

Evaluation of Turbulence Mitigation Methods

Adam W.M. van Eekeren^{a*}, Claudia S. Huebner^{b†}, Judith Dijk^a, Klamer Schutte^a,
Piet B.W. Schwering^a

^aTNO, PO Box 96864, 2509 JG The Hague, The Netherlands

^bFraunhofer IOSB, Gutleuthausstrasse 1, 76275 Ettlingen, Germany

ABSTRACT

Atmospheric turbulence is a well-known phenomenon that diminishes the recognition range in visual and infrared image sequences. There exist many different methods to compensate for the effects of turbulence. This paper focuses on the performance of two software-based methods to mitigate the effects of low- and medium turbulence conditions. Both methods are capable of processing static and dynamic scenes. The first method consists of local registration, frame selection, blur estimation and deconvolution. The second method consists of local motion compensation, fore-/background segmentation and weighted iterative blind deconvolution. A comparative evaluation using quantitative measures is done on some representative sequences captured during a NATO SET 165 trial in Dayton. The amount of blurring and tilt in the imagery seem to be relevant measures for such an evaluation. It is shown that both methods improve the imagery by reducing the blurring and tilt and therefore enlarge the recognition range. Furthermore, results of a recognition experiment using simulated data are presented that show that turbulence mitigation using the first method improves the recognition range up to 25% for an operational optical system.

Keywords: Turbulence mitigation, deconvolution, image restoration, quantitative evaluation

1. INTRODUCTION

In general, long range detection, recognition and identification in visual and infrared (video) imagery are hampered by turbulence caused by atmospheric conditions. Especially over long ground-to-ground ranges atmospheric effects are dominant. These atmospheric effects result in blurry and shaky images, which can be reduced by turbulence mitigation methods. An extensive statistical analysis of the turbulence effects scintillation, blur and beam wander over horizontal paths in a littoral environment are presented in de Jong et al.¹. Relevance of turbulence effects at infrared wavelength is provided in a statistical study by Schwering and Kunz². Turbulence mitigation is a challenging problem because 1) the amount of turbulence is spatially varying and can be assumed constant only in a small local neighborhood and 2) the amount of turbulence varies significantly over time.

Typical approaches to mitigate the effects of turbulence are described in^{3,4}. Basically there exist three approaches: 1) a hardware approach, 2) a software approach and 3) a combination of both approaches. In this paper we will focus on two software approaches that are capable of processing static and dynamic scenes. The first method⁵ consists of local registration, frame selection, blur estimation and deconvolution. The second method^{6,7} consists of local motion compensation, fore-/background segmentation and weighted iterative blind deconvolution. The main advantages of a software approach are that it is flexible, cheap and effective, especially given the processing power available nowadays.

In this paper we will describe the two turbulence mitigation methods and we will quantitatively evaluate their image quality improvement. This evaluation is done on some representative sequences containing turbulence captured during a NATO SET 165 trial in Dayton in 2011⁸. The amount of blurring and tilt in the imagery are relevant measures for such an evaluation. These general, basic measures give a good indication of the image quality improvement while no specific target in the scene is needed such as the Triangle Orientation Discrimination (TOD) method⁹ and the Minimum Resolvable Temperature Difference (MRTD) method¹⁰. It is shown that both turbulence mitigation methods improve

* adam.vaneeekeren@tno.nl; www.tno.nl

† claudia.huebner@iosb.fraunhofer.de; www.iosb.fraunhofer.de

image quality significantly. Furthermore, simulation results show that turbulence mitigation using the first method improves the recognition range of an operational optical system between 10% and 25%.

The setup of the paper is as follows. Section 2 and 3 describe the two software-based turbulence mitigation methods. In Section 4 both methods are evaluated using quantitative quality measures on real-world data. The impact on the recognition range of turbulence mitigation is presented in Section 5. Finally, conclusions will be drawn in Section 6.

2. TURBULENCE MITIGATION METHOD A

This section describes the turbulence mitigation method developed by TNO and it is referred to in this paper as ‘method A’.

2.1 Imaging model

The imaging model which is used by method A assumes that the camera’s field-of-view, *the scene*, is modeled as a properly sampled 2D HR image \mathbf{z} (see Figure 1). It is assumed that this HR image is free of degradation to motion, blur and noise. The output frames \mathbf{y}_k are a result of the effects caused by atmospheric turbulence and the camera itself on *the scene*.

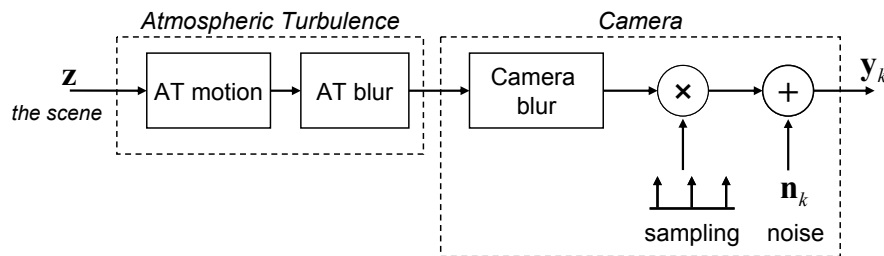


Figure 1. Flow diagram illustrating the degradation of a properly sampled pristine scene \mathbf{z} into a turbulence and camera deteriorated image \mathbf{y}_k .

The degradation of camera frame \mathbf{y}_k can be split up in two parts: a part that is due to the atmospheric turbulence and a part which is caused by the camera.

Atmospheric turbulence causes blurring, transformation (motion) and scintillation in the observed image of the scene. All these phenomena are assumed to be constant in a small region (patch) within the field-of-view. The size of such a region is determined by the isoplanatic angle, which is from a theoretical perspective, for a horizontal path through the atmosphere, often smaller than even a pixel. For the blurring component we simplified the model by assuming that it is only temporally varying, which is described by a Gaussian function with standard deviation $\sigma_t(t)$. The transformation which is caused by turbulence is locally modeled by a translation. In this paper scintillation, i.e. the effect of illuminance variation, is not modeled.

The camera is modeled by a ‘camera blur’ (optics and sensor), a sampling device and additive Gaussian distributed noise. The camera blur is modeled by a Gaussian function with standard deviation σ_c . In this paper it is assumed that the camera blur is small compared to the turbulence blur. Therefore the overall blur is assumed to be a Gaussian with standard deviation $\sigma_{tot} \approx \sigma_t$.

The imaging model as depicted in Figure 1 can be described by the following equation:

$$\mathbf{y}_k = \mathbf{R}\mathbf{W}_k\mathbf{M}_k\mathbf{z} + \mathbf{n}_k \quad (1)$$

where \mathbf{M}_k represents the transformation matrix, \mathbf{W}_k describes the blurring matrix and \mathbf{R} is the resampling matrix.

