Foreground segmentation in atmospheric turbulence degraded video sequences to aid in background stabilization

Philip E. Robinson[a,*], **André L. Nel**[b]
[a]University of Johannesburg, Faculty of Engineering and the Built Environment, Department of Electrical and Electronic Engineering Science, Corner Kingsway and University Road, Johannesburg, South Africa, 2006
[b]University of Johannesburg, Faculty of Engineering and the Built Environment, Department of Mechanical Engineering, Corner Kingsway and University Road, Johannesburg, South Africa, 2006

Abstract. Video sequences captured over a long range through the turbulent atmosphere contain some degree of atmospheric turbulence degradation (ATD). Stabilization of the geometric distortions present in video sequences containing ATD and containing objects undergoing real motion is a challenging task. This is due to the difficulty of discriminating what visible motion is real motion and what is caused by ATD warping. Due to this, most stabilization techniques applied to ATD sequences distort real motion in the sequence. In this study we propose a new method to classify foreground regions in ATD video sequences. This classification is used to stabilize the background of the scene while preserving objects undergoing real motion by compositing them back into the sequence. A hand annotated dataset of three ATD sequences is produced with which the performance of this approach can be quantitatively measured and compared against the current state-of-the-art.

Keywords: Atmospheric Turbulence, Video Stabilization, Background Subtraction, Optical Flow, Foreground Detection.

*Philip E. Robinson, philipr@uj.ac.za

1 Introduction

In surveillance systems that capture images or video over a long range, typically more than 1 km, the effects of atmospheric turbulence become apparent and can severely degrade the quality of the captured imagery. The distortions caused by atmospheric turbulence occur due to the temporal variation of the refractive index in regions of atmosphere through which light rays travel from a scene to the imaging system. This variation of the refractive index causes the direction of light rays to be perturbed in a constantly fluctuating fashion.[1] This phenomenon, coupled with scattering caused by aerosols in the air, results in a number of distortions appearing in captured images.[2] The majority of the literature on atmospheric turbulence degradation (ATD) consider the distortions to fall into two categories, blurring and geometric distortions. Both these distortion types are temporally and spatially variant. The geometric distortions are caused by the changing refractive

index and cause elements of the scene to appear to warp and move in a pseudo-periodic fashion.[3] In addition to these two types of distortion caused by ATD there is also another type of distortion associated with imaging through the atmosphere. This is the loss of contrast caused by aerosols in the air which scatter the light rays travelling from the scene. These aerosols mostly consist of dust and water vapour and the effect is commonly referred to as haze.[4]

There exists an extensive literature focused on dealing with the effects of ATD, some authors have focused on tackling the distortion types in isolation but the majority address both the blurring and geometric distortions in tandem. Examples of work aimed at mitigating the blurring in ATD imagery can be found in the following sources.[1,5–8] These approaches mostly consist of blind deconvolution techniques and in the work of Aubailly et al.[6] a lucky-imaging approach is used. Stabilizing the geometric distortions in an ATD sequence is often an initial step in ATD restoration algorithms because by mitigating geometric distortions the assumptions about the blur characteristics in the sequences become simpler. Examples of the use of a variety of registration techniques to stabilize the geometric distortions in the video sequences before performing blind deconvolution or speckle imaging to deblur the images can be seen in the following work.[9–11] In another class of ATD restoration algorithms the registration of a set of frames containing ATD is a critical step to producing a higher quality reconstruction of the scene; a process called super-resolution which fuses data from multiple frames.[12–14] These approaches require accurate geometric registration in order to align sampled pixels from multiple frames which are then fused to produce a sharper, super-resolved reconstruction of the scene. Most of the techniques referenced above are designed to take a large number of frames from a sequence containing ATD and process them to produce a single high-quality frame. This is not ideal for processing video sequences where the salient objects tend to be transient and exhibit real motion.

The registration techniques used in the work referred to above all make assumptions about the nature of the apparent motion caused by ATD in these sequences. It is generally assumed that the motion is quasi-periodic in nature.[3] This is acceptable unless an element of the scene that is undergoing real motion does not conform to this assumption. In this case the objects undergoing real motion are typically blurred, or otherwise destroyed, by the registration techniques employed. Thus, for an algorithm aimed at mitigating ATD in video sequences that contain moving objects it is necessary to distinguish which elements in the scene are stationary and exhibit motion only caused by ATD and which elements are undergoing real motion.

In the field of automated video surveillance analysis elements of a scene that are salient and exhibit real motion are referred to as *foreground* objects and the non-salient regions are referred to as *background*. In this field a popular method for foreground detection is through the process of *Background Subtraction (BS)* where a model of the video background is produced and then subtracted from a video frame which should produce a high response in foreground regions that do not appear in the background model.[15]

There are a number of techniques in the literature that seek to classify regions undergoing real motion in ATD video sequences and these will be discussed in detail in the following section. These techniques are either aimed at reducing motion blur during the fusion process inherent in stabilization or at detecting and tracking moving targets of interest through time. However, to date, there exists no dataset or experimental methodology in the literature that can be used to quantitatively measure the accuracy of these techniques at classifying regions undergoing real motion in ATD video sequences, at a pixel level.

In this work we will present a novel technique for classifying foreground regions with a pixel level accuracy in ATD video sequences without needing to track regions. The proposed technique is a

hybrid approach using a background subtraction scheme that is designed to work in the adverse conditions created by ATD and optical flow analysis. These two algorithmic approaches are combined to produce a robust classification of foreground regions in video sequences containing ATD. Using this foreground segmentation, the background regions of the scene can be stabilized to mitigate the geometric distortion component of the ATD and the foreground regions can be composited back into the sequence. This does mean that the geometric ATD will still be present in the foreground regions themselves but the regions will be preserved and classified for further processing. We will also present an annotated dataset of three video sequences containing ATD consisting of 100 hand annotated frames from each sequence. We will use this dataset to compare the performance of our technique against the current state of the art ATD foreground detection algorithm[16] and three popular generic background subtraction techniques from the literature.

The remainder of this paper will be structured as follows. Section 2 will present an overview of the literature and existing techniques that fall into the context of this study. Section 3 will present the proposed technique in detail. Section 4 will contain a description of the dataset produced for this study, the experimental metrics and the methodology used to evaluate the algorithms and the results of the experiments. Finally, Section 5 will contain some concluding remarks.

2 Literature Study

The problem of ATD mitigation is quite extensively studied and this section serves to provide an overview of the literature and foundational theory relevant to the development of the algorithm presented in this work. Firstly, the most popular form of elastic stabilization used in ATD mitigation, which is optical flow based registration, will be discussed. Secondly, the popular foreground classification approach called background subtraction will be discussed and how it has been applied to

ATD sequences in the past will be presented. Finally, one novel approach to detecting foreground regions in ATD sequences, while stabilizing the background, exists in the literature and will be presented.

2.1 Optical Flow Stabilization

When performing stabilization on ATD sequences some form of elastic registration is required due to the spatially variant nature of ATD geometric distortions. By and large the techniques present in the literature employ some form of dense optical flow algorithm to measure the geometric distortion in a sequence. Optical flow techniques are used to compute an approximation of the 2D motion field representing the apparent motion of pixels between two frames of the same scene.[17] This is an intensely studied problem in the field of computer vision and a large number of different techniques have been tested using the Middlebury datasets.[18] It is apparent that the area is still receiving a great deal of attention. For the purposes of measuring and stabilizing the geometric component of ATD a wide variety of these optical flow techniques have been employed. While many methods exist to produce the flow field the following equation describes the geometric transformation we refer to as optical flow for two single channel image frames from a sequence.

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t) \tag{1}$$

$$I(x, y, t) = I(\phi(x, y, t), t + \delta t) \tag{2}$$

where (x, y) is a spatial coordinate of a pixel, $\phi(x, y, t)$ is a 2 dimensional mapping describing the geometric transformation, or shift vectors, for each pixel between two frames captured at times t and δt.

Optical flow measurements are used in two primary ways for stabilization in ATD sequences. The first approach is to compute the optical flow between adjacent frames in the sequence. In these methods for a given frame to be stabilized the optical flow is computed between the current frame and a window of adjacent frames. The assumption is then made that due to the quasi-periodic nature of geometric ATD that the pixel will oscillate around its true position.[9] Thus, the optical flow fields between the current frame and the adjacent frames are averaged which results in a vector, for each pixel, pointing to the estimate of that pixel's true position. The pixels in the current frame are then warped to their estimated stable positions by using the stable flow field to resample pixel values from their *stable* positions. Some form of interpolation is employed during this resampling phase. Examples of this approach can be found in the following work.[3,9,19]

The second more popular approach to using optical flow for ATD stabilization involves first computing a stable reference frame from a set of frames in a video sequence. The same assumption about the quasi-periodic nature of the geometric ATD is used for this process. Due to the fact that geometric ATD causes a scene element to randomly oscillate around its true position the intensity values of a window of frames can be averaged to produce a stable reference frame of the scene. This temporal mean image will contain motion blurring but the structures present in the reference frame will be in their true positions. The first use of this approach can be found in the work of Cohen et al.[20] Optical flow techniques can now be used to compute the shift vectors between a frame to be stabilized and the stable reference frame. The resultant flow field can be used to warp the current frame to its stable geometry by using the shift vectors to resample the pixels in the current frame from their stable locations. The temporal mean and the temporal median have both been used to produce the stable reference frame and these techniques are basic background models used in the field background subtraction, which will be discussed shortly. Examples of stabilization

techniques that use the temporal mean as a stable reference frame can be found in the following work[14,21] and examples of work that make use of the temporal median are given in the following articles.[11,22,23] These background models will be discussed further in the following section.

The first work to propose classifying foreground regions in ATD video sequences was presented by Frakes et al. who speculate that if one analyses an inter-frame optical flow field, regions with large flows compared to the mean could be considered to be undergoing real motion and can be classified with a thresholding operation.[3] However, details of implementation and performance of this idea are not presented in this work.

Gepshtein et al. were one of the first to propose a detailed optical flow method for foreground detection in their ATD stabilization work.[22] In this technique the current frame is registered to the stable reference frame, calculated using the temporal median, and the resulting stabilized frame is then subtracted from the reference frame. In background regions where the intensity values of the stabilized frame and reference frame are very similar this produces a low response but in regions where foreground objects are present the response is large. These regions are classified with a simple thresholding scheme and composited back into the stabilized frame. This approach is essentially the most basic form of background subtraction which will be discussed presently.

