

An optical tracker for the maritime environment

Asheer K. Bachoo^{a,b}, Francois le Roux^a and Fred Nicolls^b

^aOptronic Sensor Systems, CSIR, Pretoria, South Africa

^aElectrical Engineering Department, University of Cape Town, South Africa

ABSTRACT

Optical (visual) tracking is an important research area in computer vision with a wide range of useful and critical applications in defence and industry. The tracking of targets that pose a threat or potential threat to a country's assets and resources is a critical component in defence and security. In order to complement radar sensing applications, an optical tracker provides additional functions such as target detection, target identification and intent detection at the visual level. A tracker for the maritime environment is an optical system that performs the automatic tracking of an above water target. Ideally, a track of the target is required for as long as is possible. Some examples of targets include boats, yachts, ships, jet-skis and aircraft. A number of factors mitigate the performance of such a system — change in target appearance, target occlusions, platform vibration and scintillation in the atmosphere are some common examples. We present the implementation of a first-generation system that is robust to platform vibration, target appearance changes and short-term occlusions. The optical tracker is developed using a particle filter and an appearance model that is updated online. The system achieves real-time tracking through the use of non-specialized computer hardware. Promising results are presented for a number of real-world videos captured during field trials.

Keywords: visual tracking, particle filter, maritime surveillance

1. INTRODUCTION AND BACKGROUND

Modern naval forces are expected to operate in a wide range of operational conditions and against an extensive range of potential threats. The threats above the surface can vary from large ships to small rubber boats and canoes, airborne threats as well as land-based forces in littoral operations. In a number of these scenarios, radar tracking must be supplemented by optical tracking. An optical tracker for the maritime environment must be able to cope with an almost limitless set of scenarios. Aspects that can influence the performance of an optical tracker include the target occlusion due to swells, the target's wake, water splashes, white caps on waves and sensor motion. The combination of these factors places the implementation of an optical tracker in the maritime environment firmly in the domain of a "difficult problem". In this paper, visual tracking of maritime targets is considered.

One can distinguish between two important components in a visual tracking system: *target description* and *target state estimation*. Target description refers to the way that one extracts and describes information about the object of interest in the video. As such, we are concerned with the image processing component. There are several ways to describe a target: some examples include raw pixel intensities, grey scale (or colour) histograms¹, edge gradients², silhouettes², active contours³ or appearance models^{4,5}. A target may also be represented by a single point (centroid). The target description is constructed when tracking is initiated and is usually referred to as the *target model*. Thereafter, likely observations called *target candidates* are extracted and compared to the target model in the state estimation stage. Generally, the target extraction process also requires a transformation component that addresses changes in scale and rotation of the object.

State estimation in target tracking refers to the way in which a variable of interest is estimated over time. In the simplest case, the state is the target position in the video sequences. Statistical approaches generally use observation and state models to estimate the state of the object in each video frame. In most instances, the target description is used to define the observation model — it can be referred to as the target likelihood function

Further author information: (Send correspondence to A.K.B)

A.K.B: E-mail: abachoo@csir.co.za, Telephone: +27 12 841 3787

and provides a measure of how closely a candidate target matches the target model. The state model defines how the state evolves in time. For example, if we are tracking the object position, it will describe the target motion dynamics. A commonly used technique is statistical filtering such as the methods of the Kalman and particle filters (discussed later)⁶. These are statistical approaches that use target observations at each time t to minimize the uncertainty of the target state distributions. With these methods, probability density functions for the observation and state models must be specified. These methods are reliant to the form and representation of the probability density functions. Other approaches in the literature are forms of data association: they consider the most likely way to associate an observation (candidate target) to the target model. These include nearest neighbour, gating and multiple hypothesis tracking⁷. In statistical approaches, the likelihood functions for matching models to candidates are dependent on the target description model. For example, in point tracking, points are compared using the Euclidean distance function.