2.2 Processing steps method A

This section describes the turbulence mitigation method A. An overview of this method is depicted in Figure 2 and previously described in⁵. The different steps are explained in more detail in the subsections below.

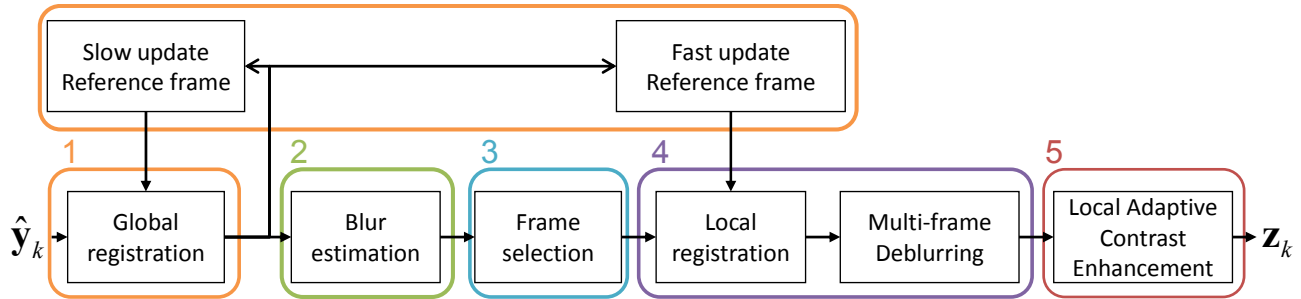


Figure 2. Flow diagram of turbulence mitigation method A.

2.3 Global registration and slow/fast update reference frame

To stabilize the global structure of the scene a global registration¹¹ is performed assuming a translational motion model. This registration is done to a reference frame, which is slowly updated with registered input frames. With “slowly” we mean an update factor α in the order of 0.01, according to:

$$\mu_i = (1 - \alpha)\mu_{i-1} + \alpha y_{stab}, \quad (2)$$

where μ_i is the reference frame at time i and y_{stab} is a stabilized input frame.

For performing local registration (in step 4) a fast updated reference frame is used. Here an update factor in the order of 0.1 is used. This fast update enables us to process slow moving objects in the scene. Both reference frames are initialized with the first frame of the sequence and are updated with registered input frames.

2.4 Blur estimation

In the second step the mean blur of each registered frame is estimated. This blur estimation is used later on (step 4) to perform the multi-frame deblurring. The method from Bouma et al.¹² is used which gives a precise and un-biased result.

2.5 Frame selection

The third step in the flow diagram of the proposed method is frame selection. This is done by calculating the 99th percentile of the Gaussian gradient magnitude of each registered frame within a window of N_l frames and afterwards selecting N_s frames with the highest percentiles. This results in an output sequence which is sharper and less fluctuating in sharpness compared to an output sequence without frame selection. Typical parameter setting ranges are $N_s = [10..50]$ and $N_l = [20..75]$ for frame rates between 30 and 70 Hz.

2.6 Local registration and Multi-frame Deblurring

The fourth step consists of local registration and multi-frame deblurring. Local registration is done using optical flow¹³. Optical flow estimates a per pixel local translation. To speed up computation of the optical flow first local translations are calculated at a sub-sampled and equally spaced grid in the image. From this grid the optical flow is interpolated. Typical parameter settings used are a grid space of 40 pixels and a window size of 64×64 pixels.

For multi-frame deblurring the method of Hardie et al.^{14,15} is used. The main difference is that our method uses optical flow registration while Hardie et al. uses global translation as registration. Equal to Hardie et al. the blurring is assumed spatially constant. For implementation reasons the mean blur of the selected N_s frames is used in the optimization. To estimate the scene at frame k (Z_k), the following cost function is minimized:

$$C(\mathbf{z}_k) = \sum_m^{N_s} (\hat{\mathbf{y}}_m - \mathbf{R}\mathbf{W}_k\mathbf{M}_{k,m}\mathbf{z}_k)^2 + \lambda_r \sum_n (L(\mathbf{z}_k))^2, \quad (3)$$

where $\hat{\mathbf{y}}_m$ are N_s observed frames used for deblurring, $\mathbf{M}_{k,m}$ represents the transformation matrix containing the global and local registration, \mathbf{W}_k describes the blurring matrix and \mathbf{R} is the resampling matrix. The final part of the cost function is the so-called regularization part which forces a smoothness constraint on the solution. Here λ_r is a weighting factor and L indicates a Laplacian operator. The cost function is minimized using a conjugate-gradient optimization. Typical parameter setting ranges for the deblurring part of our method are: resampling (upsampling) factor = [1..4], $\lambda_r = [10^{-3} \dots 10^{-1}]$ and [5..10] iterations for the conjugate-gradient minimization procedure.

2.7 Local Adaptive Contrast Enhancement

The last step in the flow diagram of the proposed method is local adaptive contrast enhancement (LACE)¹⁶. This method enhances the contrast locally based on the local variation in the scene.

3. TURBULENCE MITIGATION METHOD B

This section describes the turbulence mitigation method developed by the IOSB and it is referred to in this paper as ‘method B’.

When dealing with atmospherically degraded and noisy data it often is practically unavoidable to apply at least some kind of averaging procedure which has become an integral part of various turbulence compensation schemes for this very reason. Consequentially, occurring motion blur, either real or created by averaging, is amplified proportionately to the number of stacked images. Therefore, in this method motion detection by using block-matching is employed in order to separate directed object motion from undirected movement due to turbulence. This way, a segmentation of static scene elements and moving objects is possible, provided that the object movement exceeds the false motion caused by the atmospheric turbulence, and image restoration can be carried out for static and moving elements, independently. Local image stacking is used to help correct for directed object motion, effectively reducing motion artefacts and improving the overall restoration result, while motion compensating image integration is applied to the static background.

The turbulence compensation algorithm ‘method B’ consists of the following steps:

1. Pre-processing of the input data (all optional)
 - a. Image quality assessment (edge-based metric)
 - b. Frame integration
 - c. Contrast enhancement (CLAHE)
2. Motion compensation
 - a. Global motion compensation / Image registration
 - b. Algorithm
 - c. Local motion compensation
3. Motion detection and estimation (only if motion content in scene is high)
 - a. Local image stacking
 - b. Background/foreground segmentation
4. Weighted iterative blind deconvolution

3.1 Pre-Processing

Depending on the severity of image degradation caused by atmospheric turbulence, some form of pre-processing of the data may be advisable before executing the main processing steps. The following pre-processing steps are all optional and dependent on the quality of the input data.

