

Simultaneous Video Stabilization and Moving Object Detection in Turbulence

Omar Oreifej, Member, IEEE, Xin Li, Member, IEEE, and Mubarak Shah, Fellow, IEEE

Abstract—Turbulence mitigation refers to the stabilization of videos with non-uniform deformations due to the influence of optical turbulence. Typical approaches for turbulence mitigation follow averaging or de-warping techniques. Although these methods can reduce the turbulence, they distort the independently moving objects which can often be of great interest. In this paper, we address the novel problem of simultaneous turbulence mitigation and moving object detection. We propose a novel three-term low-rank matrix decomposition approach in which we decompose the turbulence sequence into three components: the background, the turbulence, and the object. We simplify this extremely difficult problem into a minimization of nuclear norm, Frobenius norm, and ℓ_1 norm. Our method is based on two observations: First, the turbulence causes dense and Gaussian noise, and therefore can be captured by Frobenius norm, while the moving objects are sparse and thus can be captured by ℓ_1 norm. Second, since the object's motion is linear and intrinsically different than the Gaussian-like turbulence, a Gaussian-based turbulence model can be employed to enforce an additional constraint on the search space of the minimization. We demonstrate the robustness of our approach on challenging sequences which are significantly distorted with atmospheric turbulence and include extremely tiny moving objects.

Index Terms—Three-Term Decomposition, Turbulence Mitigation, Rank Optimization, Moving Object Detection, Particle Advection, Restoring Force.

1 INTRODUCTION

THE refraction index of the air varies based on several atmospheric characteristics including the air's temperature, humidity, pressure, carbon dioxide level, and dust density. Such conditions are typically not homogeneous; for instance, a non-uniform temperature distribution might be observed above a surface receiving sunlight. Therefore, light rays travelling through the air with such non-uniform changes in its relative refraction index, will go through a complex series of refraction and reflection causing extreme spatially and temporally varying deformations to the captured images [1], [2], [3], [4], [5].

On the other hand, if the objects of interest are additionally moving in the scene, their motion will be mixed up with the turbulence deformation in the captured images, rendering the problem of detecting the moving objects extremely difficult. In this paper, we are interested in the dual problem of turbulence mitigation (stabilizing the sequence) and moving object detection under the turbulent medium. To the best of our knowledge, such a problem has never been explored before. Relevant previous approaches

have either focused on detecting moving objects or de-warping a deformed sequence, but not on both tasks concurrently. Note that other than image deformation, atmospheric turbulence may cause blur if the camera exposure time is not sufficiently short. In this paper, however, we focus only on image deformation because of the inherent confusion between the motion of the object and the motion caused by the turbulence.

Given a sequence of frames $\{I_1, \dots, I_T\}$ acquired from a stationary camera residing in a turbulent medium while observing relatively tiny moving objects, we decompose the sequence into background, turbulence, and object components. More precisely, consider the frames matrix $F = [\text{vec}\{I_1\} \dots \text{vec}\{I_T\}]$ for $I_k \in \mathbb{R}^{W \times H}$ ($k = 1, 2, \dots, T$), where $W \times H$ denotes the frame resolution (width by height), and $\text{vec} : \mathbb{R}^{W \times H} \rightarrow \mathbb{R}^M$ is the operator which stacks the image pixels as a column vector. We formulate our decomposition of F as:

$$\min_{A, O, E} \text{Rank}(A) \quad \text{s.t.} \quad F = A + O + E, \quad (1)$$

$$\|O\|_0 \leq s, \quad \|E\|_F \leq \sigma,$$

where F , A , O , and E are the matrices of frames, background, object, and error (turbulence), respectively. Here, the $\|\cdot\|_0$ -norm counts the number of nonzero entries, $\|\cdot\|_F$ -norm is the Frobenius norm which is equal to the square root of the sum of squared elements in the matrix, s represents an upper bound of the total number of moving objects' pixels across all images, and σ is a constant which reflects our knowledge of the maximum total variance due to

• O. Oreifej, and Mubarak Shah are with the Department of Electrical Engineering and Computer Science at University of Central Florida, 4000 Central Florida Blvd., Orlando, FL 32816. E-mail: oreifej@eecs.ucf.edu, shah@eecs.ucf.edu.

• Xin Li is with the Math Department at University of Central Florida, 4000 Central Florida Blvd., Orlando, FL 32816. E-mail: xli@math.ucf.edu.