Fishbain et al. present a two-part foreground classification scheme that makes use of the optical flow data and a background subtraction technique to classify foreground regions.[24] Fishbain et al. first use a basic background subtraction operation as a coarse classification of foreground regions. The current frame is subtracted from a stable reference frame computed using the temporal median, if the response is higher than a selected threshold then the pixel is likely to be part of a foreground region. The second stage of the algorithm then examines the optical flow vectors between the current frame and the reference frame at each of these pixels. The translational optical

flow vectors are converted to polar form and the magnitude and angular components are analysed. It is assumed that optical flow magnitudes that are relatively small and irregular are due to ATD and larger regular magnitudes are due to objects undergoing real motion. These classes are selected using a fuzzy thresholding operation with manually selected thresholds. Next the angular component of the optical flow data is compared to the pixel neighbourhood. Areas that are moving due to ATD will exhibit a larger variance of directions in the neighbourhood than regions undergoing real motion which will exhibit a dominant direction. The resulting data is then classified using manually selected thresholds. These two measures are then combined using a fuzzy logic scheme to produce a foreground classification.

Huebner proposes a background classification technique which is performed using a block-matching based optical flow technique.[25] Block matching is a form of optical flow algorithm that uses a brute force type search to produce the shift vectors. During the block-matching process the patch around the current pixel is compared to a search space surrounding it using a distance measure as a similarity metric. Patches that are similar will produce a lower distance metric score. If the highest distance metric score for a given pixel is lower than some selected threshold it means that the current pixel patch is extremely similar to the background reference image. If this is the case, it can be classified as background. Foreground regions should always exhibit a relatively high distance metric score as they will not be similar to the background model in which they do not appear.

2.2 Background Subtraction

Background subtraction refers to a class of techniques that have been widely employed in the field of automated video surveillance to detect foreground objects.[15] The first step in these techniques is to produce some estimate, or model, of the background of the scene. Once the background model

has been calculated the following equation describes the general process that is used to produce a foreground classification using the background model.

$$F(x,y,t) = \begin{cases} 1 & \text{, if } |I(x,y,t) - B(x,y)| > T(x,y) \\ 0 & \text{, otherwise} \end{cases} \tag{3}$$

where $F(x,y,t)$ is the foreground mask which classifies the pixel at location (x,y) in a frame of the sequence at time t as foreground if the absolute difference between that pixel's intensity and the background model $B(x,y)$ at that location is higher than some threshold value $T(x,y)$. Note that in this formulation the threshold is not scalar but can be spatially variant. This process is based on the concept that the background model will only contain background information,when the background intensity values are subtracted from an input frame the regions in the input frame that are the same as the background will produce a very small difference. However, areas in the input frame that contain foreground objects will differ from the background and the absolute difference will be large. When the difference is larger than the threshold it may imply that an object exists in that region that is not present in the background. The thresholding scheme employed in this process is also a broad field of research and can range from manually selected scalar values to adaptive methods.[26]

The classic background models employed in background subtraction are a per-pixel temporal mean and a per-pixel temporal median.[15] These are also the most popular methods for calculating a stable reference frame in ATD stabilization algorithms.[24,27] These methods work on the assumption that any foreground objects moving in a scene will move fast enough so as not to occupy a given pixel for a very long period. Therefore, in a sufficiently large temporal window the majority of intensity

values for a given pixel will be made up of background data with some noise. The temporal mean is effective and computationally the simplest approach but it is prone to producing motion blur. The temporal median was first proposed by MacFarlane and Schofield to track piglets and is more resistant to outliers as only 50% or more of the intensity values in a window need to be samples of the background for it to produce a stable background model.[28] Far more complex background models exist but as shown by Elkabetz and Yitzhaky in video sequences containing ATD the more complex multi-modal models such as Gaussian Mixture Models (GMM) and Kernel Density Estimation (KDE) methods perform no better than the simpler unimodal temporal mean and median models.[29]

As previously discussed Gepshtein et al. use basic background subtraction after performing optical flow registration of a given frame to detect foreground regions in the frame that were destroyed during registration.[22] However, due to the geometric distortions present in ATD imagery basic background subtraction produces many false positives, especially near strong edges. This is caused by subtracting the distorted geometry from the stable background model which contains stable geometry. Due to this Fishbain et al. use basic background subtraction only as a first stage to their foreground detection scheme to limit the number of regions that they need to analyse using their more accurate optical flow based foreground detection scheme.[24]

The most recent application of background subtraction to foreground detection in ATD sequences can be found in work by Chen et al. and Apuroop et al.[30,31] These two algorithms are both focused on detecting and tracking foreground objects and not stabilization of the ATD distortions. Both approaches produce a temporal median background model, perform background subtraction and use an adaptive thresholding scheme for foreground classification. The difference between the two approaches can be seen in how the adaptive threshold is calculated. Chen et al.[30] calculate their

threshold for each pixel based on the temporal median of a window of previous absolute difference values calculated during the background subtraction phase defined as

$$D(x, y, t) = \mid I(x, y, t) - B(x, y, t) \mid. \tag{4}$$

Their threshold is then calculated as

$$T(x, y, t) = K \cdot med(D(x, y, t - N : t)) + O, \tag{5}$$

where N is the number of frames in the window and K and O are manually tuned scalar values. This scheme produces an adaptive per-pixel threshold based on the temporal history of that pixel which provides some resilience to the noise caused by varying levels of ATD. Apuroop et al. propose an adaptive threshold that makes use of a more complex statistical analysis of the histograms of the current frame and the background model.[31] This scheme is also tuned based on the temporal variance of the background model to provide some adaptivity to the turbulence level present in the sequence.

Background subtraction in the presence of the geometric warping in ATD video sequences produces significant noise, especially around sharp edges. To mitigate the false positives that arise both Chen et al. and Apuroop et al. make use of tracking to validate that a region classified by background subtraction belongs to a real moving object and is not a transient response caused by ATD warping.[30,31] When a region is first detected it is tracked over a number of frames to ensure it is undergoing consistent motion and is persistently detected in a series of frames before the region is classified as being a real moving object. Detected regions produced by ATD warping will be

transient and will not persist in the sequence for multiple consecutive frames. This approach is effective but it has an inherent latency as an object can only be classified as foreground after being tracked for a number of frames. In addition, as the techniques are both focused on detecting and tracking targets a positive result is considered to be detecting a blob whose bounding box lies at least 50% within the bounding box of the true target. With this context the thresholding levels are chosen to reduce false positive detections as opposed to detecting accurate object contours at a pixel level.

2.3 Three-term matrix decomposition

Oreifej et al. present a novel method for classifying foreground regions and producing a stable background scene based on a three-term low-rank matrix decomposition.[16] This approach seeks to isolate three distinct components of an ATD sequence, namely, the background, motion due to turbulence and objects that are exhibiting real motion. The characteristics of these components are modelled directly and a constrained minimisation process is performed to decompose the matrix containing the frames of the sequence into the three components. This approach produces excellent results and the authors graciously provide their implementation which will be used for comparison.

3 Proposed Technique

3.1 Proposed Algorithm Overview

The proposed algorithm builds on the existing literature surrounding the detection of foreground regions in ATD video sequences and optical flow based stabilization. In Figure 1 a high-level overview of the structure and data flow in the proposed system is presented. The design choices of each stage, based on performance experiments, are presented in this work and the stages which

the authors consider to be the primary novel contributions are highlighted in green.

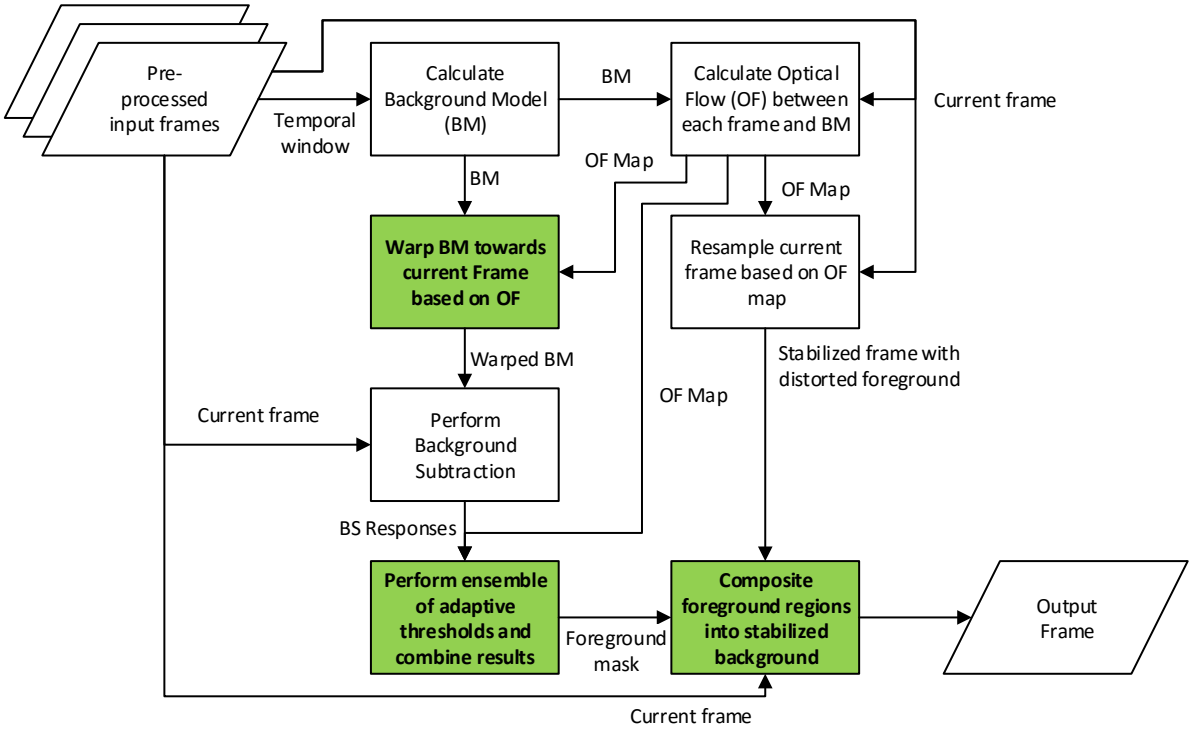


Fig 1 Overview of the structure and data flow of the proposed algorithm with areas of contribution highlighted

From Figure 1 it can be seen that the system takes pre-processed frames as input. These frames have their contrast enhanced as discussed in the following sections. The next step is to calculate the background model which is used by both background subtraction and optical flow stages. This is discussed in Section 3.2. The design of the optical flow analysis stage is discussed in Section 3.3. The design and contribution to the classic background subtraction stage is discussed in Section 3.4. The proposed adaptive thresholding scheme and novel combination of the ensemble of data used for classification is presented in Section 3.5 and finally the details of the final algorithm integration is discussed in Section 3.6.