3.1.1 Image Quality Assessment

Optionally, image quality can be assessed in accordance with the Lucky Shot principle by means of either variance- or edge-based quality metrics¹⁷. Then a given percentage of extremely degraded frames can be discarded entirely (similar to the frame selection described in section 2.5).

Naturally, this works best on scenes containing little or no object movement and if the frame-rate is sufficiently high (e.g. ≥ 70 Hz). In the results presented here, image quality was evaluated in selected regions of interest (ROIs) using the sum of edges in the Sobel-filtered image. Typically, the percentage of selected frames lies between 50% (frame-rate ≥ 500 Hz) and 90% (frame-rate ≤ 50 Hz). Here, this was only used in the two sequences ‘10h56h41s_two-people’ and ‘16h19m46s_stadium_lighting_200mm’, depicted in Figure 5 and Figure 8, respectively. Given the high frame-rate and large number of frames available, in both cases the best 50% frames were selected.

3.1.2 Frame Integration

In case of severely degraded and noisy data, the success rate of the subsequent motion detection improves, when the average of every couple of frames is taken, thereby reducing image distortions and noise⁷. This works best for short exposures and high frame-rates.

This was only done in the case of the ‘10h56h41s_two-people’-sequence (see Figure 5), using an average of every 3 frames (i.e. ≈ 6 frames, given only 50% of the frames were selected).

3.1.3 Contrast Enhancement

Additionally, depending on the quality and contrast of the input data, some prior contrast enhancement, e.g. according to the CLAHE-algorithm¹⁸ (Contrast Limited Adaptive Histogram Equalization), can improve the results when applied as pre-processing step (also as a final post-processing step).

Since contrast was low in all sequences presented here, some minor CLAHE contrast enhancement was applied in all cases except the sequence ‘16h14m51s_top-antenna-66mm’ shown in Figure 7.

3.2 Global Motion Compensation

This first main processing step (aka image registration) is an inherent part of method B. Here, the current frame of the input sequence is shifted slightly within a given search space of N pixels in every direction such that the input frame best matches a given reference image. Ideally, this reference would be an image unimpaired by atmosphere and optical system. Since normally such an ideal image will not be available, a moving average presents a simple and reasonable substitute. Other alternatives include the temporal median or the temporal modal value of the sequence.

The results are written into a long-term buffer of length W which gets updated with every iteration. This buffer is used for calculating the reference image $R^{(n)}$ used in image registration and equally for the ensuing background segmentation. Typically, W is comparatively large, depending on the severity of turbulence degradation and, of course, on the frame-rate. In the examples presented here, $W=100$.

If necessary, i.e. in strong turbulence conditions, local motion compensation by using block-matching⁶ can also be applied as an alternative to the motion detection procedure described below. This works best if directed object motion in the scene is small.

3.2.1 Algorithm

Let $I^{(n)}$ and $I^{(n+1)}$ denote the n -th and $n+1$ -st input image, respectively, $\underline{x} = (x, y)$ the pixel coordinates, N the size of the search space and N_{max} the total number of shift directions such that $\underline{M} \in \mathbb{Z} \times \mathbb{Z}$, $\underline{M}(\underline{x}) = (\delta x, \delta y)$ denotes the vector containing all possible integer displacements $\delta x, \delta y \in \{-N, -N-1, \dots, -1, 0, +1, +2, \dots, +N\}$.

The N_{max} shift images $S_i^{(n)}$ for frame $I^{(n)}$ are then given by:

$$S_i^{(n)}(\underline{x}) = I^{(n)}(\underline{x} + \underline{M}(\underline{x})), \quad (i = 1, \dots, N_{max}) \quad (4)$$

The reference (here: moving average) image $R^{(n)}$ with window length W is given by:

$$R^{(n)}(\mathbf{x}) = \frac{1}{W} \sum_{k=1}^W I^{(n-k)}(\mathbf{x}) \quad (5)$$

The registered image $I_{reg}^{(n)}$ is obtained by determining the best match between $R^{(n)}$ and shift images $S_i^{(n)}$ using either normalized cross-correlation or simple error summation methods as similarity measure, e.g. Root Mean Squared Error (RMSE). The index idx of the shift image $S_{idx}^{(n)}$ matching $R^{(n)}$ best,

$$idx = \underset{i \in \{1, \dots, Nmax\}}{\text{Best match}} \left(R^{(n)}, S_i^{(n)} \right) \quad (6)$$

yields in turn the registered image $I_{reg}^{(n)} = S_{idx}^{(n)}$ as well as the corresponding global motion vector $(\delta x(idx), \delta y(idx))$.

3.3 Motion Detection and Estimation

In case of moving objects in the observed scene, motion detection and estimation, as detailed in ⁷ and ¹⁹, constitute the next steps in the algorithm. The procedure employs block-matching (BM) for the segmentation of foreground and background, i.e. moving objects and static scene elements. At the same time, BM estimates motion vectors between consecutive frames for use in the successive local image stacking.

Here, the complete motion detection procedure as described in the following was only applied to the sequence ‘10h56h41s_two-people’ to be seen in Figure 5.

3.3.1 Algorithm

In this case, the best match between consecutive images $I^{(n)}$ and $I^{(n+1)}$ is determined. For every pixel \underline{x} , the corresponding index $\text{Idx}(\underline{x})$ of the best matching block is obtained by block-wise implementation of:

$$\text{Idx}(\underline{x}) = \underset{i \in \{1, \dots, Nmax\}}{\text{Best match}} \left(I^{(n+1)}(\underline{x}), S_i^{(n)}(\underline{x}) \right) \quad (7)$$

Analogue to Section 3.2.1, this yields the corresponding vector field with local motion vectors $(\delta x(\text{Idx}(\underline{x})), \delta y(\text{Idx}(\underline{x})))$.

3.3.2 Local Image Stacking (LIS)

In this step, as described in ⁷, short-term buffers, typically holding between 5 and 10 frames (depending on the frame-rate and the amount of motion present), are used for the images as well as for the corresponding motion vectors calculated for every pair of consecutive frames. The images $I^{(1)}, \dots, I^{(n)}$ of this buffer are stacked in inverse order. Each pixel block of image $I^{(n)}$ is shifted according to the displacements given by its corresponding motion vector, in order to best match the current position, and added to the matching (not shifted) block in the previous frame $I^{(n-1)}$. This step is repeated, shifting each block of the result to match the frame before, $I^{(n-2)}$, until the first element in the buffer is reached. The LIS underlying principle is illustrated in the following Figure 3.