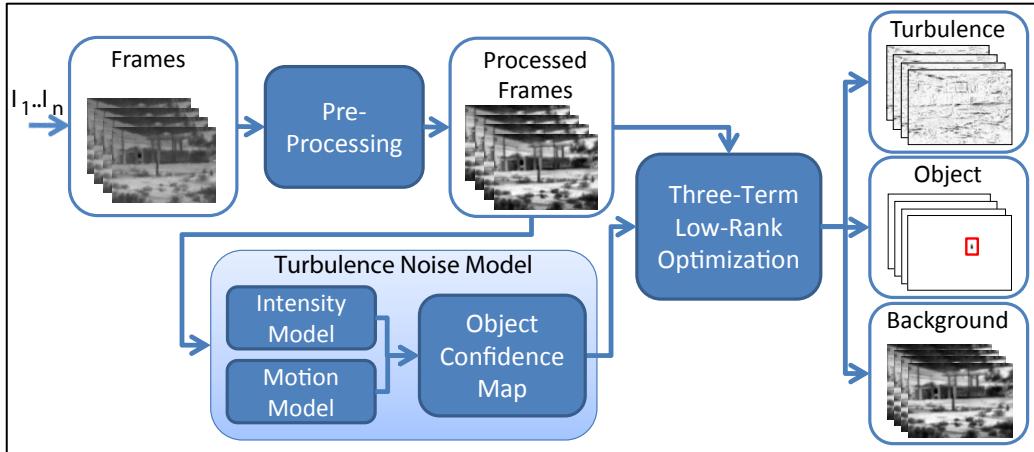


Fig. 1: The various steps of the proposed algorithm.

corrupted pixels across all images.

Our decomposition is based on the intrinsic properties of each of the components:

- 1) The Background: The scene in the background is presumably static; thus, the corresponding component in the frames of the sequence has linearly correlated elements. Therefore, the background component is expected to be the part of the matrix which is of low rank. Minimizing the rank of the low-rank component of the frames matrix F emphasizes the structure of the linear subspace containing the column space of the background, which reveals the background.
- 2) The Turbulence: Previous work dealing with turbulence, such as [6], [7], [8], [9], [10], demonstrated that the fluctuations of fluids (for instance air and water) attain Gaussian-like characteristics such as being unimodal, symmetric, and locally repetitive; therefore, the projected deformations in the captured sequence often approach a Gaussian distribution (we discuss this in more detail in Section 3.2). For this reason, the turbulence component can be captured by minimizing its Frobenius norm. The Frobenius norm of a matrix is the same as the Euclidean norm of the vector obtained from the matrix by stacking its columns. Therefore, as in the well-known vector case, constraining the error in the Euclidean norm is equivalent to controlling the sample variance of the error. Furthermore, theoretically, the estimate obtained by the Frobenius norm has several desirable statistical properties [11].
- 3) The Moving Objects: We assume that the moving objects are sparse in the sequence. This means that the number of pixels occupied by the moving objects is small (or can be considered as outliers) compared to the total number of pixels in the frames. This is a reasonable assumption

for most realistic surveillance videos. For this reason, the moving objects are best captured by restricting the number of nonzero entries (denoted by the ℓ_0 norm of the matrix), which is desirable for finding outliers.

In practice, parts of the turbulence could also appear as sparse errors in the object matrix O . Therefore, an additional constraint needs to be enforced on the moving objects. We employ a simple turbulence model to compute an object confidence map which is used to encourage the sparse solutions to be located on regions exhibiting linear motion that is dissimilar from the fluctuations of the turbulence. Under the new constraint, the optimization problem (1) must be reformulated as:

$$\begin{aligned} \min_{A,O,E} & \text{Rank}(A) \quad \text{s.t. } F = A + O + E, \\ & \|\Pi(O)\|_0 \leq s, \quad \|E\|_F \leq \sigma, \end{aligned} \quad (2)$$

where $\Pi : \mathbb{R}^{M \times T} \rightarrow \mathbb{R}^{M \times T}$ is the object confidence map, which is a linear operator that weights the entries of O according to their confidence of corresponding to a moving object such that the most probable elements are unchanged and the least are set to zero.

Figure 1 shows a diagram of the proposed approach. We first apply a pre-processing step to improve the contrast of the sequence, and reduce the spurious and random noise. Consequently, we obtain an object confidence map using a turbulence model which utilizes both the intensity and the motion cues. Finally, we decompose the sequence into its components using three-term rank minimization.

This paper makes three main contributions: First, we propose a new variant of matrix decomposition based on low-rank optimization and employ it to solve the novel problem of simultaneous moving object detection and turbulence mitigation in videos

distorted by atmospheric turbulence. Second, we propose a turbulence model based on both intensity and motion cues, where the motion distribution is derived from the Lagrangian particle advection framework [12]. The turbulence model is used to enforce an additional constraint on the decomposition which encourages the sparse solutions to be located in areas with non-Gaussian motion. Finally, we propose an additional force component in the particle advection framework in order to stabilize the particles in the turbulent medium and handle long sequences without discontinuities.