3.2 Background Model Selection

The first step in building the proposed algorithm is to select a background model to be a stable reference frame for the optical flow registration stage and for background subtraction. Heubner and Scheifling have investigated three fundamental statistical models for the purposes of background modelling in ATD sequences; the temporal mean, median and mode.[32] The mode is shown to be very noisy and of the three models the median produces the sharpest results with the least amount of motion blur. These models all assume that a background pixel will exhibit a unimodal temporal distribution, however in modern video surveillance literature it is generally considered that more complex multi-modal distributions are more accurate. These include GMMs and the KDE.[15,33] Haik and Yitzhaky investigated the KDE method[34] and Elkabetz et al. investigated GMMs for use in ATD sequences.[29] In both cases it is found that the temporal median performs at least as well as the more complex models in the presence of ATD. This is because the effects of ATD are completely independent of the content in the scene and essentially modulates the scene content with a quasi-periodic random geometric and photometric distortion. This process does not produce clean multi-modal distributions that KDE and GMM can model better than basic unimodal statistics.

In our work we found that the temporal mean and median produce similar levels of stability for equivalent sized temporal windows. The temporal median ostensibly produces a sharper background model than the temporal mean but the temporal mean is computationally far simpler to calculate so we sought to confirm whether the temporal median produced sharper and higher contrast background models than the temporal mean. We performed our tests on an ATD sequence we will refer to as the *Site* sequence (a frame of this sequence is shown in Figure 12) which will form part of the dataset released with this work described in Section 4.1. Using this sequence we

calculated both the temporal mean and median using an increasing number of frames, N, in the temporal window. For each window size we calculate the mean 2D Michelson Visibility score[35] with a local block size of 5x5 pixels to measure local image contrast. This window size is chosen by DelMarco and Agaian[35] as it allows the metric to test local contrast without factoring larger image structures into the metric. The results are shown in Figures 2. We also use the Marziliano sharpness metric[36] score to measure image sharpness, the results are shown in Figure 3.

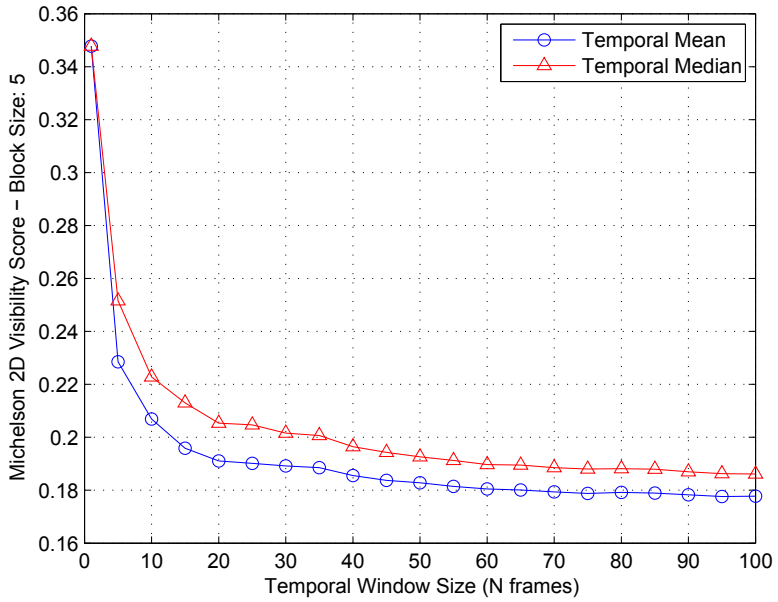


Fig 2 Michelson Visibility metric scores for varying temporal window sizes in the *Site* sequence, higher values indicate higher contrast.

It is apparent that the temporal median background model is both sharper and exhibits higher contrast than the temporal mean for all temporal window sizes tested. The quality of the background model is of utmost importance in this work and generally the computational complexity is a concern. However, we found that a GPU implementation of the temporal median could run in real-time for a megapixel image with a 50 frame temporal window on a Nvidia GTX Titan GPU. We chose the size of 50 frames for the temporal window in our implementation as we found it was

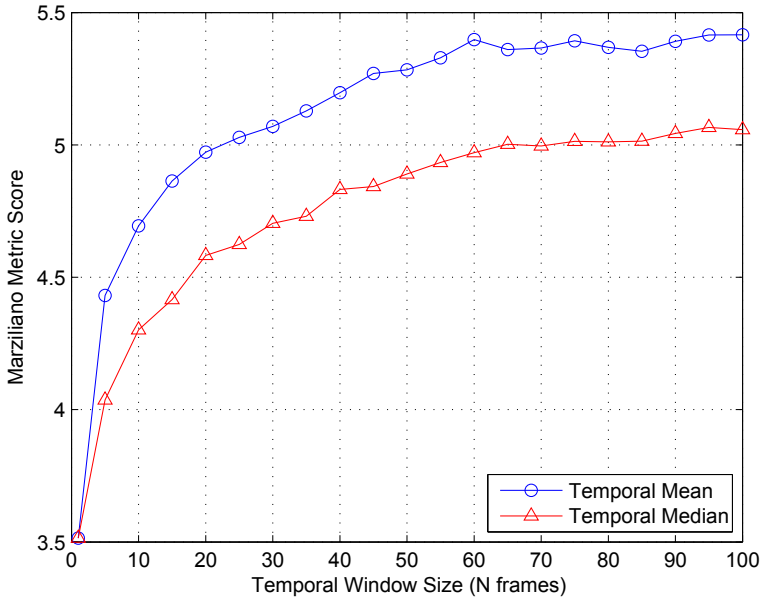


Fig 3 Marziliano metric scores for varying temporal window sizes in the *Site* sequence, lower values indicate higher sharpness.

a good balance between minimizing the blur introduced into the background model and being able to exclude the majority of foreground objects undergoing real motion.

3.3 Optical flow registration and segmentation

Optical flow algorithms are an intensely studied field and their implementation tends to be very complex. As such in this work we made use of publicly available implementations of these algorithms. We investigated three algorithms in particular and in choosing one had to balance their computational running times, accuracy and, critically, their ability to gracefully handle foreground regions that do not appear in the background model. In this work we investigated a modernised Horn-Schunck (HS) algorithm implemented by Sun et al.,[37] Farneback's algorithm[38] and the Dual TV-L1 algorithm by Zach et al.[39] The latter two algorithms have GPU based implementations available in OpenCV.

To test the stability of all three algorithms we tested each algorithm using the original sequence and

a pre-processed version of the sequence. The pre-processed sequences have been enhanced using an Adaptive Multi-Scale Retinex (AMSR) algorithm[40] to improve the contrast and a blind de-convolution algorithm to sharpen the images.[41] The pre-processing enhancement of the sequences provides sharper and higher contrast features for the optical flow algorithms to track but these steps introduce artefacts and amplify noise. To test stability we compute the stable background model using a sliding temporal window of 50 frames, compute the optical flow between the current frame and the background frame and then resample the enhanced version of the current frame's pixel values from the stable locations indicated by the optical flow vectors. The resampling is done using a bilinear interpolation scheme. The result is a stabilized video sequence and we measure its stability by calculating the mean per-pixel sum-of-absolute difference (SAD) between each frame in the sequence. The mean per-pixel SAD score between two frames, captured at time t and $t+1$, is calculated using equation 6.

$$S(t, t+1) = \frac{\sum_{x,y \in \Omega} \mid I(x, y, t) - I(x, y, t+1) \mid}{w * h}, \tag{6}$$

where Ω is the 2D image domain, w is the image width in pixels and h is the image height in pixels. The more stable the frames are, the lower this score will be as there will be less variation in the per-pixel intensity values. When we calculate the optical flow using the original image data we still warp the enhanced frame using the computed flow data so that the SAD scores are directly comparable. The results of these tests can be seen in Figure 4.

We found that the Sun et al.'s modern Horn-Schunck algorithm was the most stable algorithm when working with the unenhanced frames but that the Dual TV-L1 algorithm was the most accurate overall when working with enhanced frames. The Dual TV-L1 algorithm is clearly more

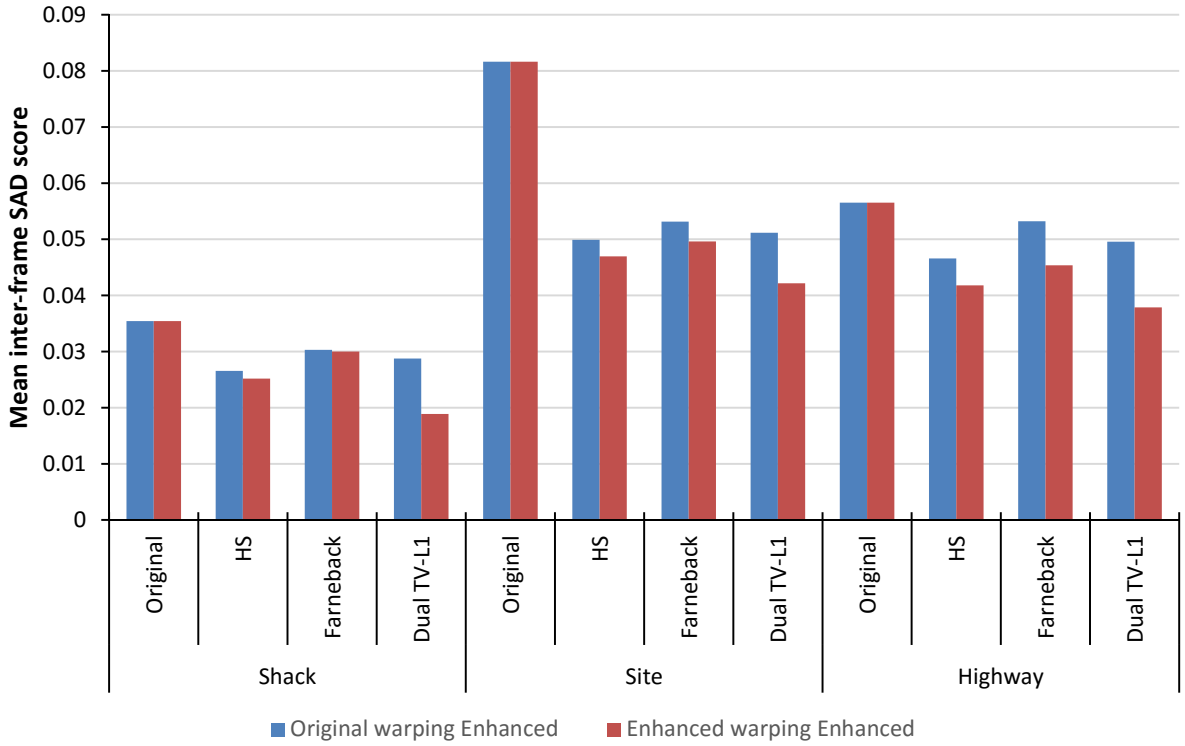


Fig 4 Comparison of mean inter-frame SAD scores for three enhanced test sequences after being warped using the three test optical flow algorithms. Results are shown for warping the enhanced frames based on optical flow between the frame and the background model created from both unprocessed and enhanced frames. Lower scores indicate higher stability.

robust and able to deal with the artefacts produced by the enhancement process. The Dual TV-L1 algorithm also has a very beneficial feature in that the regularisation function it employs allows for discontinuities in the flow fields. This is important because it accommodates calculating flows in the foreground regions that do not appear in the background model where the HS algorithm averages these artefacts into the surrounding field. The Dual TV-L1 implementation also has the added benefit of being the fastest implementation of all those tested. Videos demonstrating the comparative stabilized outputs of the HS and Dual TV-L1 algorithms can be found in the supplementary material accompanying this article on the project website.[42] When viewing these results as videos it is evident that the Dual TV-L1 algorithm produces the most stable output.