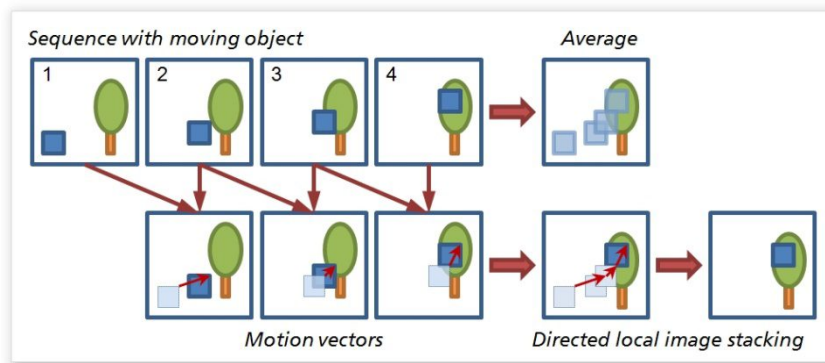


Figure 3. Illustration of Local Image Stacking (LIS) principle.

3.3.3 Background Segmentation

For background segmentation only minor changes need to be made to the algorithm. Instead of block-wise matching consecutive images $I^{(n)}$ and $I^{(n+1)}$, $I^{(n+1)}$ is again matched to reference image $R^{(n)}$. Additionally, a ‘zero motion’ threshold²⁰ is used, which forces the motion vector of the corresponding block to be zero if the similarity measure employed in the matching process falls below this threshold, simultaneously forcing non-zero motion vectors everywhere else.

A basic mask is created where the blocks with zero-motion vectors form the background while the remaining blocks constitute the foreground. The static blocks then are taken from the long-term reference image while the blocks containing motion are taken from the LIS-result. In order to improve the contours of this simple mask, image differencing is also used as described in¹⁹.

3.4 Weighted Iterative Blind Deconvolution (WIBD)

The final image restoration is carried out for static and moving elements, independently. In this work, a variation of the Iterative Blind Deconvolution (IBD) algorithm²¹ as proposed in⁶ was chosen for this image restoration. It mainly consists of a concatenation of several ordinary deconvolution steps with varying PSF support size and iteration number. Additionally, the Sobel-filtered reference image is used as weighting function in order to amplify the deconvolution effect around edges while reducing ringing effects in homogeneous image regions.

Afterwards, the two distinct deconvolution results are recombined into a composite image employing the mask created in the previous section.

4. EVALUATION OF METHOD A AND B ON REAL-WORLD DATA

4.1 Real-world data

Both turbulence mitigation methods are evaluated on some representative sequences containing turbulence artefacts. These sequences are part of the Dayton dataset that is captured during a NATO SET-165 trial in Dayton, USA, beginning of October 2011⁸. The distance over which the imagery is captured, is approximately 7 km. An example of a measurement setup is depicted in Figure 4. The camera used is the monochrome AVT Stingray F-080B with a maximum frame size of 1032×778 pixels and a pixel depth of 14-bit. For the optics a Celestron C8 is used.

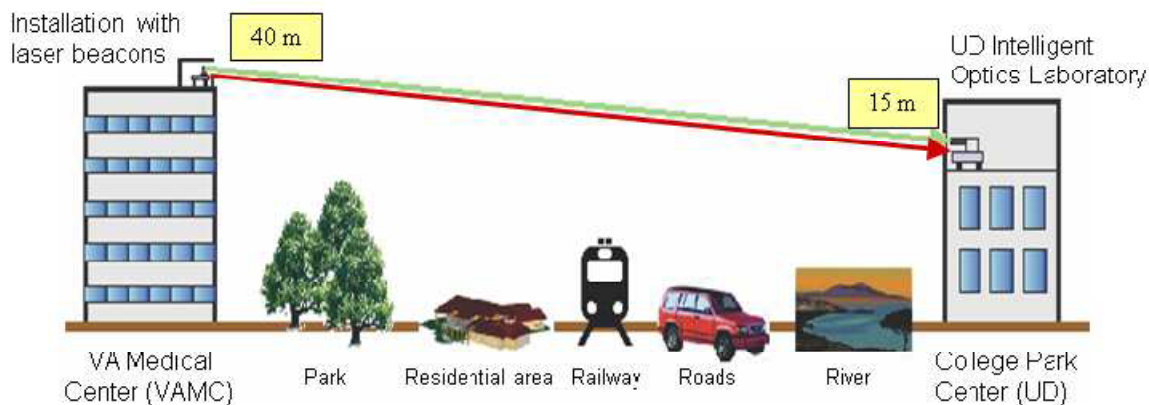


Figure 4. Graphical representation of the 7 km long atmospheric propagation path used for capturing some of the datasets.

From all the imagery captured during the trail, we selected and processed 4 image sequences of different scenes (one dynamic and 3 static) under varying turbulence conditions.

To evaluate the image quality improvement of the proposed turbulence mitigation methods the following measures are calculated: blur and tilt. Because the available imagery does not contain special evaluation targets, such as TOD / MRTD charts, measures are selected that do not depend on such targets.

4.2 Blur

The mean blur in each frame is measured with the method described by¹². For each frame the mean blur is reported as a Gaussian sigma in pixels (Gaussian blurring is assumed as indicated in Section 2.1). This mean blur is calculated in the original frames and in the processed frames as well. In this way an improvement / reduction of the blur can be calculated.

4.3 Tilt

The tilt is a measure indicating the amount of movement in the scene. This can be caused by camera movement, turbulence and moving objects in the scene. In the data we processed turbulence is the most prominent factor causing the tilt. The tilt is calculated with a gradient-based shift estimator¹¹ and reported as the length of the shift vector in pixels. Equal to the blur, the tilt is calculated for the original and processed images.

4.4 Results

Some visual results of a single image from the 4 sequences are provided in the figures below (Figure 5, Figure 6, Figure 7 and Figure 8).