Target tracking in the maritime environment is frequently performed using infrared imaging technologies. A difficulty associated with this is the lack of visual cues for discriminating targets. This is an important requirement for distinguishing between friend or foe. Additionally, the visual cues provide useful information for addressing appearance variations in a target and complex scenes with interacting targets. Szpak and Tapamo⁸ propose background (sea) modelling and level set segmentation for tracking ships in maritime scenes. A limitation of the approach is the requirement for a stationary camera so that statistics can be computed for each pixel. In spite of this shortcoming, promising results are presented by Szpak and Tapamo for several different types of watercraft. A mean shift tracker for visual tracking at sea is presented by Bibby and Reid⁹. A novelty in their approach is background and foreground modelling for improved target localization. However, no conclusive or quantified results are presented for their tracker. Wei *et al.* describe a surveillance system for ships at sea¹⁰. Horizontal line detection is used to detect the horizon. This is then used to locate the sea region, which is modelled using a regression function. The regression model is thereafter used to detect ships by thresholding pixels. Tracking is achieved using nearest neighbour association and histogram matching for the target signature.

The work presented in this paper describes a first generation tracker for the maritime environment that processes visual data. In the next section, the problem is described in more detail and we outline the initial assumptions and problem constraints. Sections 3 and 4 describe the optical tracker and present the experimental results respectively. This is followed by a short discussion on future work and concluding remarks.

2. PROBLEM DESCRIPTION AND INITIAL ASSUMPTIONS

This paper considers high zoom tracking of targets on the ocean. A number of problems, as shown in Figure 1, are present in the maritime environment under typical real-world conditions:

1. **Low contrast:** In high zoom tracking, optical performance over long range is limited by the operating conditions, lens design and the camera sensor sensitivity. Atmospheric perturbations result in scintillation in the video and poor contrast; poor lens design can also lead to low contrast and soft focus in the acquired images.
2. **Camera motion:** A surveillance system with a high zoom lens is prone to severe camera shake if the lens undergoes an uncontrolled motion. For example, wind or platform vibration will cause camera shake. One common approach for solving this problem includes the use of an inertial measurement unit for stabilizing the video. Video stabilization can also be done by image processing in software.
3. **Dynamic appearances:** The ocean is a dynamic object with spatio-temporal variations in pixel intensity; targets of interest also have dynamic appearances. In monocular imaging, depth information is generally absent and all that is available is a 2-dimensional (2D) projection of the 3-dimensional world (3D). Thus, when a 3D object changes appearance as a result of changes in its 6 degrees-of-freedom position in space, the 2D model must be able to capture some of these appearance dynamics. In other words, an appearance model is required to robustly track the target. This is an extremely difficult problem.
4. **Watercraft wake:** Objects moving on the surface of the ocean usually generate a wake. In certain scenarios and under certain viewing angles, the wake visually obstructs the target visibility. For example, tracking of small targets can be inhibited by the wake.

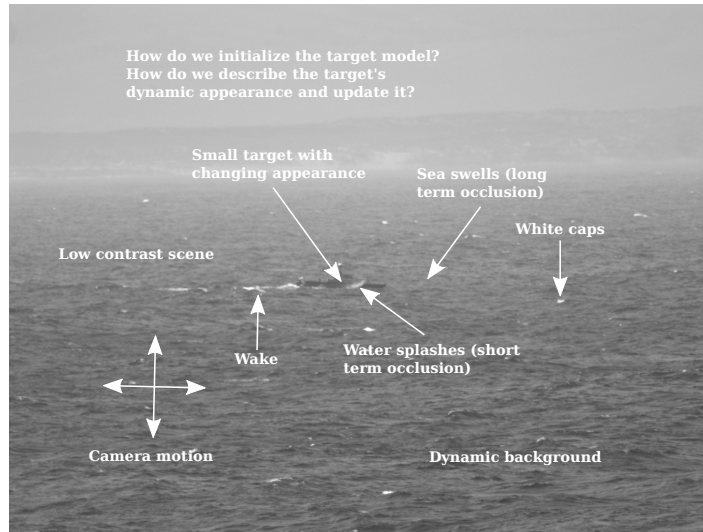


Figure 1. The tracking problem in a maritime environment.