As noted by Fishbain et al.[24] the magnitude of the mappings between the current frame and the background model can be used to classify foreground objects undergoing real motion that is larger than the motion caused by ATD warping. An example of the optical flow magnitude for a frame in the *Site* sequence can be seen in Figure 5. This image has been normalized for easier visualisation.

Fig 5 Normalized optical flow magnitude from the *Site* sequence with foreground region indicated in red

In Figure 5 the foreground region stands out from the mean optical flow magnitudes caused by the ATD in the image. This data can be used to classify some foreground regions but it is not very effective, often missing true foreground regions which is why Fishbain et al. make use of a number of other cues in their work. We include it in this work as it uses data that we are computing for the stabilization stage in the algorithm in any case and while the background subtraction stage we describe in the next section produces far superior classifications with sharper boundaries there are cases where a foreground object is a very similar colour to the background model and thus the background subtraction stage fails. In these cases the motion cues from the area around the foreground regions often can still distinguish the foreground object which is why this data is included in our classification ensemble. To threshold the optical flow magnitude data we use an adaptive

global thresholding scheme based on the global statistics of the data being thresholded. The global threshold is calculated as follows:

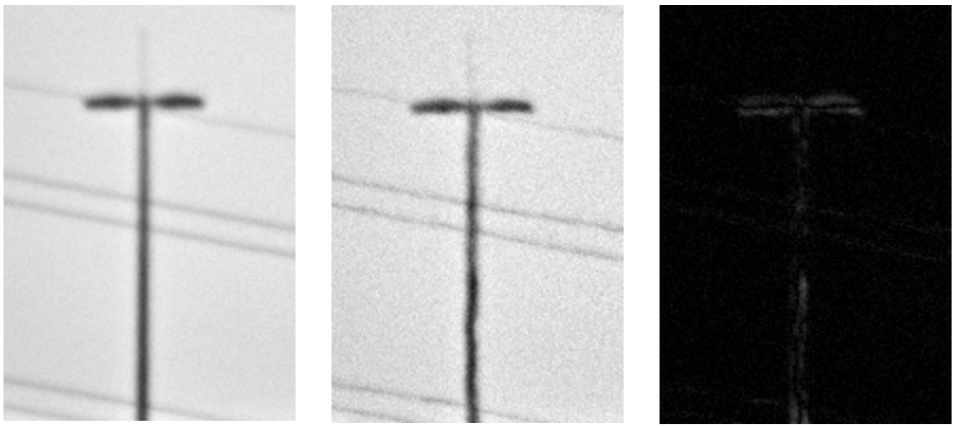
$$T = \mu + K\sigma \tag{7}$$

where μ is the global mean of the data, σ is the standard deviation and K is a tuning parameter we select to set the sensitivity of the thresholding operation. This scheme allows us to select a threshold which is a certain number of standard deviations above the mean optical flow magnitude which is caused by ATD. We calculate two thresholds for every frame and employ the hysteresis technique first popularised by Canny which will be discussed in detail in Section 3.5.

3.4 Background Subtraction in ATD sequences

As discussed in the Section 3.2 the temporal median background model not only effectively excludes foreground objects from the background model, as it does in conventional contexts, it has an added benefit when applied to ATD video sequences. In sequences with ATD the per-pixel temporal median also produces the stable geometry of the scene at the cost of some motion blurring. The fact that the geometry of the background model is stable is the reason that the background subtraction (BS) process described in Section 2.2 produces so many false positives. Figure 6 demonstrates why this is the case. The lamp post in the background model is straight and as the sliding window used to compute the background model moves through the sequence its local and global position will remain stable. However we can see that the structure of the lamp post in a single frame from the sequence is very warped. This misalignment of these structures causes a high response when BS is performed. This effect is strongest around sharp edges in the image which is why Chen et al. incorporate an edge map into their system which weights the probability of a detected blob being a false positive higher in regions surrounded by hard edges.[30] While this is acceptable when one

is only trying to detect the presence of an object it explicitly rejects the edge regions which define the boundaries of foreground objects.



(a) (b) (c)

Fig 6 Illustrating background subtraction of a lamp post in *Highway* sequence. (a) Background Model, (b) Current Frame, (c) Absolute Difference

This phenomenon is one of the primary obstacles to making use of background subtraction for pixel-level foreground classification tasks in ATD sequences. To minimise the effect of this structural misalignment during the background subtraction stage we propose a simple modification to the BS stage which produces significantly better results. The optical flow data we calculate for the purposes of stabilization provides a mapping that can be used to warp a given frame towards the geometry of the stable background model. However, in the foreground detection phase of the algorithm we aim to detect which pixels in the current frame are foreground pixels. These pixels however occupy their ATD warped geometry, not the stable scene geometry. So we propose that we use the optical flow data to warp the *background model* towards the warped geometry of the current frame before performing BS. In this way we avoid the misalignment of the stable and warped geometries and the response of the BS stage will occur in the warped locations of the current frame and not at the stabilized positions. This process is described in the equation 8 where the negative of the shift vectors in the optical flow map are used to perform the warping before performing the

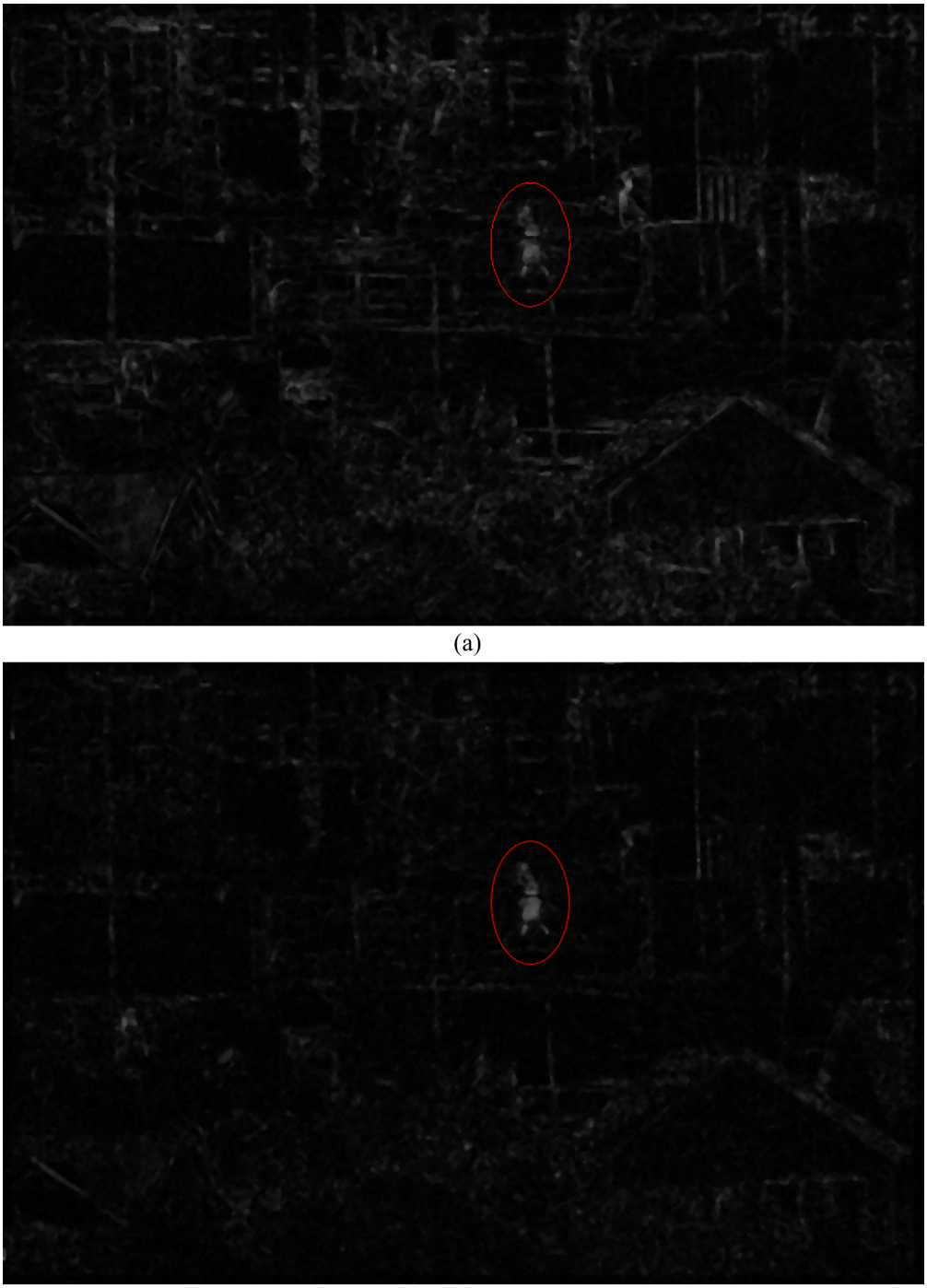
absolute difference.

$$D(x, y, t) = |I(x, y, t) - B(-\phi(x, y, t))| \tag{8}$$

where $-\phi(x, y, t)$ is the negative of the function containing the vectors of the optical flow field between the current frame and the background model and *D(x,y,t)* contains the background subtraction response for this time step. The results of applying this simple modification to the background subtraction stage is shown in Figure 7.

It is apparent that the overall response of BS is significantly reduced in the background regions of the frame and around the sharpest edges. Furthermore, the response in the foreground region is stronger and the boundary is better defined. It is difficult to fully appreciate the results from the figures embedded in this article, the full resolution images and a video comparing the results are included in the supplementary material.[42] To quantitatively demonstrate the reduction in BS response due to ATD warping we calculated the mean BS response over 100 frames for each of our three sequence with and without prior warping. The results are shown in Figure 8 and it is apparent that the proposed background warping stage decreases the BS response caused by ATD warping in all the sequences. This will make the thresholding process better able to classify true foreground regions and reduce false positives.