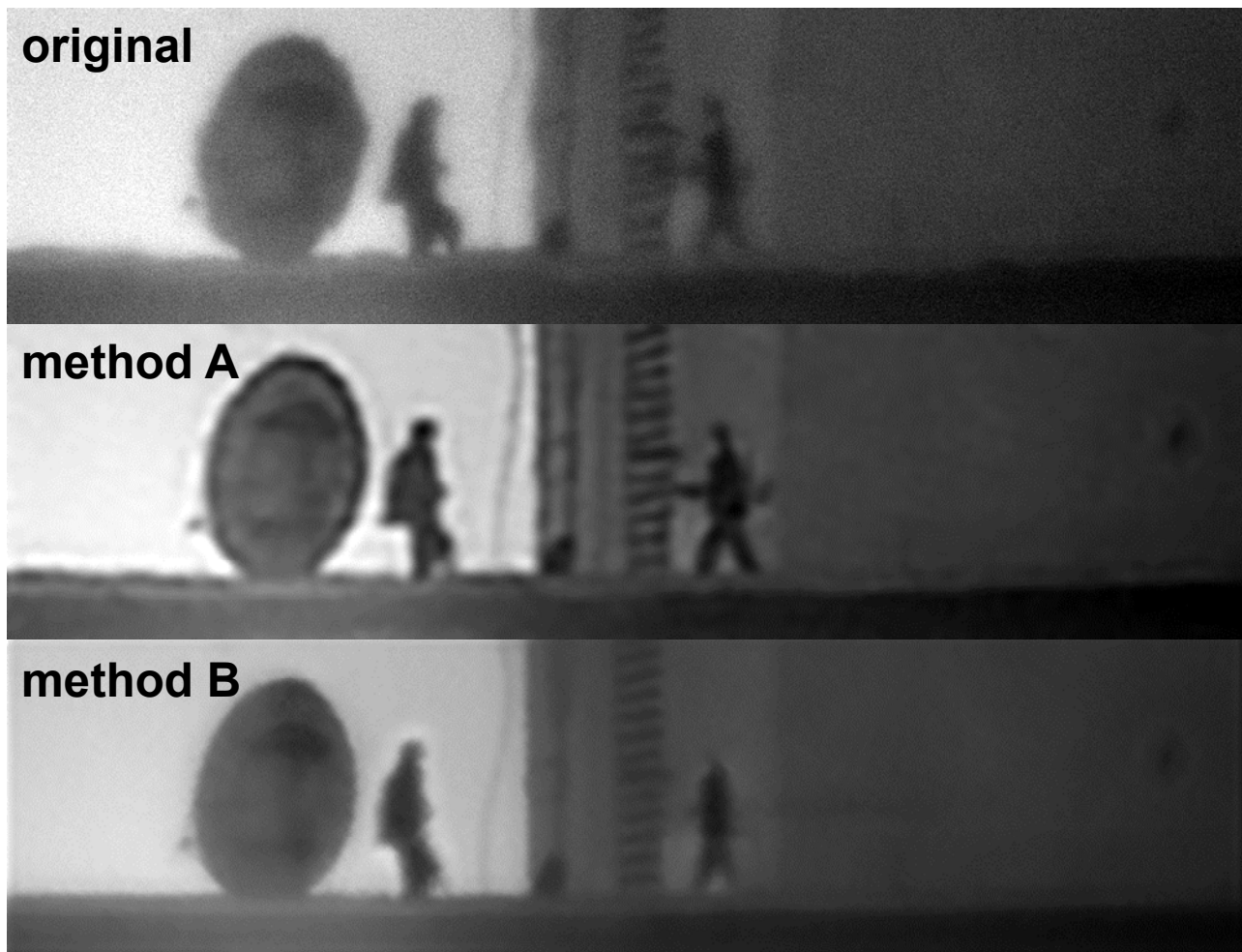


Figure 5. Visual comparison of an original image of sequence '10h56h41s_two-people' with the mitigation results of method A and B.

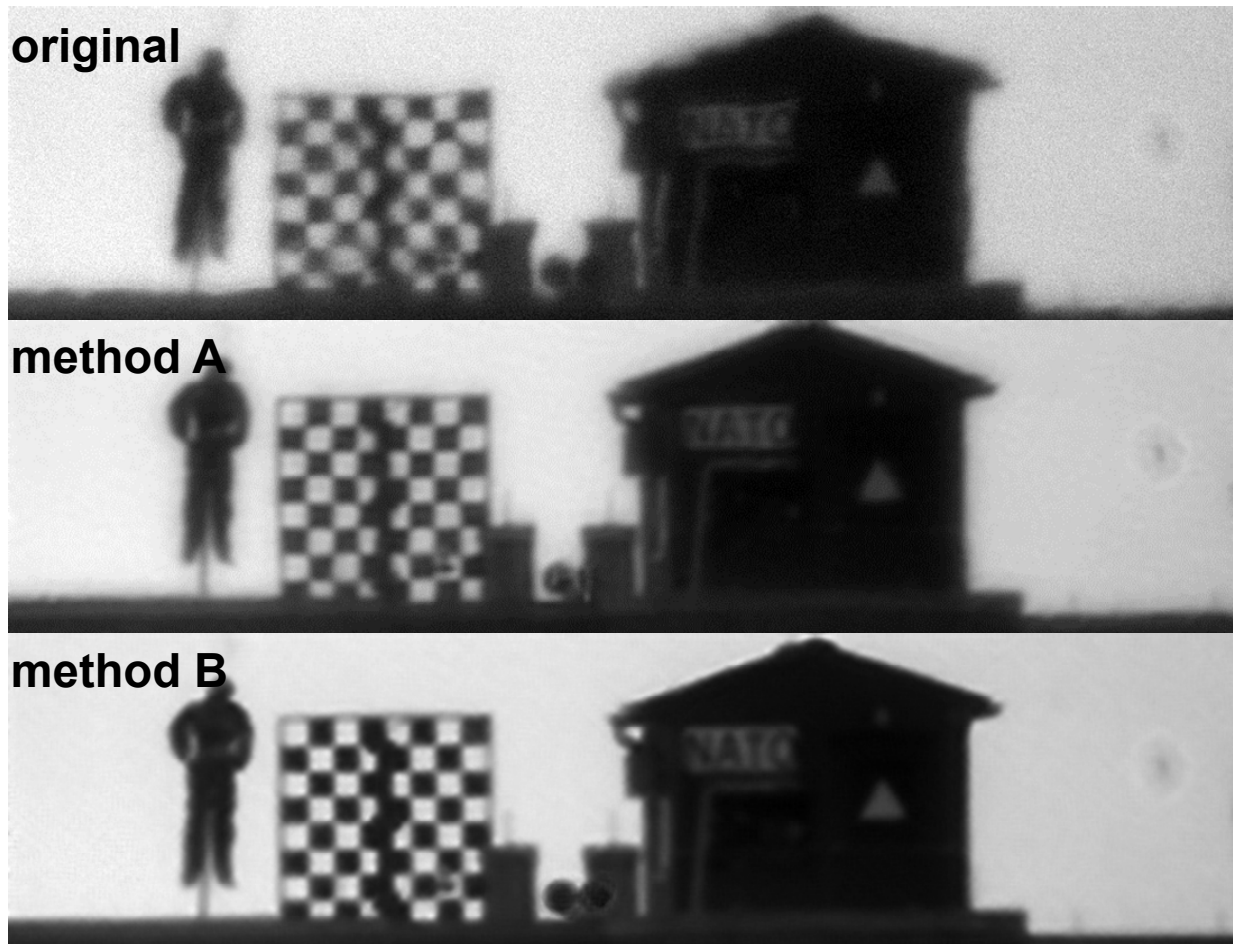


Figure 6. Visual comparison of an original image of sequence '15h34h37s_nato' with the mitigation results of method A and B.