5. **Occlusions:** Occlusions occur when targets of interest intersect or move behind other objects. A common occurrence is the short-term occlusions caused by waves breaking on a water craft. These can be full or partial occlusions. A long-term occlusion apparent during inclement weather is the disappearance of a watercraft in sea swells.
6. **Track initialization:** The initialization of a track is an especially difficult problem when the aforementioned factors are present in a scene. Of particular importance is the initialization of the target model and its subsequent update. A requirement in this regard is knowledge of the background and foreground. Thus, a suitable approach would involve image segmentation.

In this paper, a few of the problems described above are addressed in the first generation tracker. Some assumptions are first outlined:

1. *The target undergoes appearance changes. However, its shape and visibility remain the same i.e. there is no significant change in the target's orientation relative to the camera (except for target rotation around a point in the image) and the object is rigid.*
2. *Camera motion is accommodated in the state (motion) model.*

The assumptions above have resulted in the first generation tracker described in the next section. Assumption 1 leads to the appearance model being represented by a template update strategy. Assumption 2 enables the particle filter and its associated motion model to be robust to camera shake. The proposed method can also accommodate low contrast, white caps and occlusions through the use of a robust error function for the observation model.

3. THE OPTICAL TRACKER

The system described in this section can track multiple targets and perform occlusion detection. Currently, some of the system parameters are fixed. The parameter adaptation will be examined in future work. In this paper, however, only single target tracking is evaluated.

3.1 Target description

In a video frame, a target is represented by an image patch of pixel intensities. A candidate image patch is described spatially through a set of pose parameters (position, scale and rotation) applied to a unit square at the origin. When the tracker is initialized, a template consisting of the patch intensity values and its pose parameters are stored in the state model. This template is normalized to a size of 31×31 pixels. For every subsequent frame at time t , the template is matched to candidate patches using a robust error function (discussed in the next section). The best match denotes a possible location of the actual target at time t .

3.2 State estimation

In most tracking problems the posterior distribution is non-Gaussian/non-linear. The particle filter^{6,11} is a Monte Carlo method that makes use of recursive Bayesian estimation to estimate a posterior density with non-linear/non-Gaussian form. The posterior density is the probability that a set of measurements has resulted in the current state. It is represented by a set of weighted particles, where each particle is a multidimensional state, and the estimated state is obtained by a weighted sum of the particles (or the particle with the highest weight). The particle filter tracks a patch of image pixels undergoing an affine transform. This is similar to Jepson *et al.*⁴ except that we use a rectangular patch of pixels rather than an ellipse-shaped one. The state vector is

$$\begin{bmatrix} \mathbf{x}_t \\ \mathbf{s}_t \\ \mathbf{v}_t \\ r_t \end{bmatrix}$$

where \mathbf{x} , \mathbf{s} and \mathbf{v} are the component position, scale and velocity of the target respectively and t is in units of $\frac{1}{20}$ seconds. The symbol r is the rotation of the target around \mathbf{x} . The dynamic state model is

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \mathbf{v}_{t-1} + \boldsymbol{\omega}_{t-1} \tag{1}$$

