in a highly cluttered scene (road and trees), a tank moving rapidly to the right and another object moving towards the upper left of the image. Static ATR systems and even human observers may have difficulties detecting the targets in this image. Our dynamic system successfully detects both objects' head and tail and recovers

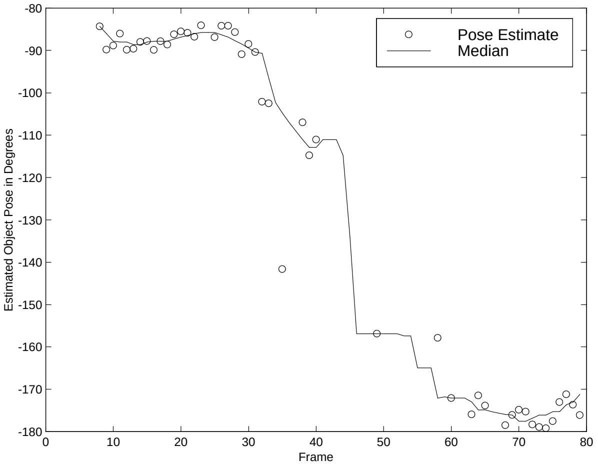

Figure 10. Pose development during turning action of object.

their poses. A false alarm tail in the upper right corner is also detected at first, but is rejected later, since it can not be paired. The sequence on the right-hand side (Figure 11(b)) shows a tank moving across an unobstructed field towards the observer. The system successfully detects the heated right wheels and gives a good estimate of the tank pose. It can also be seen that the hot exhaust fumes induce a false alarm by appearing to be a head part. However, the fumes do not follow rigid motion. A heat edge appears on the fumes' front but, due to the gradual dilution and cooling, no corresponding tail exists. Consequently, the falsely detected head remains unpaired and is rejected.

6 Conclusion and Future Work

In this paper we propose a novel motion-based object detection system for FLIR sequences. Motion is a very strong cue, especially in highly cluttered environments, that has not been considered sufficiently in previous work. The shortcomings of the sensor and requirements for real-time processing induce the need for a fast and robust system. Our detection system adapts well-known robust techniques from the visible to the FLIR domain. An iterative approach, used for the most time-costly operation, image registration, assures a scalable algorithm complexity. We propose a new methodology to link the new dynamic information and static cues, such as object pose, enabling the construction of more redundant and fault-tolerant systems. Our algorithm has been implemented, and results on difficult, real sequences are presented.

In future work we want to integrate the presented dynamic scene analysis system with existing static image ATR

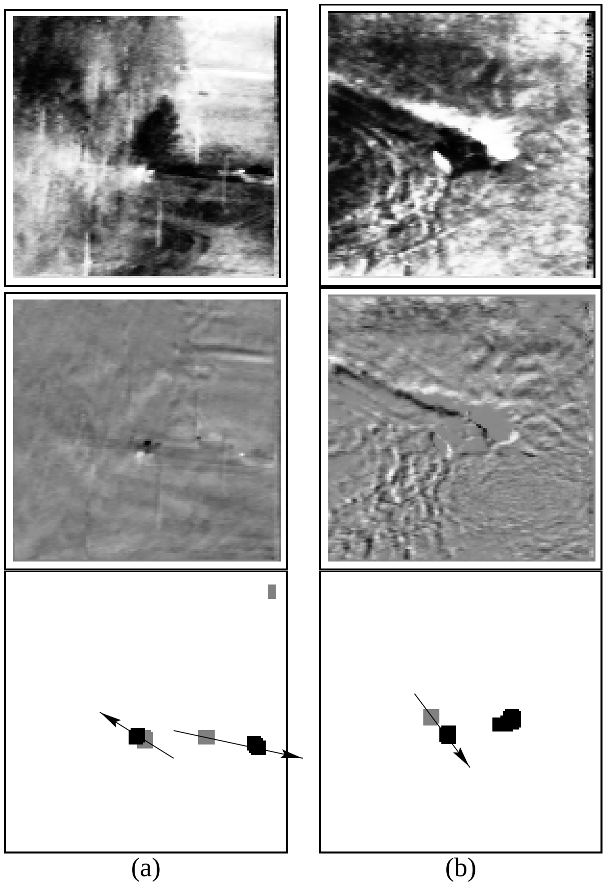
(a) (b)

Figure 11. More results on complex and difficult sequences. Reference frames (top row), differences before alignment (middle) and candidate parts with superimposed pose estimates for final pairs (bottom row).