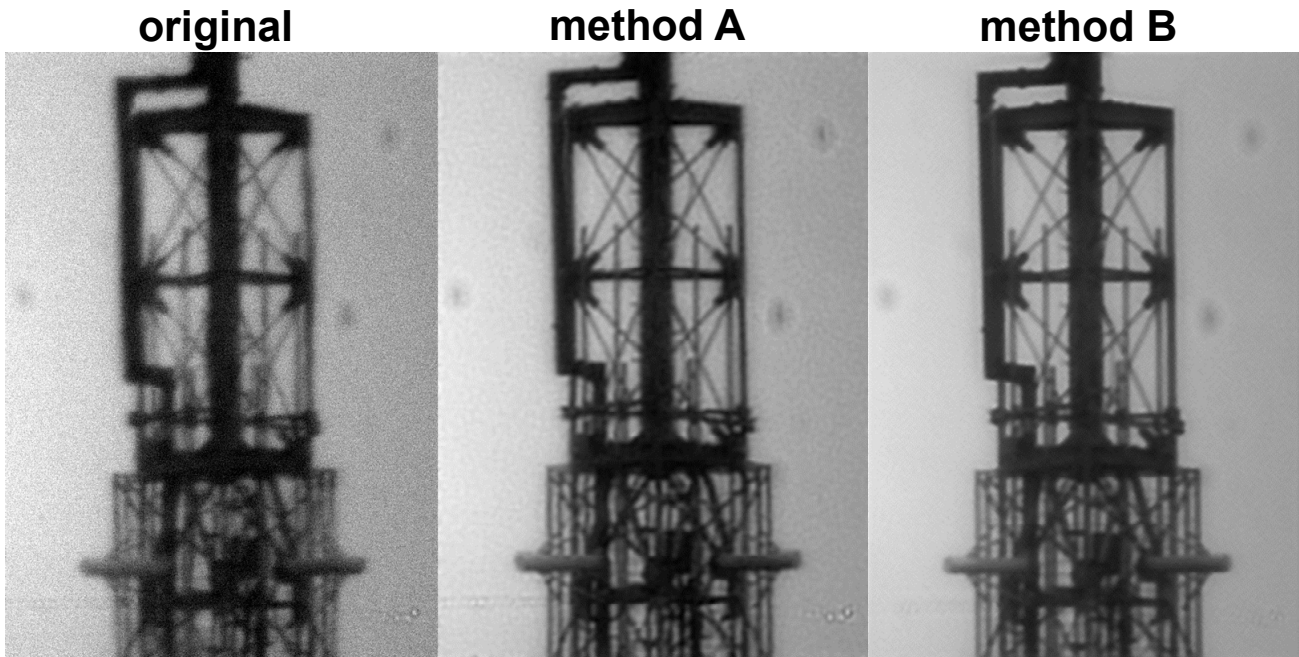


Figure 7. Visual comparison of an original image of sequence '16h14m51s_top-antenna-66mm' with the mitigation results of method A and B.

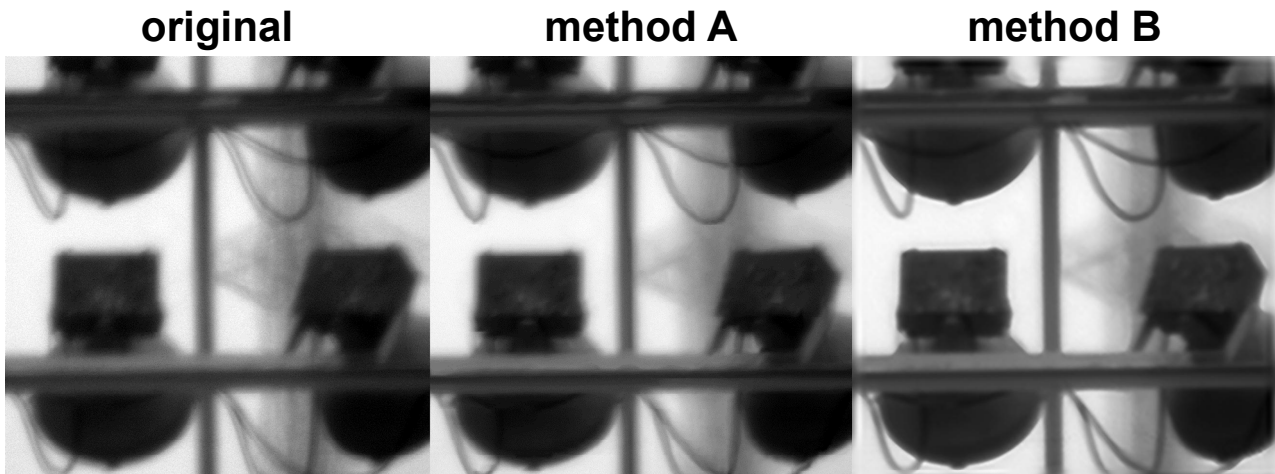


Figure 8. Visual comparison of an original image of sequence '16h19m46s_stadium_lighting_200mm' with the mitigation results of method A and B.

In Table 1 we present a quantitative comparison of the blur and tilt values of the original image, with the ones after processing with methods A and B. The blur size in the table is represented by the standard deviation of the Gaussian in pixels. Both methods show improvements in blur. Both methods also show small residual tilt at sub-pixel level. The methods are established with different philosophies, such as motion robustness and calculation speed for implementation in real-time. These results give insight into ways ahead, which could be in adapting the algorithm based on preliminary turbulence estimates in these low to medium turbulence conditions. Additionally, different approaches might have to be developed for deep turbulence.

Table 1. Quantitative comparison of method A and B.

Name sequence	BLUR (pixels)			TILT (pixels)		
	Original	Method A	Method B	Original	Method A	Method B
10h56m41s_two_people	4.88	2.99	3.55	1.80	0.45	0.41
15h34m37s_nato	4.45	2.92	3.03	3.09	0.02	0.01
16h14m51s_top_antenna_66mm	4.24	2.58	1.94	2.63	0.01	0.01
16h19m46s_stadium_lighting_200mm	7.29	4.95	3.58	2.55	0.01	0.01

5. EVALUATION OF METHOD A ON SIMULATED DATA

To measure the impact of a turbulence mitigation algorithm on the recognition range the following experiment has been done. Using an artificial scene, the 1951 USAF resolution test chart, different amounts of turbulence at varying distances are simulated. Three turbulence conditions are simulated: weak ($C_N^2 = 5.10^{-16} \text{ m}^{-2/3}$), moderate ($C_N^2 = 5.10^{-14} \text{ m}^{-2/3}$) and strong ($C_N^2 = 5.10^{-12} \text{ m}^{-2/3}$). The distance is varied between 3 and 11 km, so we are focussing on a long range scenario. As sensor a visual Near Field-Of-View(FOV) optical system is assumed with the following specifications: wavelength of 0.4-0.9 μm , 768x576 pixels, 2x1.5 degree FOV.