$$\mathbf{s}_t = \mathbf{s}_{t-1} + \mathcal{N}(0, 0.01) \tag{2}$$

$$\mathbf{v}_t = \mathbf{x}_t - \mathbf{x}_{t-1} \tag{3}$$

$$r_t = r_{t-1} + \mathcal{N}(0, 0.02). \tag{4}$$

The noise process for Equations 2 and 4 are assumed to be the normal Gaussian and the standard deviations shown above are in pixels. The vector $\boldsymbol{\omega}$ is drawn from a zero mean Gaussian with independent variables — the Gaussian’s covariance matrix is adapted so that its diagonal elements are proportional to the magnitudes of the component velocities. The minimum value for a diagonal element is provided before tracking. State variables at time $t - 1$ are stored in memory during tracking. In our experiments, the above equations work well since the velocity is always updated at every frame. A history of velocities is maintained for the last 25 frames and this is used to adapt the variance of the Gaussian in Equation 1 to match the target motion. Thus, greater pixel displacements will increase the variance of the noise function. The state variables \mathbf{x} , \mathbf{s} , r are used to describe the image patch. Particles are sampled using the transitional prior⁶. If an occlusion is detected (discussed in the next paragraph), the variance of the Gaussian in Equation 1 is set to a large application specific value. Occlusion detection implies that target is being blocked by another object or *camera shake* has occurred (since the camera motion is much larger than what the model predicts). Thus, the greater variance scatters the particles and improves target detection. An adaptive number of particles will be examined in future work.

The observation model is a robust error function similar to that used by Zhou *et al.*¹², which compares a particle’s patch to an image template. This template is created when a track is initialized by the user. At each time t , an image patch is extracted using the state vector of a particle. The state estimate is selected as the particle that most likely matches the template (i.e. it has the highest likelihood (weight)). This template is updated online using drift correction¹³ (discussed later). A comparison function is defined as follows:

$$\rho_i(\epsilon) = \begin{cases} 1.0 & \text{if } \epsilon < T \\ 0.0 & \text{otherwise} \end{cases} \tag{5}$$



(a) Video 1: Patrol boat in rough seas with camera shake and occlusions caused by water. There are white caps and the video has low contrast.

(b) Video 2: Small target in a low contrast scene. The video contains a single long term occlusion (caused by a lighthouse).

(c) Video 3: Patrol boat in rough seas with severe occlusions, camera shake and extremely low target visibility at times.

Figure 2. The 3 video sequences used for the experiments.

The pixel number is represented by i . In the above equation, $\epsilon = \frac{|x_i - y_i|}{d}$, where x is the candidate pixel and y the template pixel. The parameter d is a normalizing factor and also controls the effective width of the error function. It can be adapted to perform appropriately in the presence of different levels of image contrast. For all pixels in the target template, $\rho(\epsilon)$ is computed and then summed and normalized by dividing the sum by the total number of pixels processed:

$$\gamma = \frac{1}{N} \sum_{i=1}^N \rho_i \quad (6)$$

where N is the number of pixels in the template. Thus, the patch likelihood is determined by γ . The threshold T is currently determined empirically (it is set to 1.0). The γ function effectively detects occlusion if its output falls below a threshold otherwise a target is detected. When occlusion (or no target) is detected, the history of velocities is used to predict the target's position for tracking through the occlusion and for evolving the state vector. The parameter d is set to 0.05 and the particle filter is created with 400 particles.

3.3 Appearance model

An appearance model describes the variations in a target's pixel intensities as it evolves in time. Most appearance models are online i.e. they update themselves in a sequential fashion as new data becomes available. A simple approach to appearance modelling is that of template updating when using template-based tracking¹³. The main idea in this approach is to update the template at time t using the latest target pixel intensities. A well-cited approach to the template update problem, and one that we use, is that of Matthews *et al.*¹³. To prevent the drift problem during template update, Matthews proposes aligning a candidate template with the initial template using gradient descent optimization. The candidate template becomes the new template if its alignment error is bounded by a predetermined value. Note that the initial template i.e. the template at $t = 0$, is always stored in memory and used for the candidate template alignment. Thus, this method is only suitable for tracking if there are no extensive variations in the target visibility. This means that it can handle slight changes in shape and it is assumed that no new object parts come into view. Hence, lighting and other gradual changes are represented by the technique. Schreiber¹⁴ provides an extension to Matthews' work. The next section presents the experimental results and discussion.