For the BS data we employed both global and local per-pixel adaptive thresholding schemes. Local per-pixel thresholding schemes are used in cases where there is a non-uniform spatial distribution of activity in an image. In these cases the adaptive threshold can be calculated from the statistics of the data in the local region as opposed to the whole scene. This makes these thresholds adapt to the specific activity in the local region which can be affected by local image structures and spatially variant elements of ATD. Using the *Highway* scene as an example the sky region in this scene



(a)

(b)

Fig 7 Illustration of background subtraction result for a frame in the *Site* sequence with and without prior warping of the background model (constructed from 50 prior frames) towards the current frame, a foreground region is indicated in red. (a) Background subtraction result with no prior warping. (b) Background subtraction result with prior warping.

has no texture or structure in it and as such produces a consistently low BS response despite the ATD warping in the sequence. This does not mean that there is no ATD in that region but that it

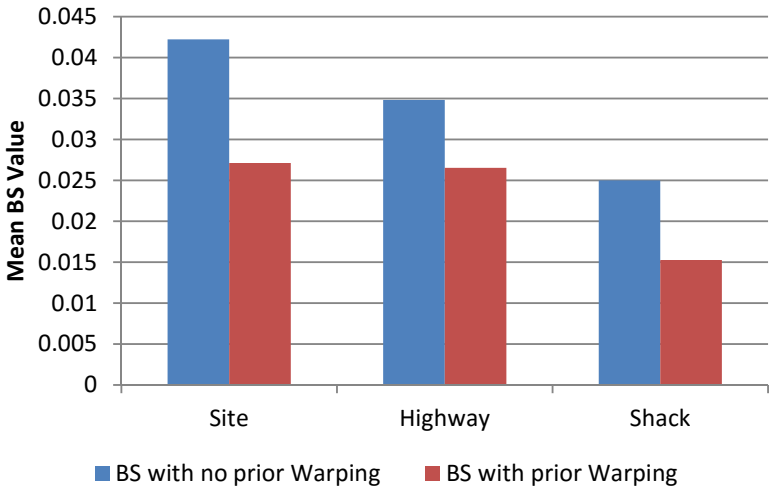


Fig 8 Mean BS response values averaged over 100 frames for the three ATD sequences with and without prior warping.

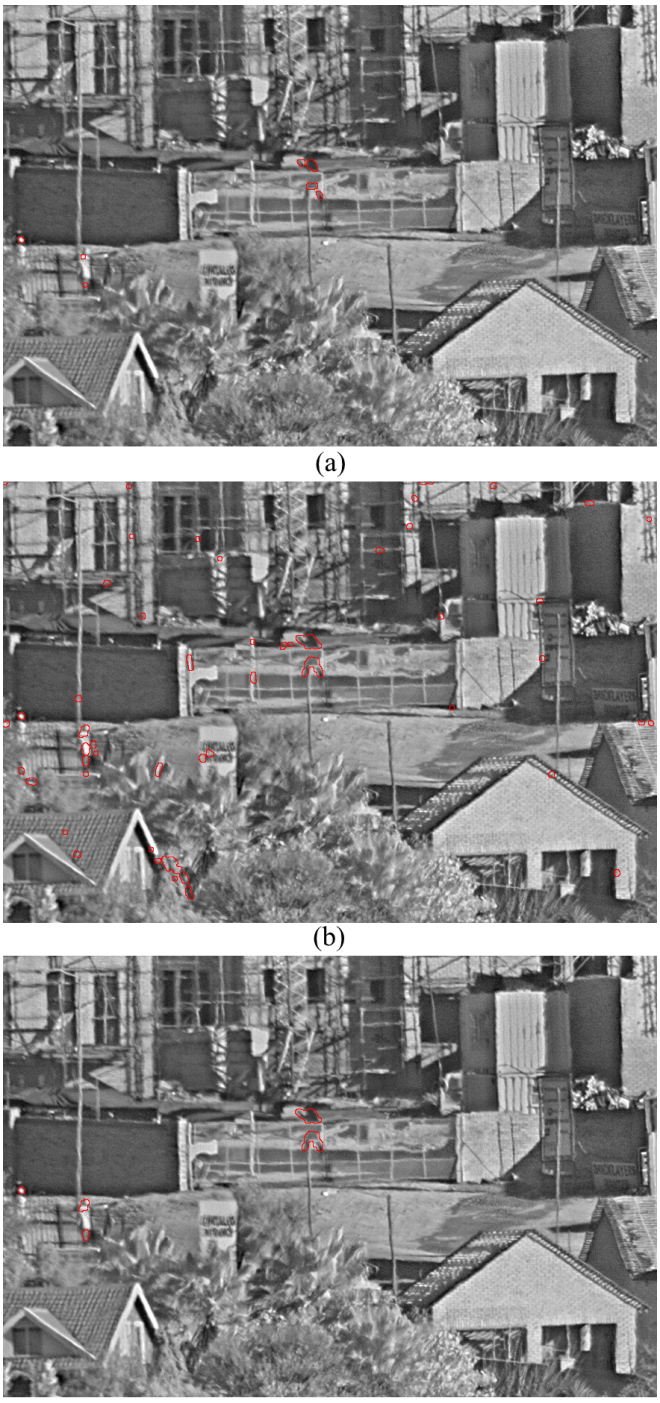
cannot be perceived due to the uniform intensity of the sky. This large region throws off the global statistics of the BS response and can cause false classifications.

We use the same per-pixel adaptive threshold as Chen et al. which is described in equation 5, this threshold is based on the temporal statistics of the pixel which takes into account the mean activity of that pixel through time.[30] This technique, however, has its own problems as shown in Figure 9 which shows the normalized per-pixel median values for a window of 50 frames from the *Highway* sequence. In the highway region in the bottom left of Figure 9, where there is a large amount of constant traffic, the per-pixel median gives a consistently high response for the whole region. This response persists for many frames after the foreground objects have traversed the region. This means that the threshold to detect a foreground region in this busy area is significantly higher than it should be and objects will often be missed in this region. As such we also employ a global threshold and take the union of the response of the global and local per-pixel threshold classifications for our system.

Fig 9 Normalized per-pixel median of the absolute difference values for 50 frames of the *Highway* sequence illustrating the high persistent response in areas with a high degree of foreground traffic.

3.5 Hysteresis Thresholding

Whether a global or local thresholding scheme is used the challenge is always to choose a threshold that results in classifications that accurately capture the boundaries of objects but limit false positive classifications. We found that it was not feasible to achieve these two objectives with a single threshold, global or local. One can choose a low enough threshold that the resulting classification fully captures the boundaries of foreground regions but there will be many false positive regions as demonstrated in Figure 10 (b). Otherwise one can choose a threshold high enough to have very few false positive classifications but the detected regions will only be the central mass of an object and not adequately describe the outer boundary of the object as demonstrated in Figure 10 (a).



(a)

(b)

(c)

Fig 10 Illustration of the classification results using a global threshold calculated with high and low multiplier values (*K=10* and *K=3* respectively) and the result of the *Grow From Intersection* operation. (a) Classification using a high global threshold value (*K=10*). (b) Classification using a low global threshold value (*K=3*). (c) Classification from *Grow From Intersection* operation.

To solve this problem we employ a method first popularised by Canny which he calls *Hysteresis*.[43] This approach uses the two thresholds described above, firstly a high threshold to identify the central mass of true positive regions and a low threshold which fully captures the region's boundary. Hysteresis then uses the classification produced by the higher threshold (T_{high}), which should contain few false positives, to determine which blobs produced by the lower threshold (T_{low}) should be kept and which are false positives. We do this with an operation we call *Grow From Intersection (GFI)*. This is done by first finding the intersection of the sets of pixels classified by the higher threshold, S_{high} in equation 9, and the lower threshold, S_{low} in equation 10, which is calculated as in equation 11. We then used a classic 8-connected 3x3 structuring element (H) to morphologically dilate at each pixel location (x, y) in the set S_{GFI} so that they grow within the larger blobs from the S_{low} set as described by equation 12. This process is iteratively repeated until no more changes occur to the S_{GFI} set. The result is that we keep the blobs from the lower threshold classification, which have better defined boundaries, that intersect with the smaller blobs from the higher threshold classification which we are confident are not false positives.

$$S_{high} = \{(x, y) | I(x, y) \geq T_{high}\} \tag{9}$$

$$S_{low} = \{(x, y) | I(x, y) \geq T_{low}\} \tag{10}$$

$$S^0_{GFI} = \{S_{high} \cap S_{low}\} \tag{11}$$

$$S_{GFI}^{n+1} = \left\{ \left(\bigcup_{(x,y) \in S_{GFI}^{n}} H_{(x,y)} \right) \cap S_{low} \right\} \tag{12}$$

The output of this process can be seen in Figure 10 (c) where the contour around the person in the centre of the scene is the complete contour produced by the lower threshold but the false positives have been excluded. The classifications on the left of the scene are also moving people but they are obscured by vegetation. We make use of this technique for all three thresholds that we use in our system, which are the global optical flow magnitude threshold, the global BS threshold and the local per-pixel BS threshold which will all be discussed in detail in the following section.

3.6 Hybrid Foreground Detection Algorithm

The complete architecture of our proposed foreground detection system can be seen in Figure 11. It can be seen that we combine classifications from three sets of data in order to cope with the high degree of geometric distortion introduced by ATD.