For each ‘turbulence scenario’ (turbulence condition and range) a set of 200 turbulence degraded images is simulated using the method described in ²². These images are then processed with turbulence mitigation method A. Examples of a few turbulence degraded images with the mitigated results are depicted in Figure 9 and in Figure 10.

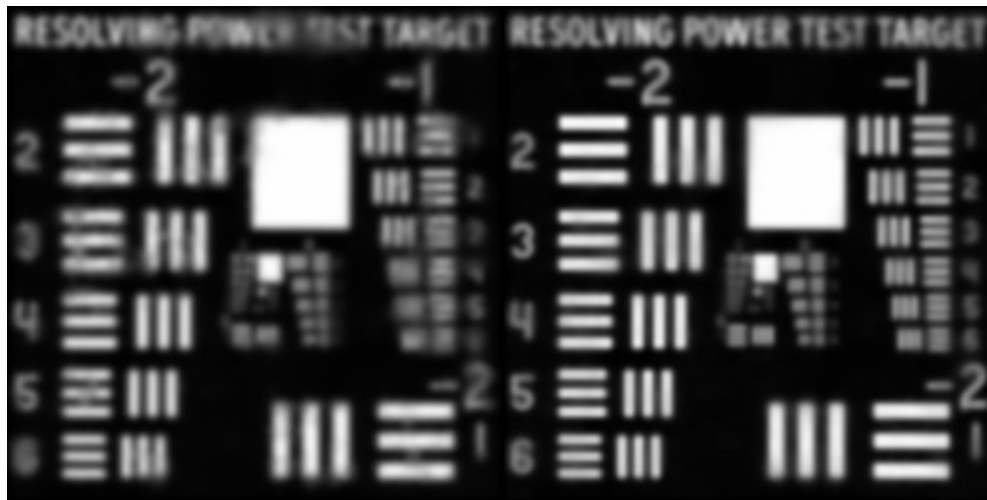


Figure 9. Left: turbulence degraded scene at 5 km under weak turbulence conditions. Right: turbulence mitigated result of same scene.

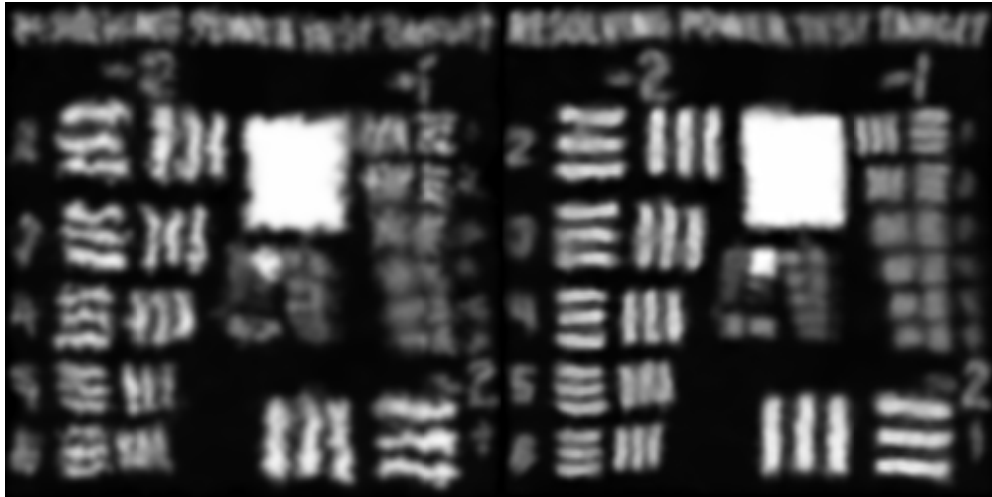


Figure 10. Left: turbulence degraded scene at 3 km under strong turbulence conditions. Right: turbulence mitigated result of same scene.

After processing all the images of the different ‘turbulence settings’, the recognition range is determined by viewing each set of images (original and processed) as a video at 25 Hz and by visually indicating for which ‘bar pattern’ the 3 line pairs are just visible. From these results a plot of the recognition range versus the target size can be made for each turbulence condition. See Figure 11, from which it is clear that turbulence mitigation increases the recognition range (blue line versus red line). An overview of the improvement in recognition range for two different target sizes (3x3m and 5x5m) is presented in Table 2.

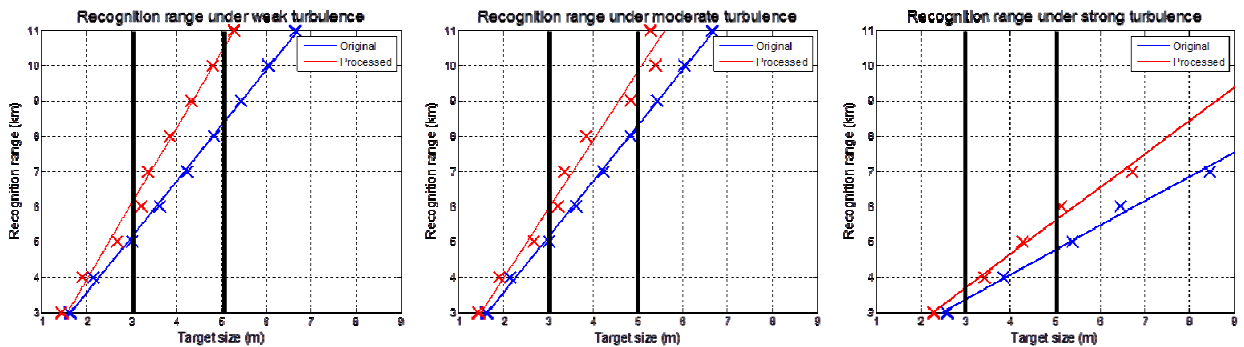


Figure 11. Target size versus recognition range for different turbulence conditions. The blue line indicates original imagery and the red line indicates turbulence mitigated imagery using method A.

Table 2. Overview of recognition range improvement due to turbulence mitigation for two different target sizes.

Turbulence condition	Target 3x3m			Target 5x5m		
	Rec. range Org. (km)	Rec. range Proc. (km)	Improvement (%)	Rec. range Org. (km)	Rec. range Proc. (km)	Improvement (%)
Weak	5.13	6.07	18%	8.30	10.38	25%
Moderate	5.13	5.92	15%	8.30	9.83	18%
Strong	3.37	3.71	10%	4.76	5.60	18%

From Table 2 it is clear that the improvement in recognition range is between 10 and 18% for a target of 3x3m. For a larger target (5x5m) the improvement in recognition range is between 18 and 25%. Also it seems that for stronger turbulence conditions the recognition range improvement is getting smaller.