The rest of the paper is organized as follows: In the next section, we discuss the related works. In Section 3, we present our three-term rank minimization, followed by our probabilistic formulation of the object confidence map using particle advection. Section 4 discusses the details of implementation and the technical challenges of the three-term optimization. The experiments and the results are described in Section 5. Finally, Section 6 concludes the paper.

2 RELATED WORK

Rank optimization-based video de-noising has recently flourished with several successful works reported, from which we will only discuss the most related articles. Robust PCA was proposed in [13], where a low rank matrix was recovered from a small set of corrupted observations through convex programming. Similar concepts were later employed in [14] for video de-noising, where serious mixed noise was extracted by grouping similar patches in both spatial and temporal domains, and solving a low-rank matrix completion problem. Additionally, in [15], linear rank optimization was employed to align faces with rigid transformations, and concurrently detect noise and occlusions. In [16], Yu et. al. proposed an efficient solution to subspace clustering problems which involved the optimization of unitarily invariant norms. Another variant of such space-time optimization techniques is the total variation minimization, where for instance in [17], Chan et. al. posed the problem of video restoration as a minimization of anisotropic total variation given in terms of ℓ_1 -norm, or isotropic variation given in terms of ℓ_2 -norm. Consequently, the Lagrange multiplier method was used to solve the optimization function.

On the other hand, moving object detection is a widely investigated problem. When the scene is static, moving objects can be easily detected using frame differencing. A better approach would be to use the mean, the median, or the running average as the background [18]. The so-called eigenbackground [19] can also be obtained using PCA. However, when the scene is constantly changing because of noise, light changes, or camera shake, the intensities of image pixels can be considered as independent random variables, which can be represented using a statistical

model such as a Gaussian, a mixture of Gaussians, or a kernel density estimator. The model can then be used to compute the probability for each pixel to belong to either the background or the foreground. Examples of such approaches include [20], [21], [22]. Additionally, the correlation between spatially proximal pixels could also be employed to improve the background modelling using a joint domain (location) and range (intensity) representation of image pixels such as in [23].

Approaches for turbulence mitigation focused mainly on registration-based techniques. In [6], [5], [24], both the turbulence deformation parameters and a super-resolution image were recovered using area-based B-Spline registration. Moreover, in [25], Tian and Narasimhan proposed recovering the large non-rigid turbulence distortions through a “pull-back” operation that utilizes several images with known deformations. In model-based tracking [26], the characteristics of the turbulence caused by water waves were employed to estimate the water basis using PCA. More recently, in [7], turbulence caused by water was overcome by iteratively registering the sequence to its mean followed by RPCA to extract the sparse errors. Averaging-based techniques are also popular for video de-noising and turbulence mitigation, including pixel-wise mean/median, non-local means (NLM) [27], [28], fourier-based averaging [29], and speckle imaging [1], [3], [30]. Another category of methods for turbulence mitigation is the lucky region approach [4], [31], [32], where the least distorted patches of the video are selected based on several quality statistics, then those selected patches are fused together to compose the recovered video.

Clearly, previous work in moving object detection in dynamic scenes mostly focused on detecting the objects and did not consider recovering the background. Inversely, previous work in turbulence mitigation did not consider the possible interest in detecting moving objects in the scene. In this paper, we pose the two problems of moving object detection and turbulence mitigation as one application for our proposed three-term low-rank decomposition. We demonstrate how to decompose a turbulent video into separate background, foreground, and turbulence components. It is important to note that our method is not directly comparable to background subtraction or turbulence mitigation approaches; though, we do provide competitive results on each task separately.

3 PROPOSED APPROACH

We decompose the matrix which contains the frames of the turbulence video, into its components: the background, the turbulence, and the objects. The decomposition is performed by solving the rank optimization in equation (2), which enforces relevant constraints on each component. In the next subsection we describe the details of the decomposition approach.

3.1 Three-Term Decomposition

When solving equation (2), it is more convenient to consider the Lagrange form of the problem:

$$\begin{aligned} \min_{A, O, E} & \text{Rank}(A) + \tau \|\Pi(O)\|_0 + \lambda \|E\|_F^2 \\ \text{s.t. } & F = A + O + E, \end{aligned} \quad (3)$$

where τ and λ are weighting parameters. The optimization of (3) is not directly tractable since the matrix rank and the ℓ_0 -norm are nonconvex and extremely difficult to optimize. However, it was recently shown in [13] that when recovering low-rank matrices from sparse errors, if the rank of the matrix A to be recovered is not too high and the number of non-zero entries in O is not too large, then minimizing the nuclear norm of A (sum of singular values $\sum_i \sigma_i(A)$) and the ℓ_1 -norm of O can recover the exact matrices. Therefore, the nuclear norm and the ℓ_1 -norm are the natural convex surrogates for the rank function and the ℓ_0 -norm, respectively. Applying this relaxation, our new optimization becomes:

$$\min_{A, O, E} \|A\|_* + \tau \|\Pi(O)\|_1 + \lambda \|E\|_F^2 \quad \text{s.t. } F = A + O + E, \quad (4)$$

where $\|A\|_*$ denotes the nuclear norm of matrix A . We adopt the Augmented Lagrange Multiplier method (ALM) [33] to solve the optimization problem (4). Define the augmented Lagrange function for the problem as:

$$\begin{aligned} L(A, O, E, Y) = & \|A\|_* + \tau \|\Pi(O)\|_1 + \lambda \|E\|_F^2 + \\ & \langle Y, F - A - O - E \rangle + \frac{\beta}{2} \|F - A - O - E\|_F^2, \end{aligned} \quad (5)$$

where $Y \in \mathbb{R}^{M \times T}$ is a Lagrange multiplier matrix, β is a positive scalar, and $\langle \cdot, \cdot \rangle$ denotes the matrix inner product ($\text{trace}(A^T B)$). Minimizing the function in equation (5) can be used to solve the constrained optimization problem in equation (4). We use the ALM algorithm to iteratively estimate both the Lagrange multiplier and the optimal solution by iteratively minimizing the augmented Lagrangian function:

$$\begin{aligned} (A_{k+1}, O_{k+1}, E_{k+1}) = & \arg \min_{A, O, E} L(A, O, E, Y_k), \\ Y_{k+1} = & Y_k + \beta_k (F_{k+1} - A_{k+1} - O_{k+1} - E_{k+1}). \end{aligned} \quad (6)$$

When β_k is a monotonically increasing positive sequence, the iterations converge to the optimal solution of problem (4) [34]. However, solving equation (6) directly is difficult; therefore, the solution is approximated using an alternating strategy minimizing the augmented Lagrange function with respect to each component separately:

$$\begin{aligned} A_{k+1} &= \arg \min_A L(A, O_k, E_k, Y_k), \\ O_{k+1} &= \arg \min_O L(A_{k+1}, O, E_k, Y_k), \\ E_{k+1} &= \arg \min_E L(A_{k+1}, O_{k+1}, E, Y_k). \end{aligned} \quad (7)$$

Following the idea of the singular value thresholding algorithm [35], we derive the solutions for the update steps in equation (7) for each of the nuclear, Frobenius, and ℓ_1 norms. Please refer to the supplementary appendix for the complete derivations. Consequently, a closed form solution for each of the minimization problems is found:

$$\begin{aligned} UWV^T &= \text{svd}(F - O_k - E_k + \beta_k^{-1} Y_k), \\ A_{k+1} &= US_{1/\beta_k}(W)V^T, \\ O_{k+1} &= S_{\tau/\beta_k \Pi}(F - A_{k+1} - E_k + \beta_k^{-1} Y_k), \\ E_{k+1} &= (1 + \frac{2\lambda}{\beta_k})^{-1} (\beta_k^{-1} Y_k + F - A_{k+1} - O_{k+1}), \end{aligned} \quad (8)$$

where $\text{svd}(M)$ denotes a full singular value decomposition of matrix M , and $S_\alpha(\cdot)$ is the soft-thresholding operator defined for a scalar x as:

$$S_\alpha(x) = \text{sign}(x) \cdot \max\{|x| - \alpha, 0\}, \quad (9)$$

and for two matrices $A = (a_{ij})$ and $B = (b_{ij})$ of the same size, $S_A(B)$ applies the soft-thresholding entry-wise outputting a matrix with entries $S_{a_{ij}}(b_{ij})$.

The steps of our decomposition are summarized in Algorithm 1. In the next subsection, we describe our method to obtain the moving object confidence map Π , which is employed as a prior in the rank minimization problem.

3.2 Turbulence Model

We employ a turbulence model to enforce an additional constraint on the rank minimization such that moving objects are encouraged to be detected in locations with non-Gaussian deformations. Exact modelling of the turbulence is in fact ill-posed as it follows a non-uniform distribution which varies significantly in time, besides having an additional complexity introduced during the imaging process; thus, rendering the problem of modelling turbulence extremely difficult. Although the refraction index of the turbulent medium is often randomly changing, it is also statistically stationary [3], [30], [36]; thus, the deformations caused by turbulence are generally repetitive and locally centered [6], [7], [8], [9], [37]; this encourages the use of Gaussian-based models as approximate distributions that are general enough to avoid overfitting, but rather capture significant portion of the turbulent characteristics.