Firstly, we analyse the BS response, $D(x, y, t)$, on a per-pixel basis by looking at the temporal history of a given pixel in a similar manner to the approach of Chen et al. shown in equation 5.[30] In this method we find the median of the temporal history of the pixel value and the manually tuned gain, K, selects how many times larger a BS response must be than the median to be classified as foreground. We found that the per-pixel temporal analysis of the BS responses was by far the most descriptive of the three extracted datasets and generally correctly classified the majority of the foreground regions. However, as discussed in the previous section in areas of strong texture or high foreground traffic the temporal history becomes saturated. For this reason we incorporate a global classification stage which calculates its threshold based on the global statistics of the current BS frame as shown in equation 7. When calculating the global thresholds the tuning parameter, K,

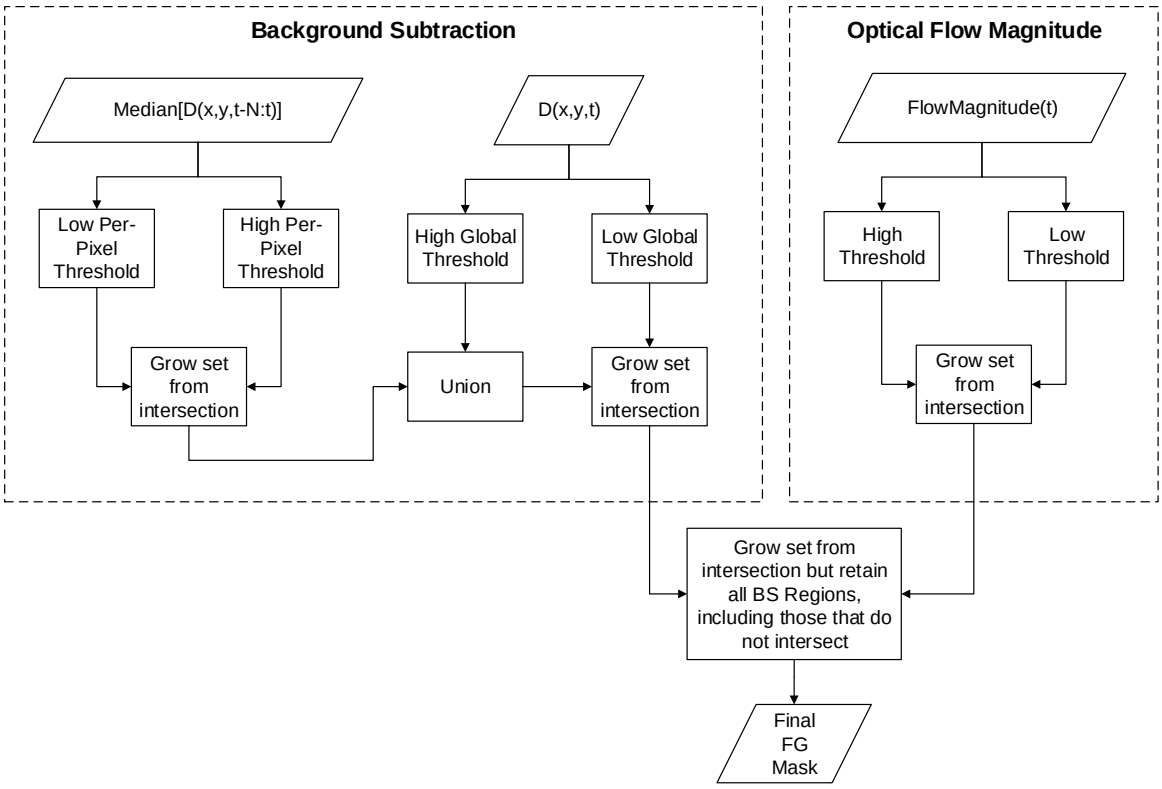


Fig 11 Illustration of the ensemble of thresholding operations and the combination of the resulting sets that make up the proposed classification scheme.

selected how many standard deviations about the mean a BS response must be before being classified as foreground. Finally, in regions where the foreground object and the background exhibit similar intensity values the BS response will be very low and classification will fail. To catch these cases we also incorporate an optical flow cue into the ensemble by examining the optical flow vector magnitude calculated between the current frame and the background model. The optical flow magnitude is also thresholded using the global statistics as described by equation 7.

All three adaptive thresholding stages employ the *hysteresis* method to cope with the high degree of geometric distortion in ATD sequences. The low threshold was chosen so that the boundaries of correctly classified regions represented accurate segmentations of those regions. If this threshold was set too high then the boundary would not be accurately defined and would segment only the

centre mass of the region. If the low threshold was was set too low this boundary would be blurred and poorly defined in addition to the increased likelihood of false positive classifications. The tuning of the high thresholds was done so as to minimize the chances of false positive classifications however if the upper threshold was set too high then there is an increased likelihood of missing foreground regions during the classification.

Figure 11 shows how the results of the three classifiers are combined. The per-pixel and global thresholding stages of the BS data are combined using a union. The optical flow magnitude threshold results are then combined with the combined BS results using the *GFI* operation to fill in any gaps that the BS process missed and the optical flow magnitude data caught. However we retain all the classifications from the BS stage, not just where the blobs intersect with the optical flow magnitude classifications as the BS data is our primary classifier and the optical flow magnitude is a secondary classifier. The result is a boolean classification mask which describes the pixel locations of foreground objects in the scene.

We manually tuned the K multipliers for all of the adaptive thresholds to give good classification performance on the three test ATD sequences and 6 additional ATD sequences not included in this report. When tuning the parameters the goal is to select parameters that maximize the number of true positive classifications and minimize the number of false positives and false negatives that the thresholds produce. The chosen values can be seen in Table 1. Potential future work will be to tune these gain parameters automatically based on measurements of the scene being analysed.

Table 1 Empirically chosen multipliers for the proposed ensemble of adaptive thresholds

	Low K	High K
Local Per-Pixel BS Thresholds	4	6
Global BS Thresholds	3	8
Global Optical flow magnitude thresholds	4	6

The foreground classification algorithm produces a binary classification map $F(x, y, t)$ where a value of 1.0 indicates foreground and 0.0 background. We can now use this map to segment out the foreground regions of the current frame and composite them back into the stabilized sequence. This is done by first dilating the foreground mask using morphological operators and applying a Gaussian blur to the result to soften the edges of the foreground mask. This mask is then used to blend together the stabilized background and foreground elements of the current frame as described in equation 13.

$$\begin{aligned} I_{composite}(x, y, t) = F(x, y, t) \times I(x, y, t) \\ +(1 - F(x, y, t)) \times I_{stable}(x, y, t) \end{aligned} \tag{13}$$

4 Experiments

To test the performance of our system we first inspect the resultant classifications and stability of the composited videos for our three test sequences described below. We use a manually labelled ground truth dataset to compare the classification performance of our system against three popular foreground classification algorithms from the literature and the specialized ATD classification algorithm from Oreifej et al.[16] The three generic classification algorithms to be considered are a GMM based method by Stauffer and Grimson,[44] a KDE based method by Elgammal et al.[45] and a very modern method that uses an ensemble of multiple image features called SuBSENSE by St-Charles et al.[46] The GMM and KDE algorithms were selected because they are both classic examples of BS techniques and are often used as reference implementations in the background subtraction literature. The SuBSENSE algorithm was selected because, at the time of writing, it was the highest performing algorithm on the ChangeDetection.net benchmark, a result that has

persisted for a number of years.[47]

The implementations of the three generic BS algorithms were obtained from the BGS Library project[48] whose author, Sobral, collected the implementations from authors and incorporated them into a library based on OpenCV. The classification performance of the proposed method was tested with and without warping the background model before background subtraction to demonstrate the improvement in accuracy this approach produces. When testing the proposed method and the three generic methods the experiments were conducted with pre-processing of the frames using the AMSR algorithm for contrast enhancement. The implementation of the method by Oreifej et al. had a contrast enhancement stage using the CLAHE algorithm built in. All implementations were executed using default parameters.

4.1 Dataset

In existing work in the literature foreground segmentation accuracy has been discussed in terms of having successfully detected and tracking the presence and trajectory of a target. In the work of both Chen et al.[30] and Apuroop et al.[31] a positive detection is considered to have occurred when a detected blob's bounding box falls at least 50% within the bounding box of the target blob. This approach does not aim to measure the accuracy of the detected object contours at a pixel level, which is the accuracy that would be required to composite foreground regions into a stabilized sequence.

To allow for quantitative measurement of this level of accuracy of foreground detection algorithms we present a dataset consisting of three sequences containing ATD effects and real moving objects. We have hand annotated 100 frames from each sequence using the labelling scheme proposed by the ChangeDetection.net benchmark project which maintains annotated datasets for the evalua-

tion of conventional background subtraction algorithms.[47] The following labels were used in our dataset:

- 0: Background
- 85: Outside region of interest
- 100: Vegetation or dynamic background regions
- 170: Unknown motion
- 255: Foreground

The *Unknown motion* label is used to mark regions where the motion blur makes it challenging for a human to determine where the true boundary of an object starts. As such this label is used to label ambiguous regions. We added one additional label to the scheme to indicate dynamic background regions that are exhibiting real motion but are part of the background. This includes waving vegetation and flapping plastic sheets in the *Shack* sequence. Conventional background subtraction algorithms are designed to handle dynamic backgrounds but in the context of ATD sequences we excluded these regions for this work and leave this capability for future work. These two ambiguous region types are omitted during the calculation of the classification metrics.

It is important to note that the process of hand annotating ATD sequences is very challenging as the sequences contain significant blur and very low contrast. We found that even by pre-processing the sequences using an Adaptive Multi-Scale Retinex (AMSR) algorithm[40] to improve the contrast and a blind deconvolution algorithm to sharpen the image[41] there were still many ambiguous regions where the person doing the labelling had to exercise their best judgement. Therefore the quality of the annotated dataset is certainly open to debate but will still serve as a useful tool for algorithm

evaluation and comparison purposes.

In our dataset we do not annotate the first 100 frames of the sequence so that they could be used for training of algorithms. The next 100 frames are annotated. The three ATD sequences were provided by the CSIR of South Africa's Optronic Sensor Systems group.[49] The annotated dataset is freely available on the project's webpage.[42] Example frames from each of the three sequences can be seen in Figures 12 through 14.

Fig 12 Frame from the $Site$ sequence which is an 8-bit grey scale sequence captured at a range of 5.5 kilometres.

Fig 13 Frame from the $Highway$ sequence which is an 8-bit grey scale scale sequence captured at a range of 9 kilometres.

Fig 14 Frame from the $Shack$ sequence which is an 8-bit grey scale sequence captured at a range of 10 kilometres.

4.2 Metrics

The metrics employed to measure the performance of binary classification algorithms are calculated from the number of pixels that are classified as True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN). Classification algorithms all need to tune themselves to avoid producing excessive false positives or false negatives. Thus it is important to make use of metrics that measure both of these aspects of the algorithms. For example the classic metric of Recall favours algorithms with a low false negative Rate whereas the classic metric of Specificity favours algorithms with low false positive rate. We will thus make use of an ensemble of the popular classification metrics to examine the performance of the tested algorithms. The list of metrics and how they are calculated are shown in Table 2.