6. CONCLUSIONS AND RECOMMENDATIONS

Turbulence compensation methods are important to mitigate the effects of atmospheric turbulence in image sequences. By significantly decreasing the blur (up to a factor of two) and decreasing the tilt (below 0.5 pixels) the image quality improves. This has a positive impact on detection and recognition ranges. A simulation experiment using method A shows that the recognition range improves with approximately 20% for weak to moderate turbulence conditions. For strong turbulence conditions the improvement of the recognition range is around 15%.

To mitigate the effects of atmospheric turbulence different approaches can be used. In general, the complexity of these approaches increases with the amount of turbulence. The effects of weak turbulence conditions can be compensated by processing on a global scale. Such processing can be done in real-time by hardware as well as software. The effects of moderate to strong turbulence conditions can be compensated by processing on a local scale. This can be done by using additional hardware (phase diversity²³) or by using a local-based software approach. Additional hardware is expensive but no temporal resolution is lost. A local-based software approach is a cheap and a flexible alternative at the expense of temporal resolution loss.

Although a lot of advances are made in the field of turbulence compensation in recent years, there remain a lot of challenges as well. The most important challenge is to develop more advanced local-based methods that are capable of 1) estimating and compensating blur on a local level¹², 2) estimating and compensating motion on a local level, 3) selecting the best temporal information on a local level and 4) performing previous capabilities on imagery that might contain a variety of scenes, ranging from static scenes to scenes with moving objects.

ACKNOWLEDGEMENTS

The authors would like to thank the NATO SET-165 group for the availability of the Dayton dataset.

REFERENCES

- [1] Jong, A. N. de., Schwering, P. B. W., Benoist, K. W., Gunter, W. H., Vrahimis, G., October, F. J., "Long-term measurements of atmospheric point-spread functions over littoral waters as determined by atmospheric turbulence," *Proc. SPIE Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXIII* **8355** (2012).
- [2] Schwering, P. B., Kunz, G. J., "Infrared scintillation effects over sea," *SPIE's 1995 Symposium on OE/Aerospace Sensing and Dual Use Photonics* **2471**, 204–215 (1995).
- [3] Roggeman, M. C., Welsh, B., "Imaging through turbulence," CRC Press, Boca Raton, USA (1996).
- [4] Eekeren, A. W. M. van., Schutte, K., Dijk, J., Schwering, P. B. W., Iersel, M. van., Doelman, N. J., "Turbulence Compensation: an overview," *Proc. SPIE Infrared Imaging Systems XXIII* **8355** (2012).
- [5] Eekeren, A. W. M. van., Schutte, K., Dijk, J., Schwering, P. B. W., "Quantitative evaluation of turbulence compensation," *Proc. SPIE Infrared Imaging Systems XXIV* **8706** (2013).
- [6] Huebner, C. S., "Compensating image degradation due to atmospheric turbulence in anisoplanatic conditions," *Proc. SPIE Mobile Multimedia/Image Processing, Security, and Applications* **7351** (2009).
- [7] Huebner, C. S., "Software-based turbulence mitigation of short exposure image data with motion detection and background segmentation," *Proc. SPIE* **8178** (2011).

- [8] Velluet, M. T., Vorontsov, M., Espinola, R., Marchi, G., Nicolas, S., Schwering, P., Riker, J., "Turbulence characterization and image processing data sets from a NATO RTO SET 165 trial in Dayton, OH, USA," Proc. SPIE Atmospheric Propagation IX **8380** (2012).
- [9] Bijl, P., Schutte, K., Hogervorst, M. A., "Applicability of TOD, MTDP, MRT and DMRT for dynamic image enhancement techniques," Infrared Imaging Systems: Design, Analysis, Modeling and Testing XVII **6207** (2006).
- [10] Krumpholz, M., Katehi, L. P. B., "MRTD: new time-domain schemes based on multiresolution analysis," Microwave Theory and Techniques, IEEE Transactions on **44**(4), 555–571 (1996).
- [11] Pham, T. Q., Bezuijen, M., Vliet, L. J. van., Schutte, K., Hendriks, C. L. L., "Performance of optimal registration estimators," Proc. SPIE Visual Information Processing XIV **5817**, 133–144 (2005).
- [12] Bouma, H., Dijk, J., Eekeren, A. W. M. van., "Precise local blur estimation based on the first-order derivative," Proc. SPIE Visual Information Processing XXI **8399** (2012).
- [13] Horn, B. K. P., Schunck, B. G., "Determining optical flow," Artificial Intelligence **17**, 185–203 (1981).
- [14] Hardie, R. C., Barnard, K. J., Bognar, J. G., Armstrong, E. E., Watson, E. A., "High-resolution image reconstruction from a sequence of rotated and translated frames and its application to an infrared imaging system," Optical Engineering **37**(1), 247–260 (1998).
- [15] Eekeren, A. W. M., Schutte, K., Oudegeest, O. R., Vliet, L. J. van., "Performance evaluation of Super-Resolution reconstruction methods on real-world data," EURASIP Journal on Advances in Signal Processing, 1–11 (2007).
- [16] Schutte, K., "Multiscale adaptive gain control of IR images," Proc. SPIE Infrared Technology and Applications XXIII **3061** (1997).
- [17] Scheifling, C., Marchi, G., "Iterative correction procedure for images degraded by turbulence," Proc. SPIE **7828** (2010).
- [18] Zuiderveld, K., "Contrast limited adaptive histogram equalization," Graphics gems IV, 474–485, Academic Press Professional, Inc. (1994).
- [19] Huebner, C. S., "Turbulence mitigation of short exposure image data using motion detection and background segmentation," Proc. SPIE **8355** (2012).
- [20] Gyaourova, A., Kamath, C., Cheung, S., "Block matching for object tracking," Lawrence livermore national laboratory (2003).
- [21] Ayers, G. R., Dainty, J. C., "Iterative blind deconvolution method and its applications," Optics letters **13**(7), 547–549 (1988).
- [22] Popkin, T., Cavallaro, A., Hands, D., "Accurate and efficient method for smoothly space-variant Gaussian blurring," Image Processing, IEEE Transactions on **19**(5), 1362–1370 (2010).
- [23] Eekeren, A. W. M. van., Schutte, K., Dijk, J., Schwering, P. B. W., "Time-Varying Phase Diversity Turbulence Compensation," Proc. SPIE Infrared Technology and Applications XXXVII **8012** (2011).