We use a Gaussian function to model the intensity distribution of a pixel going through turbulence. This is similar to [20] which employs a mixture of

Gaussians; however, we found that a single Gaussian worked better since more complicated models often require a period of training which is not available in our sequences. Therefore, the intensity of a pixel at location \mathbf{x} is modelled using a Gaussian distribution:

$$I(\mathbf{x}) \sim \mathcal{N}(\mu_I, \sigma_I), \quad (10)$$

where μ_I and σ_I are the mean and the standard deviation at \mathbf{x} , respectively. On the other hand, the deformation caused by turbulence can be captured in the motion domain besides the intensity. Therefore, we combine the intensity and the motion features to obtain a better model of turbulence. In order to capture the ensemble motion in the scene, we use the concept of a “particle” in a Lagrangian particle trajectory acquisition approach. We assume that a grid of particles is overlaid onto a scene where each particle corresponds to a single pixel (the granularity is controllable). The basic idea is to quantify the scene’s motion in terms of the motion of the particles which are driven by dense optical flow. A so-called particle advection [12], [38], [39] procedure is applied to produce the particle trajectories. Given a video clip $\in \mathbb{R}^{W \times H \times T}$, we denote the corresponding optical flow by (U_w^t, V_h^t) , where $w \in [1, W]$, $h \in [1, H]$, and $t \in [1, T - 1]$. The position vector (x_w^t, y_h^t) of the particle at grid point (w, h) at time t is estimated by solving the following differential equations:

$$\begin{aligned} \frac{dx_w^t}{dt} &= U_w^t, \\ \frac{dy_h^t}{dt} &= V_h^t. \end{aligned} \quad (11)$$

We use Euler’s method to solve them, similar to [38]. By performing advection for the particles at all grid points with respect to each frame of the clip, we obtain the clip’s particle trajectory set, denoted by $\{(x_w^t, y_h^t) | w \in [1, W], h \in [1, H], t \in [1, T]\}$.

We employ the spatial locations of the particle trajectories (i.e. (x_w^t, y_h^t)) to model the turbulence motion in the scene. The locations visited by a particle moving due to the fluctuations of the turbulence have a unimodal and symmetric distribution which approaches a Gaussian [6], [30], [10], [9]. This is dissimilar from the linear motion of the particles driven by moving objects. Therefore, we associate each particle with a Gaussian with mean μ_M and covariance matrix Σ_M :

$$\mathbf{x} \sim \mathcal{N}(\mu_M, \Sigma_M). \quad (12)$$

By augmenting the intensity model in equation (10) with the motion model in equation (12), the total confidence of corresponding to the turbulence versus the moving objects for a particle at location \mathbf{x} is expressed as a linear opinion pooling of the motion and the intensity cues

$$C(\mathbf{x}) = wP(I(\mathbf{x})|\mu_I, \sigma_I) + (1 - w)P(\mathbf{x}|\mu_M, \Sigma_M). \quad (13)$$

The parameters of our model $\{\mathbf{w}, \mu_I, \sigma_I, \mu_M, \Sigma_M\}$ can be learned by optimization using training sequences or set to constant values selected empirically. In the context of our three-term decomposition, the obtained confidence provides a rough prior knowledge of the moving objects’ locations, which can be incorporated into the matrix optimization problem in equation (4). Interestingly, this prior employs motion information; therefore, it is complementary to the intensity-based rank optimization, and can significantly improve the result.

At frame t , we evaluate all the particles’ locations against their corresponding turbulence models and obtain the turbulence confidence map $C_t \in \mathbb{R}^{W \times H}$. While C_t corresponds to the confidence of a particle to belong to turbulence, the desired Π in equation (4) corresponds to the confidence of belonging to the moving objects; therefore, we define the object confidence map Π as the complement of the stacked turbulence confidence maps:

$$\Pi = 1 - [vec\{C_1\} \cdots vec\{C_T\}]. \quad (14)$$

3.3 Restoring Force

The particles carrying the object’s motion typically drift far from their original locations leaving several gaps in the sequence. In the presence of turbulence, the drifting also occurs as a result of the turbulent motion. Therefore, the particles need to be reinitialized every certain number of frames which, however, creates discontinuities. This is a typical hurdle in the Lagrangian framework of fluid dynamics [12], [38], [39], which constitutes a major impediment for the application of particle flow to turbulence videos. In order to handle the drifting and the discontinuity problems associated with the particle flow, we use a new force component in the advection equation:

$$\begin{aligned} \frac{dx_w^t}{dt} &= U_w^t + G(x, x_o), \\ \frac{dy_h^t}{dt} &= V_h^t + G(y, y_o). \end{aligned} \quad (15)$$

We refer to the new force as “Restoring Force” - a reference to a local restoration force acting in the direction of the original location of each particle. We use a simple linear function to represent the restoring force:

$$G(x, x_o) = \frac{x - x_o}{s}, \quad (16)$$

where s is a scaling factor which trades off the detection sensitivity and the speed of recovery for the particles. In other words, if s is set to a high value, the effect of the restoring force will be negligible, and therefore the particles will require a relatively longer time to return to their original positions. In this case, the sensitivity of moving object detection will be

higher, but more prone to false positives. If s is low, the particles will be more attached to their original location, thus less affected by turbulence, but will have lower detection sensitivity. In our experiments, we set s to $0.5 \times W = 125$, which we found to be adequate for all sequences.