4. EXPERIMENTAL ANALYSIS AND DISCUSSION

Test data for the maritime tracker was captured in Simonstown on the South African coast in October 2010. Three data sequences were used for the experimental analysis. These sequences were captured in typical real-world scenarios using a pan-and-tilt unit with a 300mm lens focal length. Due to the small sensor size, the

effective focal length of the lens was 450mm (caused by the approximate $\times 1.5$ crop factor). A number of elements described in Section 2 are visible in the test data. Figure 2 shows and describes the three test sequences (Video 1 to Video 3). The video data was processed on a Dell D830 laptop. Real-time execution of the tracker is achieved using a desktop computer with quad-core central processing units and a high end graphics processing unit. The parameter settings for the tracker have been presented in Section 3.2. Tracking is initialized by the user through the selection of a target region of interest. This region of interest mainly consists of target pixels i.e. outliers are minimized. If the target cannot be initialized appropriately, frames are skipped until initialization can take place. This approach is analogous to real-world situations whereby the operator must "lock" onto a target before tracking commences. The detection parameter (γ) is set to 0.8 and the template is updated if the affine parameters are refined to achieve a γ value of greater than 0.95.

A high performance Prosilica video camera was used to capture 8-bit grey scale images of resolution 1360×1024 at 20 frames-per-second. The video sequences were then manually processed by a user to generate ground truth data i.e. the actual target (x, y) position in the frame. To accommodate the bias that may be inherent in the ground truth data as a result of the operator's personal preferences during processing, we report on both the mean and the standard deviation of the target position tracking errors. Thus, a mean error with a low standard deviation will imply that the operator's choice of the target position is different from that of the tracker or the track initialization process. The tracking performance is evaluated in terms of the algorithm's ability to accurately estimate the target position in each video frame. Performance metrics were derived using ideas from Senior *et al.*¹⁵. The following metrics are used (with symbols shown in brackets):

1. **Mean absolute error:** The mean of the error over all video frames is computed. The error for a single measurement in a video frame is defined as

$$error = |x_g - x_e|, \quad (7)$$

where x_g is the ground truth x -coordinate and x_e is the x -coordinate estimated by the tracker. The error is defined similarly for the y -coordinate. Errors are measured in units of pixels.

2. **Standard deviation of the error:** This is the standard deviation over all the errors for each video sequence.
3. **Track length (l):** This is the longest consecutive sequence of frames that is tracked. It is expressed as a percentage:

$$l = \frac{\text{length of consecutive track}}{n} \times 100 \quad (8)$$

where n is the total number of frames in the video sequence. A track length of 100 implies uninterrupted tracking. This metric must be assessed in conjunction with the mean errors. An estimated track breaks when the x or y error for a frame is greater than 100 pixels. This large threshold was selected so as to accommodate the lens shake that is possible during tracking and to allow the tracker time to converge to a solution. When a track breaks, the user reinitializes the template and the tracker. If initialization is not possible, frames are skipped until it is successful (as discussed earlier).

4. **Number of re-initializations (ϕ):** This is the number of times the tracker is re-initialized, per 100 frames, so that it can complete tracking a target in a video sequence.
5. **Track ratio (r):** This measures, as a percentage, how often the tracker believes it is tracking the target. Thus, it is a measure of false and true positives. r should be examined in conjunction with the other metrics since we do not have knowledge of the non-target labels. r is not meaningful if the mean error is very large. This measure should decrease in very noisy video with a large number of occlusions and poor video quality. It is computed as

$$r = \frac{\# \text{Tracker believes it is tracking the target}}{n} \times 100, \quad (9)$$

where $\#$ is "number of times".

Table 1. Summary of tracking results.

Video	Global Metrics				Target		No Target	
	n	l (%)	ϕ	r (%)	x_{error}	y_{error}	x_{error}	y_{error}
1	683	83	0.15	57	6.68±7.95	2.47±1.92	17.42±20.48	23.75±21.10
2	1457	83	0.06	73	5.53±5.00	2.27±1.60	14.72±16.35	2.92±1.89
3	2278	16	0.57	53	5.82±4.84	2.58±1.81	31.39±28.05	4.62±5.31

The mean and standard deviation are computed over two types of errors for all video frames: *target* and *no target*. *Target* means that the tracker believes it is tracking the target; *no target* refers to when the tracker believes it is not tracking the target. Thus, we want a low mean error when tracking the target and higher mean errors when not tracking the target. The mean and standard deviation are displayed as $\mu \pm \sigma$, where μ is the mean and σ the standard deviation.