Table 2 List of binary classification metrics

Recall (Re)	$\frac{TP}{TP+FN}$
Specificity (Sp)	$\frac{TN}{TN+FP}$
Precision (Pr)	$\frac{TP}{TP+FP}$
F-measure	$\frac{2.Pr.Re}{Pr+Re}$

The Recall metric measures the proportion of correctly classified positive pixels relative to

the total number of positive pixels in a frame. The Specificity metric measures the proportion of correctly classified negative pixels relative to the total number of negative pixels in the frame. The Precision metric measures the proportion of correct classified positive pixels relative to the total number of positively classified pixels in the frame. Finally we calculate the F-measure, which is also called the F_1 Score that considers both the Precision and the Recall metrics and is only high when both of those metrics are high. A more detailed discussion of these metrics and their bias' can be found in.[50]

4.3 Experimental Results

To appreciate the accuracy of the proposed foreground classification scheme during the course of a video sequence one must view the results as a video. This makes it far clearer to the viewer where the foreground regions are that are undergoing real-motion. Video files showing the classification results of compositing the foreground regions onto the stabilized background for the three sequences can be found in the supplementary material that accompanies this work.[42]

From these videos one can see that the accuracy of the classification of the foreground regions is fairly good with minimal false positives. Each sequence does present a different challenge however. In the *Shack* sequence the walking people are correctly classified but in the centre of the frame is the shack that is covered by a plastic sheet that is flapping in the wind. This is an example of a dynamic background which is changing in appearance with time but is not actually part of the foreground. The proposed algorithm does detect these regions as foreground objects for a number of frames. In the *Site* sequence there are two main challenges. There is a large stand of vegetation in the frame of which a portion waves in the wind from time to time. As a human observer it is actually quite difficult to determine exactly what motion in the vegetation is true motion or what

is caused by the ATD. In the *Site* sequence there is also a person standing to the left of the sign who spends a large portion of the sequence changing their posture and appearance but not actually moving. Towards the end of the sequence this person starts to walk very slowly to the left. This is a challenging target to classify as for much of the sequence their real motion is smaller than the ATD warping. In the *Highway* sequence we see a region of the scene which experiences a very high degree of foreground motion. This results in the background model containing a lot of blur in that region and in addition many of the vehicles in this scene have very similar intensity values as the background which makes them hard to pick out against the background. For this sequence the proposed algorithm relies very heavily on the optical flow magnitude for its segmentation.

The following Tables 3 through 5 present the classification results for all three sequences. For the Recall, Specificity, Precision and F-Measure metrics a higher score is better. The best scores for each metric and experiment are indicated in bold. When calculating these metrics a 10 pixel border region is excluded for all algorithms as this border area often contains artefacts caused by the optical flow algorithms. The results show some of the weaknesses of the popular classification metrics. The Specificity metric is not very descriptive because of the large proportion of true negatives that appear in images. Furthermore, the Specificity metric will be 1 if an algorithm does not attempt to make any classification at all which means that there cannot be any false positives. This is not a desirable feature in an algorithm but produces a high score. Similarly, the Precision metric also produces a higher score when an algorithm is more conservative with its classification, favouring algorithms that are overly cautious. For these reasons the F-measure represents a measure of a balance between the number of false positives and false negatives an algorithm produces.

From these results we can see that the generic KDE, GMM and SuBSENSE algorithms struggle with the ATD sequences. The SuBSENSE algorithm does achieve some top scores for Specificity

Table 3 Results for *Site* sequence

	GMM	KDE	SuBSENSE	Oreifej	Proposed	Proposed (No Warp)
Rc	0.62644	0.40782	0.28941	0.63123	**0.74702**	0.70328
Sp	0.92689	0.99086	**0.99998**	0.99906	0.99854	0.99821
Pr	0.01111	0.05996	**0.79261**	0.47231	0.41869	0.38821
F-Measure	0.02439	0.13787	0.54331	0.57737	**0.57882**	0.53133

Table 4 Results for *Shack* sequence

	GMM	KDE	SuBSENSE	Oreifej	Proposed	Proposed (No Warp)
Rc	0.68283	0.33364	0.11281	0.62405	**0.9158**	0.85846
Sp	0.96741	0.99371	**1**	0.99962	0.99807	0.99747
Pr	0.02987	0.06705	**0.74**	0.67125	0.62551	0.49107
F-Measure	0.05484	0.11101	0.27318	0.64422	**0.77993**	0.67533

Table 5 Results for *Highway 1* sequence

	GMM	KDE	SuBSENSE	Oreifej	Proposed	Proposed (No Warp)
Rc	0.71267	0.19016	0.0007	0.19802	**0.85819**	0.81419
Sp	0.94427	0.99549	**1**	0.99978	0.99773	0.99571
Pr	0.09788	0.26721	0.1	**0.87394**	0.75697	0.63811
F-Measure	0.17311	0.24324	0.01387	0.33302	**0.81476**	0.75522

and Precision but this is because it is extremely conservative in its classifications. It is clear that the SuBSENSE algorithm can detect that the noise levels are too high for it to confidently provide a classification and so often does not produce any classification at all. This means that there are no true or false positives. The GMM algorithm performs a bit better than the other two generic BS algorithms achieving similar Recall and Specificity to Oreifej et al.'s algorithm but it produces a high rate of false positives which is why its Precision scores, and thus F-Measure scores, are low. Oreifej et al.'s and the proposed algorithm, which are built for the ATD conditions, clearly fare much better and it can be seen that the proposed algorithm exhibits an excellent Recall rate compared to the others. It also produces the best F-Measure rates which show a good balance between Precision and Recall even though it never exhibits the highest Precision. In the Specificity metric the two ATD algorithms are extremely similar.

It is also apparent from the results that the approach of warping the background model towards

the current frame before performing background subtraction does indeed significantly improve the classification performance of the proposed algorithm. The most dramatic improvement in performance is in the Precision metric score. This is because this technique reduces the false positives produced at strong edges in the image as discussed in Section 3.4.

Overall our algorithm shows a balanced performance and outperforms the current state-of-the-art algorithm in terms of the F-measure. This is a significant result as the proposed algorithm makes use of very simple models and statistics to achieve this performance. Computationally we were not able to directly compare the algorithms as they are all built on different platforms but the prototype OpenCL implementation of the proposed algorithm runs at 2 FPS on a consumer desktop computer when processing a megapixel sequence without significant effort being put into optimisation. In terms of complexity the proposed algorithm is executed holding only a window of frames in memory at any given moment and consists largely of reduce operations to calculate global mean and variance and per-pixel operations which scale linearly with image size. The most costly parts of the algorithm are the optical flow calculations and the per-pixel median calculations which require a sorting process to be conducted for a window of values for each pixel. The only iterative process in the proposed algorithm is the set growing process which is dependant on the size of the regions. This is in contrast to Oreifej et al.'s approach which performs an iterative optimization on the full set of frames in a sequence which is very memory and computationally intensive when the image sizes reach the megapixel range.

To measure the stabilization performance of the ATD algorithms we measured the mean inter-frame SAD scores averaged for 100 frames of the sequence. The SAD scores are calculated as described in equation 6. The more stable a sequence is the smaller the variation in pixel values will be throughout the sequence. The results can be seen in Figure 15 and it is apparent that both

algorithms improve the stability of the sequence but that Oreifej's method produces significantly higher stability. However, this is deceptive as the background model produced by Oreifej's method is essentially a per-pixel average across all the frames in the input sequence. This results in a very high degree of blur in the output of Oreifej's technique. To demonstrate this we use the Marziliano no-reference sharpness metric[36] to measure the relative sharpness between the output of the proposed algorithm and Oreifej's algorithm. The results can be seen in Figure 16 and it is apparent that our technique produces a sharper result.

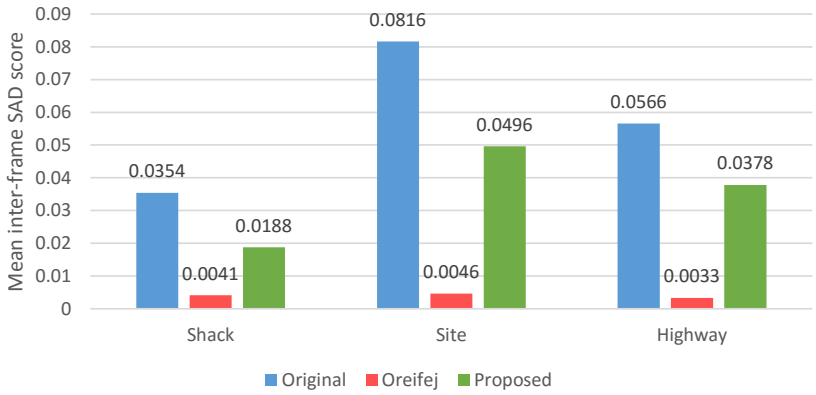


Fig 15 Stability of the original sequences compared to Oreifej's and the proposed method using mean inter-frame SAD score as a metric. Lower is better. Results are averaged over 100 frames.

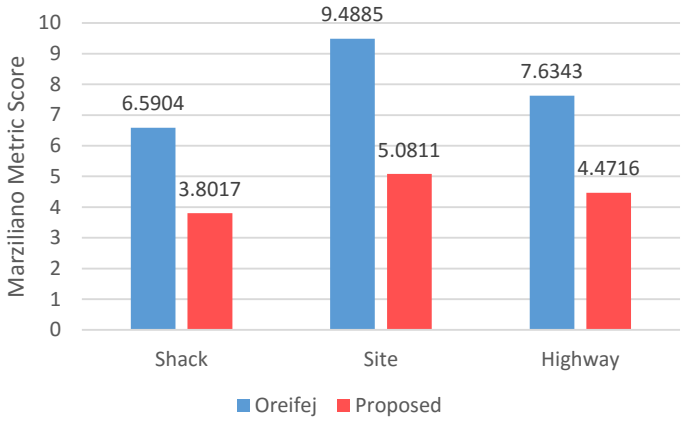


Fig 16 Sharpness of Oreifej's method and the proposed method using Marziliano's sharpness metric averaged over 100 frames. Lower scores indicate a sharper image.

5 Conclusion

In this work we investigated techniques for classifying foreground objects undergoing real motion in video sequences containing ATD. This is important for the purposes of stabilizing the geometric warping caused by ATD. If we are not able to identify regions undergoing real motion the process of correcting the apparent motion caused by ATD can distort any real motion present in the sequence.

While surveying the literature a number of approaches were found that use optical flow data and background subtraction to identify foreground regions. However these methods produce many false positives and to overcome this problem detected regions had to be tracked over multiple frames to test whether they persist in the scene. We present a new method of combining the classification power of background subtraction and optical flow data thresholded using a novel hysteresis method to detect foreground regions without the need for tracking.

To quantitatively test and compare the pixel level accuracy of our approach to those in the literature we produced a hand annotated dataset of three ATD sequences with 100 frames of ground truth data each. Using this dataset we compared our algorithm to three popular conventional background subtraction algorithms and the state-of-the-art ATD foreground classification algorithm.

The proposed algorithm exhibits excellent Recall and a balanced Precision performance and overall outperformed the current state-of-the-art solutions in foreground classification in terms of the F-measure score. This performance is achieved by a system using very simple models and statistics. We also demonstrated that the algorithm's classification was accurate enough to allow for stabilizing the background and compositing the foreground regions back into the sequence to preserve them.