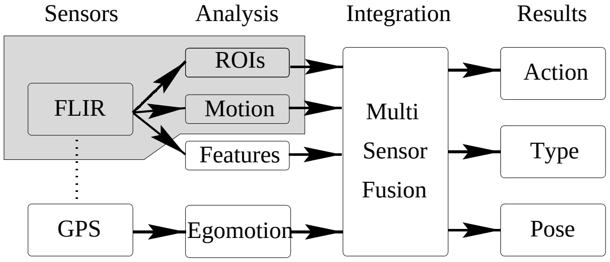

Figure 12. Overview of our proposed future target detection and recognition system.

systems (such as [15]) into a comprehensive system (Figure 12). The shaded box highlights the parts of the system described in this paper. Together with cues from other modules, it can be used in a Bayesian sensor fusion paradigm to improve detection accuracy and reduce false alarms. In such a fusion stage detection, recognition and pose results from various cues such as motion, target shape, size or parts can be integrated using a Bayesian meta-classifier. The different paradigms can be used to mutually verify their results and synergetically improve performance. Compared to existing systems, dynamic scene analysis enables the inclusion of target action recognition. This action recognition could enable multi-frame analysis results such as object starts and stops and changes in acceleration and direction to be extracted automatically. Target action knowledge provides a high-level abstraction based on motion analysis that has great potential to extend and enhance existing systems.

Acknowledgement

We would like to thank Debi Paxton for her assistance in editing this paper.

References

[1] G. Adiv. Determining three-dimensional motion and structure from optical flow generated by several moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(4):384–400, 1985.

[2] J. K. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images: A review. *Proceedings IEEE*, 76:917–935, August 1989.

[3] R. Battiti, E. Amaldi, and C. Koch. Computing optical flow across multiple scales: an adaptive coarse-to-fine strategy. *International Journal of Computer Vision*, 6(2):133–145, 1991.

[4] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model–based motion estimation. In *Proceedings European Conference on Computer Vision, Berlin, Germany*, volume 588 of *LNCS*, pages 237–252. Springer, May 1992.

[5] B. Bhanu, D. E. Dudgeon, E. G. Zelnio, A. Rosenfeld, D. Casasent, and I. S. Reed. Introduction to the special issue on automatic target detection and recognition. *IEEE Transactions on Image Processing*, 6(1):1–6, January 1997.

[6] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, January 1996.

[7] W. Burger and B. Bhanu. Estimating 3-D egomotion from perspective image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(11):1040–1058, November 1990.

[8] P. Burt and E. H. Adelson. The laplacian pyramid as a compact image code. *IEEE Transactions on Communication*, COM-31(4):532–540, April 1983.

[9] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, November 1986.

[10] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, Reading, MA, 1990.

[11] B. K. P. Horn and B. G. Schunk. Determing optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[12] P. J. Huber. *Robust Statistics*. Wiley, New York, 1981.

[13] M. Irani and P. Anandan. A unified approach to moving object detection in 2D and 3D scences. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(6):577–589, June 1998.

[14] C. H. Morimoto, D. Dementhon, L. S. Davis, R. Chellappa, and R. Nelson. Detection of independently moving objects in passive video. In *Proceedings of Intelligent Vehicles Workshop, Detroit, MI*, pages 270–275, September 1995.

[15] D. Nair and J. K. Aggarwal. A focused target segmentation paradigm. In *Fourth European Conference on Computer Vision*, volume 1, pages 579–588, Cambridge, UK, April 1996.

[16] N. Nandhakumar and J. K. Aggarwal. Multisensory computer vision. *Advances in Computers*, 34:60, 1992.

[17] J. A. Ratches, C. P. Walters, R. G. Buser, and B. D. Guenther. Aided and automatic target recognition based upon sensory inputs from image forming systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(9):1004–1019, September 1997.

[18] S. Rogers, J. Colombi, C. Martin, J. Gainey, K. Fielding, T. Burns, D. Ruck, M. Kabrisky, and M. Oxley. Neural networks for automatic target recognition. *Neural Networks*, 8(7–8):1153–1184, 1995.

[19] I. E. Sobel. *Camera Models and Machine Perception*. PhD thesis, Stanford Univ., 1970.

[20] P. H. S. Torr and A. Zisserman. Concerning bayesian motion segmentation, model averaging, matching and the trifocal tensor. In *Fifth European Conference on Computer Vision*, Freiburg, Germany, June 1998.

[21] B. A. Wandell. *Foundations of Vision*. Sinauer Associates, Inc., 1995.

[22] J. Y. A. Wang and E. H. Adelson. Spatio-temporal segmentation of video data. In *Proceedings of SPIE on Image and Video Processing II, vol 2182, San Jose*, pages 120–131, February 1994.

[23] A. H. Wertheim. Motion perception during self-motion: The direct versus the inferential controversy revisited. *Behavioral and Brain Sciences*, 17:293–355, 1994.

[24] Q. Wu. A correlation-relaxation-labeling framework for computing optical flow – template matching from a new perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(9):843–853, September 1995.