Table 1 summarizes the tracking results. The video sequences are discussed in the next sections. An immediately noticeable result is the very high mean errors when not tracking the target; the mean errors for tracking the target are relatively low. This is a good result since it implies tracking is taking place correctly when the tracker believes the target has been detected. The mean errors and standard deviations in the horizontal direction are much larger than the y -direction and this is caused by the camera panning horizontally. The track ratios achieved are over 50%. This implies that the tracker locates the target almost every second frame. The track lengths achieved are over 80% except for the case of Video 3 (which is under fairly extreme conditions). By looking at the largest mean error and standard deviation, we can also safely assume that the tracking error is less than 2.5% when taking the image dimensions into consideration. These are very promising results considering the extreme noise effects present in the videos (Section 2) and demonstrate the robustness of the system.

4.1 Sequence 1

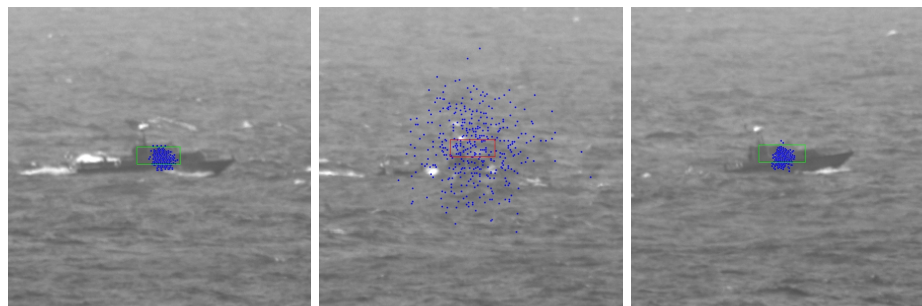
The first test sequence (Figure 2(a)) shows a patrol boat on the ocean. The camera has a moderate amount of movement in the x and y directions and there are occlusions caused by water splashing onto the boat and moderate sea swells. The camera also pans to the right to keep the target in its field of view. The minimum variance for the motion model is set to 4.0 pixels and the variance for occlusion handling is set to 20.0 pixels. The tracker is able to maintain a track length of 83% and a track ratio of 57%. The mean error is 6.68 pixels. Occlusion detection for the water splashes is handled well by the tracker (Figure 3).

4.2 Sequence 2

Video 2 (Figure 2(b)) contains a low contrast target with minimal camera shake and the camera pans to the left to keep the target in view. A light house is present and causes a long term occlusion of the target. The tracker tracks through the occlusion successfully (Figure 4). The minimum variance for the motion model is set to 4.0 pixels and the variance for occlusion handling is set to 20.0 pixels. Given the fairly stable target track, the tracker achieves a track length of 83% and a track ratio of 73%.

4.3 Sequence 3

The third test video (Figure 2(c)) is similar to the first one. However, there is extremely low target contrast at times and excessive camera panning and camera shake. The patrol boat sometimes disappears completely in sea swells and there is also significant target blur when the camera pans. The minimum variance for the motion model is set to 10.0 pixels and the variance for occlusion handling is set to 30.0 pixels. Figure 5 presents the frame to frame displacements of the target computed using the ground truth for the first 500 frames. The sharp spikes in the plot are caused by sudden camera movements. The displacements (of up to almost 150 pixels) are much larger in the x direction and this is indicative of the camera panning. The vertical displacement is also quite moderate but shows sharp spikes from frame 400 onwards. Figure 5 highlights the fact that during very fast camera panning, the tracker will most likely fail. In this scenario, panning should be automated so that tracking is consistent and robust and the target motion model should also incorporate the camera motion. Video



(a) Tracking of a patrol boat. The particles are shown in blue.