Future work will entail creating algorithms that can adaptively tune the thresholding multipliers based on measured activity in the scene being stabilized. This should allow for higher accuracy across a wider range of scene types. In addition while the foreground regions can be preserved in a sequence using this method the ATD distortion is still present in these regions. Now that these regions can be detected the next step will be to correct for this distortion in these regions while preserving the real motion in the foreground regions themselves.

Acknowledgments

The authors would like to thank ARMSCOR for their financial support of this work through the LEDGER PRISM programme coordinated by the Optics and Sensor Systems group at the CSIR. We also would like to thank them for the ATD sequences that are provided as part of the PRISM programme.

References

1 C. Bondeau and E. Bourennane, "Restoration of images degraded by the atmospheric turbulence," *Fourth International Conference on Signal Processing Proceedings* **2**, 1056 –1059 (1998).

2 M. C. Roggeman and B. M. Welsh, *Imaging through turbulence*, CRC Press (1996).

3 D. H. Frakes, J. W. Monaco, and M. J. Smith, "Suppression of atmospheric turbulence in video using an adaptive control grid interpolation approach," *Proceedings of Acoustics, Speech and Signal Processing (ICASSP)* **3**, 1881–1884 (2001).

4 K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 1956–1963 (2009).

5 D. Li, R. Mersereau, and S. Simske, "Blur identification based on kurtosis minimization," *IEEE International Conference on Image Processing (ICIP)* , 905–908 (2005).

6 M. Aubailly, M. A. Vorontsov, G. W. Carhart, and M. T. Valley, "Automated video enhancement from a stream of atmospherically-distorted images: the lucky-region fusion approach," *Proc. SPIE* **7463** (2009).

7 C. S. Huebner and M. Greco, "Blind deconvolution algorithms for the restoration of atmospherically degraded imagery: a comparative analysis," *Proc. SPIE Optics in Atmospheric Propagation and Adaptive Systems XI* **7108** (2008).

8 O. Shacham, O. Haik, and Y. Yitzhaky, "Blind restoration of atmospherically degraded images by automatic best step-edge detection," *Pattern Recognition Letters* **28**(15), 2094 – 2103 (2007).

9 D. Li, R. Mersereau, D. H. Frakes, and M. J. T. Smith, "New method for suppressing optical turbulence in video," *Proceedings of European Signal Processing Conference* (2005).

10 C. J. Carrano, "Speckle imaging over horizontal paths," *Proc. SPIE* **4825**, 109–120 (2002).

11 J. Gilles, T. Dagobert, and C. Franchis, "Atmospheric turbulence restoration by diffeomorphic image registration and blind deconvolution," *Proceedings of the 10th International Conference on Advanced Concepts for Intelligent Vision Systems* , 400–409 (2008).

12 M. Shimizu, S. Yoshimura, M. Tanaka, and M. Okutomi, "Super-resolution from image sequence under influence of hot-air optical turbulence," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 1–8 (2008).

13 X. Zhu and P. Milanfar, "Removing atmospheric turbulence via space-invariant deconvolution," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **35**(1) (2013).

14 A. W. M. van Eekeren, M. C. Kruithof, K. Schutte, J. Dijk, M. van Iersel, and P. B. W. Schwering, "Patch-based local turbulence compensation in anisoplanatic conditions," *Proc. SPIE Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXIII* **8355** (2012).

15 T. Bouwmans, "Traditional and recent approaches in background modeling for foreground detection: An overview," *Computer Science Review* **11-12**, 31–66 (2014).

16 O. Oreifej, X. Li, and M. Shah, "Simultaneous video stabilization and moving object detection in turbulence," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **35**(2), 450–462 (2013).

17 J. Barron, D. Fleet, and S. Beauchemin, "Performance of optical flow techniques," *International Journal of Computer Vision* **12**(1) (1994).

18 S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *International Journal of Computer Vision* **92** (2011).

19 T. Avidor and M. Golan, "A method for removal of turbulence disturbance from video, enabling higher level applications," *6th International Conference on Image Analysis and Recognition* , 647–656 (2009).

20 B. Cohen, V. Avrin, M. Belitsky, and I. Dinstein, "Restoration of an image representing a video sequence recorded under turbulence effects," *Proc. SPIE, Applications of Digital Image Processing XX* **3164**, 535–543 (1997).

21 M. Micheli, Y. Lou, S. Soatto, and A. Bertozzi, "A linear systems approach to imaging through turbulence," *Journal of Mathematical Imaging and Vision* **48**(1), 185–201 (2014).

22 S. Gepshtein, A. Shtainman, and B. Fishbain, "Restoration of atmospheric turbulent video

containing real motion using rank filtering and elastic image registration," *Proceedings of European Signal Processing Conference* (2004).

23 L. P. Yaroslavsky, B. Fishbain, G. Shabat, and I. Ideses, "Super-resolution in turbulent videos: making profit from damage," *Optics Letters* **32**(21) (2007).

24 B. Fishbain, L. P. Yaroslavsky, I. A. Ideses, and A. Shtern, "Real-time stabilization of long-range observation system turbulent video," *Journal of Real-Time Image Processing SPIE* **2**, 11–22 (2007).

25 C. S. Huebner, "Turbulence mitigation of short exposure image data using motion detection and background segmentation," *Proc. SPIE Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXIII* **8355** (2012).

26 M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging* **13**(1), 146–165 (2004).

27 Y. Yitzhaky, I. Dror, and N. S. Kopeika, "Restoration of atmospherically blurred images using weather-predicted atmospheric modulation transfer function (mft)," in *Proc. SPIE Image Propagation through the Atmosphere*, **2828**, 386–396 (1996).

28 N. McFarlane and C. Schofield, "Segmentation and tracking of piglets in images," *Machine Vision and Applications* **8**(3), 187–193 (1995).

29 A. Elkabetz and Y. Yitzhaky, "Background modeling for moving object detection in long-distance imaging through turbulent medium," *Applied Optics* **53**(6), 1132–1141 (2014).

30 E. Chen, O. Haik, and Y. Yitzhaky, "Detecting and tracking moving objects in long-distance imaging through turbulent medium," *Applied Optics* **53**(6), 1181–1190 (2014).

31 A. Apuroop, A. S. Deshmukh, and S. S. Medasani, "Robust tracking of objects through turbulence," *Proceedings of the 2014 Indian Conference on Computer Vision Graphics and Image Processing* , 29:1–29:8 (2014).

32 C. S. Huebner and C. Scheifling, "Software-based mitigation of image degradation due to atmospheric turbulence," *Proc. SPIE Optics in Atmospheric Propagation and Adaptive Systems XIII* **7828** (2010).

33 T. Bouwmans, "Recent advanced statistical background modeling for foreground detection: A systematic survey," *Recent Patents on Computer Science* **4**(3), 147–171 (2011).

34 O. Haik and Y. Yitzhaky, "Effects of image restoration on automatic acquisition of moving objects in thermal video sequences degraded by the atmosphere," *Applied Optics* **46**(36) (2007).

35 S. DelMarco and S. Agaian, "The design of wavelets for image enhancement and target detection," *Proc. SPIE Mobile Multimedia/Image Processing, Security, and Applications* **7351** (2009).

36 P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "A no-reference perceptual blur metric," *Proceedings of International Conference on Image Processing* (2002).

37 D. Sun, S. Roth, J. Lewis, and M. Black, "Learning optical flow," *Proceedings of European Conference on Computer Vision (ECCV)* (2008).

38 G. Farneback, "Two-frame motion estimation based on polynomial expansion," *Lecture Notes in Computer Science* , 363–370 (2003).

39 C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV-L1 optical flow," *Proceedings of Pattern Recognition (DAGM)* , 214–223 (2007).

40 P. E. Robinson and W. J. Lau, "Adaptive multi-scale retinex algorithm for contrast enhancement of real world scenes," *Proceedings of the 23rd Annual Symposium of the Pattern Recognition Association of South Africa (PRASA)* , 60–67 (2012).

41 P. E. Robinson and Y. Roodt, "Blind deconvolution of Gaussian blurred images containing additive white Gaussian noise," *IEEE International Conference on Industrial Technology (ICIT)* (2013).

42 P. E. Robinson, "Project website for "Foreground segmentation in atmospheric turbulence degraded video sequences to aid in background stabilization" article," (2016). http://www.hypervision.co.za/index.php/foreground-segmentation-atd-sequences/.

43 J. Canny, "A computational approach to edge detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **8**(6), 679–698 (1986).

44 C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* **2** (1999).

45 A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *Proceedings of European Conference on Computer Vision (ECCV)*, *Lecture Notes in Computer Science* **1843**, 751–767, Springer Berlin Heidelberg (2000).

46 P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, "Flexible background subtraction with self-balanced local sensitivity," *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* , 414–419 (2014).

47 Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "CDnet 2014: An expanded change detection benchmark dataset," *Proc. IEEE Workshop on Change Detection (CDW-2014) at CVPR-2014* , 387–394 (2014).

48 A. Sobral, "BGSLibrary: An OpenCV C++ Background Subtraction Library," *IX Workshop de Viso Computacional (WVC'2013)* (2013).

49 CSIR, "PRISM Datasets," (2016). Accessed: 3 March 2016, http://prism.csir.co.za/.

50 D. Powers, "Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation," *Journal of Machine Learning Technologies* **2**(1), 37–63 (2011).

Philip E. Robinson is a lecturer in electrical and electronic engineering at the University of Johannesburg. His current research interests are Image and Video Enhancement, Automated Surveillance Systems and Robotics.

André L. Nel is a professor of mechanical engineering at the University of Johannesburg. He has spent 25 years in tertiary education and his current research interests lie in image processing, CFD and robotic locomotion and control systems.

List of Figures

6 Illustrating background subtraction of a lamp post in *Highway* sequence. (a) Background Model, (b) Current Frame, (c) Absolute Difference

7 Illustration of background subtraction result for a frame in the *Site* sequence with and without prior warping of the background model (constructed from 50 prior frames) towards the current frame, a foreground region is indicated in red. (a) Background subtraction result with no prior warping. (b) Background subtraction result with prior warping.

8 Mean BS response values averaged over 100 frames for the three ATD sequences with and without prior warping

9 Normalized per-pixel median of the absolute difference values for 50 frames of the *Highway* sequence illustrating the high persistent response in areas with a high degree of foreground traffic

10 Illustration of the classification results using a global threshold calculated with high and low multiplier values ($K=10$ and $K=3$ respectively) and the result of the *Grow From Intersection* operation. (a) Classification using a high global threshold value ($K=10$). (b) Classification using a low global threshold value ($K=3$). (c) Classification from *Grow From Intersection* operation.

11 Illustration of the ensemble of thresholding operations and the combination of the resulting sets that make up the proposed classification scheme

12 A frame from the *Site* sequence

13 A frame from the *Highway* sequence

14 A frame from the *Shack* sequence

List of Tables