Using the restoring force allows continuous processing of the sequence without the need to reinitialize the particles. For instance, if an object moves to one side of the frame then comes back, we can still capture its motion when it returns. Additionally, the restoring force maintains the particles' motion within a certain range and provides robustness against random noise, thus reducing the number of false object detections. Figure (2) shows the overlaid particles on selected frames and the corresponding object particles with and without restoring force. It is clear that the restoring force stabilizes the particles and delivers better moving object confidence.

Algorithm 1: Simultaneous Turbulence Mitigation and Moving Object Detection

Input : Distorted image set stacked as
 $F \in \mathbb{R}^{M \times T} = [\text{vec}\{I_1\} \dots \text{vec}\{I_T\}]$
Initial Object Confidence Map
 $\Pi : \mathbb{R}^{M \times T} \rightarrow \mathbb{R}^{M \times T}$

Output: Solution to equation (1)
Background $A \in \mathbb{R}^{M \times T}$
Turbulence $E \in \mathbb{R}^{M \times T}$
Moving Objects $O \in \mathbb{R}^{M \times T}$

while not converged **do**

\\Minimize the Lagrange function in equation (5)

$UWV^T = \text{svd}(F - O_k - E_k + \beta_k^{-1}Y_k),$
 $A_{k+1} = US_{\frac{1}{\beta_k}}(W)V^T,$
 $O_{k+1} = S_{\frac{\tau}{\beta_k}}\Pi(F - A_{k+1} - E_k + \beta_k^{-1}Y_k),$
 $E_{k+1} = (1 + \frac{2\lambda}{\beta_k})^{-1}(\beta_k^{-1}Y_k + F - A_{k+1} - O_{k+1}),$
 $Y_{k+1} = Y_k + \beta_k(F - A_{k+1} - O_{k+1} - E_{k+1}),$
 $\beta_{k+1} = \rho\beta_k.$

end

4 ALGORITHM AND IMPLEMENTATION DETAILS

4.1 Pre-Processing

The noise caused by turbulence often has several random and spurious components which are difficult to model. Therefore, we employ temporal averaging to mitigate such components. We use a small averaging window of 8 frames to avoid distorting the objects' motion. In order to improve the contrast of the sequences, we apply adaptive histogram equalization, which is similar to the ordinary histogram equalization, however it operates on small patches of size 8×8 . Deviation around the optimal values

of such parameters only causes smooth and graceful degradation in the results.

4.2 Determining the Optimization Parameters

The parameters τ and λ from equation (4) correspond to the total number of moving objects' pixels across all images, and the total variance due to corrupted pixels across all images, respectively. In other words, a higher τ leads to an increased significance in minimizing the O component, thus obtaining sparser moving objects. A higher λ leads to an increased significance in minimizing the noise in E component, thus obtaining less noise in E , and a more turbulent background. On the other hand, decreasing τ and λ leads to placing more emphasis on minimizing the rank, thus obtaining a more static background in A , large turbulence in E , and less sparse moving objects in O .

Several theoretical considerations were previously studied to derive optimal values for similar parameters in [13]. However, such analysis does not apply to all practical scenarios. In the context of our three-term decomposition, the matrix to be decomposed is not an exact composition of the expected components. In addition, the components do not correspond exactly to their expected model. For instance, the background is expected to be of low rank; however, the exact desired rank is debatable, as a background of rank 1 is often not desirable since it will be a repetition of a static image, which is not realistic. Similarly, the desired sparsity of error varies significantly among different applications. Therefore, we argue that, in practical scenarios, such parameters are problem-dependant and highly heuristic. Therefore, we empirically set τ to 0.1 and λ to 2.0, which, as we will show, worked as a good compromise among the constraints of the optimization in all sequences.

4.3 Discussion of the Three-Term Model

Our method relies on a special three-term decomposition of matrices, which we formulated as the optimization problem (4). A similar optimization model has been proposed and studied very recently in [40], [41], and [33]. In [40], sufficient conditions are obtained in order to find an optimal solution to (4), recovering the low-rank and sparse components of F . In this paper, however, we are interested in the computational methods for finding the desired decompositions. This leads us to scheme (7) via the Alternating Directions Method of Multipliers (ADMM), which was first introduced in the mid-1970's by [42], [43], and is the current method of choice for large-scale non-smooth convex optimization, as in [41], [44], [45], [46].

A formulation similar to (7) can be obtained by replacing E with $F - A - O$ in the objective function in (4) in order to solve $\min_{A,O} \|A\|_* + \tau\|O\|_1 + \lambda\|F - A - O\|_F^2$.