(b) Detection of an occlusion caused by a water splash. Note the scattering of the particles.

(c) Continuation of tracking after the water splash.

Figure 3. Detection of water splashes in Video 1.



(a) Before the occlusion.

(b) Occlusion detection.

(c) Occlusion detection: The prediction is still accurate.

(d) Track continuation after the occlusion.

Figure 4. Occlusion handling in Video 2.

stabilization will also improve tracking. The sea swells that cause the patrol boat to disappear for a few frames are shown in Figure 6. The tracker treats sea swells similarly to the detection of water splashes in Figure 3.

The results for Video 3 highlight the factors mentioned above. The track length is much lower than the previous test sequences and there is a much higher number of reinitializations required for consistent tracking. This is caused by the harsh camera panning and shake. However, the track ratio is a respectable 53%. The mean error when tracking is also low; the mean error is very large when not tracking the target.

5. FUTURE WORK

The optical tracker presented in this paper has shown very promising results. However, there are further research avenues that must be explored. To this aim, several important ideas are proposed. The first idea will extend the appearance model so that greater appearance variations can be described. Statistical modelling for the target appearance is likely to be more robust than template based matching. Online appearance models for maritime tracking is envisaged in future work. Secondly, particle resampling methods will be examined for improved target detection. A final avenue for future work is an evaluation of the tracker using a standard dataset for comparison to existing systems

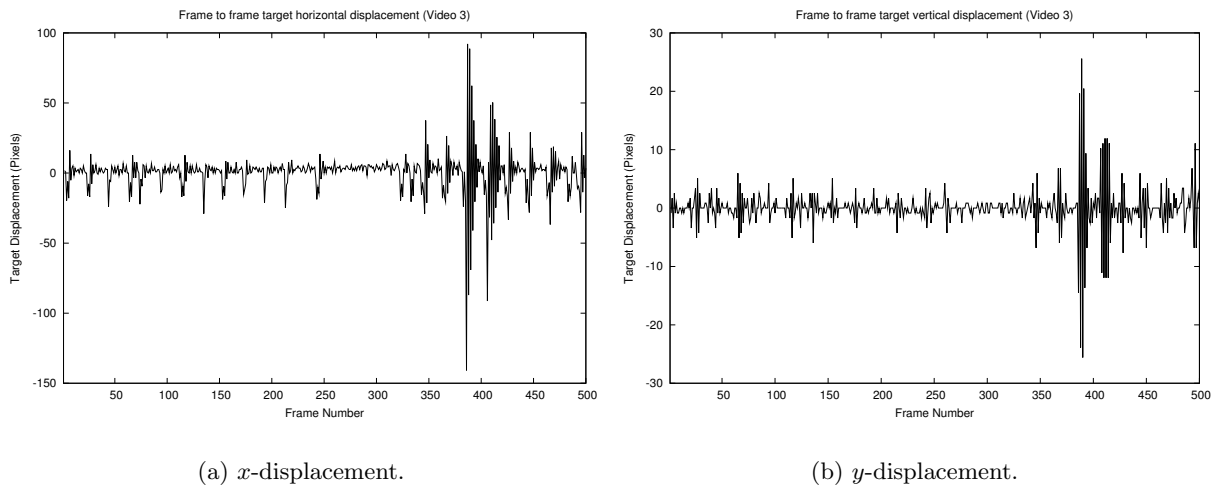


Figure 5. Target displacement for Video 3.

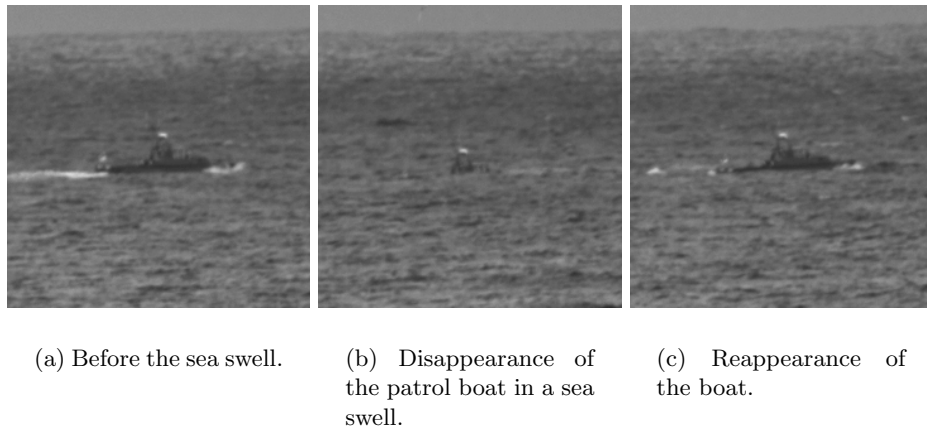


Figure 6. Severe sea swells in Video 3.

6. CONCLUSIONS

The experimental results presented in this paper are very promising for optical tracking in the maritime environment. The system can track a target in high resolution video with an error of less than 2.5% in the presence of low contrast, white caps, camera motion and occlusions. The test sequences used are from real-world scenarios and demonstrate the robustness of the system.

REFERENCES

- [1] Gonzalez, R. and Woods, R., [*Digital Image Processing*], Addison-Wesley Publishing Company (2002).
- [2] Sonka, M. and Hlavac, V. and Boyle, R., [*Image Processing: Analysis and Machine Vision*], PWS Publishing Company (1999).
- [3] Blake, A. and Isard, M., [*Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion*], Springer-Verlag New York, New Jersey, 1st ed. (1998).
- [4] Jepson, A., Fleet, D., and El Maraghi, T., "Robust online appearance models for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25**, 1296–1311 (October 2003).

- [5] Cootes, T., Edwards, G., and Taylor, C., “Active appearance models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(6), 681–685 (2001).
- [6] Arulampalam, S., Maskell, S., Gordon, N., and Clapp, T., “A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking,” *IEEE Transactions on Signal Processing* **50**(2), 174–188 (2001).
- [7] Yilmaz, A., Javed, O., and Shah, M., “Object tracking: A survey,” *ACM Computing Survey* **38**(4), 13+ (2006).
- [8] Szpak, Z. L. and Tapamo, J. R., “Maritime surveillance: Tracking ships inside a dynamic background using a fast level-set,” *Expert Systems with Applications* **38**(6), 6669 – 6680 (2011).
- [9] Bibby, C. and Reid, I. D., “Visual tracking at sea,” in [*ICRA*], 1841–1846 (2005).
- [10] Wei, H., Nguyen, H., Ramu, P., Raju, C., Liu, X., and Yadegar, J., “Automated intelligent video surveillance system for ships,” in [*Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*], *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference* **7306** (May 2009).
- [11] Ristic, B., Arulampalam, S., and Gordon, N., [*Beyond the Kalman Filter: Particle Filters for Tracking Applications*], Artech House (2004).
- [12] Zhou, S., Chellappa, R., and Moghaddam, B., “Visual tracking and recognition using appearance-adaptive models in particle filters,” *IEEE Transactions on Image Processing* **13**, 1434–1456 (2004).
- [13] Matthews, I., Ishikawa, T., and Baker, S., “The template update problem,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26**, 810 – 815 (June 2004).
- [14] Schreiber, D., “Robust template tracking with drift correction,” *Pattern Recognition Letters* **28**, 1483–1491 (September 2007).
- [15] Senior, A., Hampapur, A., Tian, Y., Brown, L., Pankanti, S., and Bolle, R., “Appearance models for occlusion handling,” in [*2nd IEEE Workshop on Performance Evaluation of Tracking and Surveillance*], (2001).