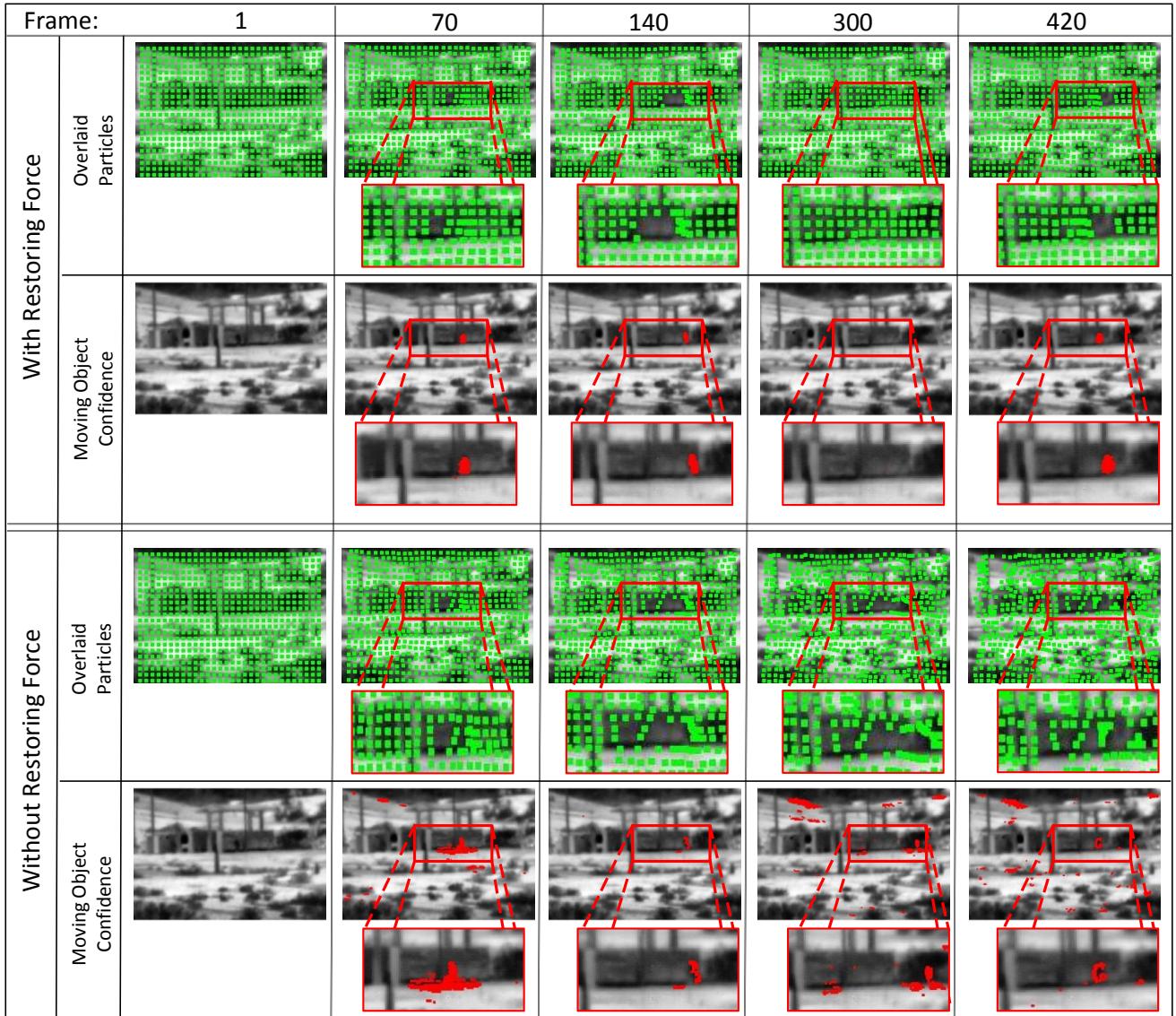


Fig. 2: Overlaid particles (the end points of the particle trajectories), and the corresponding moving object confidence (equation (14)) for sample frames. Rows one and three show the overlaid particles (the granularity is reduced for better visualization). Shown in red, in rows two and four, are the particles with high confidence of belonging to a moving object rather than turbulence (a fixed threshold is used). After frame 1, the object starts moving to the right, gets occluded at frame 300, then moves back to the left at frame 420. In the top two rows, the restoring force is employed; therefore, the particles gain two new properties: First, they become intact and attached to their original position, which make them robust against drifting due to the turbulence. Second, the particles automatically return to their original positions after the moving object disappears, which can be seen on frame 300. In the bottom two rows, the restoring force is not employed and such properties are not available; therefore, the particles continue to float and drift along the sequence, resulting in a poor object confidence performance.

This was indeed done by Lin et. al. [33], in which they proposed two methods to solve this optimization: the Augmented Lagrange Multiplier (ALM) and the Inexact Augmented Lagrange Multiplier (IALM). The IALM is an alternating direction approach which converges almost as fast as the exact ALM, but with significantly fewer partial SVDs. This was further explored in [41] where an additional three-term alternation method was discussed. As noted in [41], the convergence of the scheme with more than two terms alternation is still an open problem in general,

although numerical experiments strongly suggest that the alternation scheme is globally optimal under relatively mild conditions. However, for special three-term alternations with very strict and potentially non-practical assumptions, there do exist proofs as given in [45] and [44].

The convergence of scheme (7) to the optimal solution of (4) clearly requires further study; however, we are able to prove the feasibility of the accumulation points produced by the iterations of the algorithm.

Theorem 1: The sequences $\{A_k\}$, $\{O_k\}$, and $\{E_k\}$

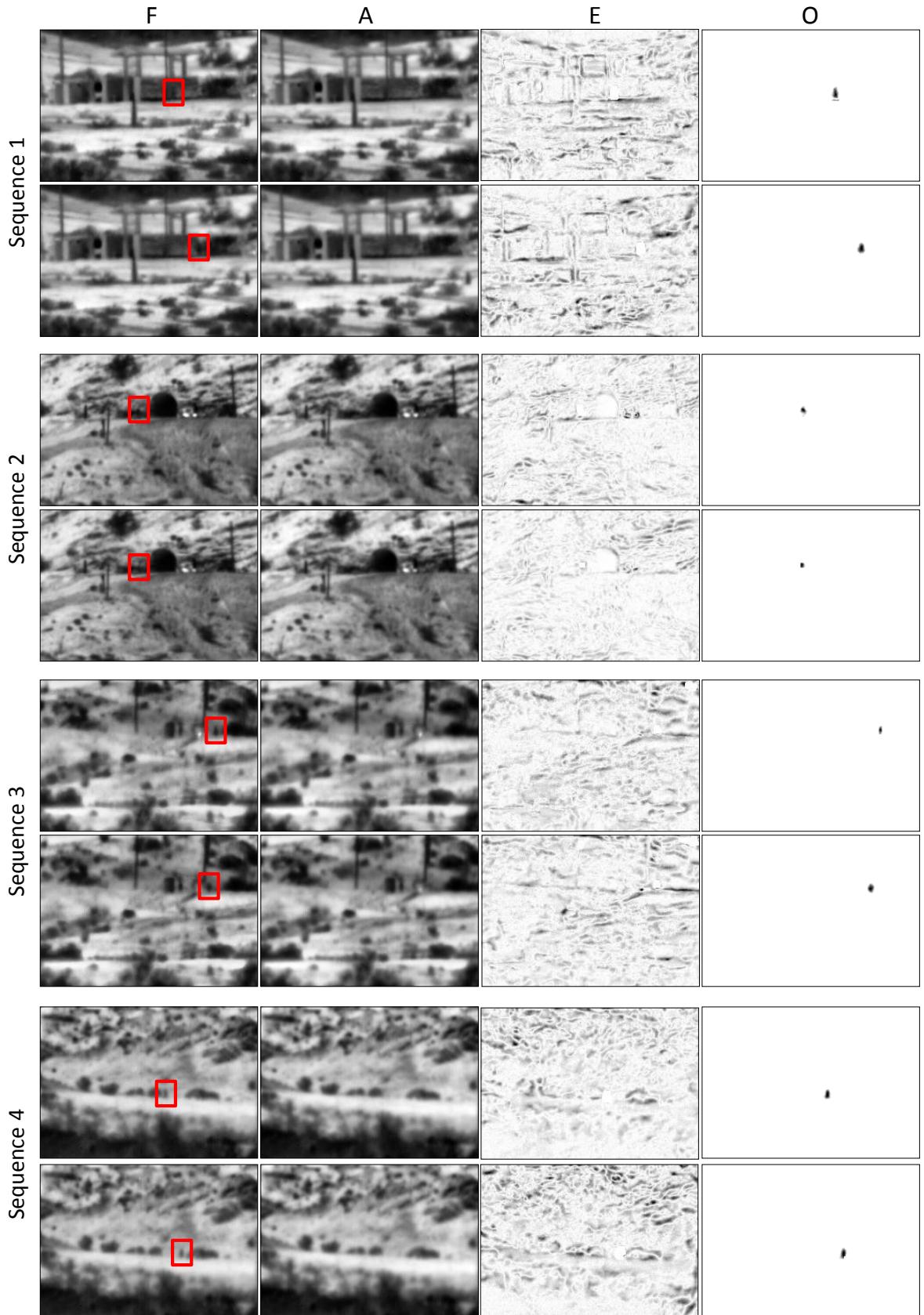


Fig. 3: Three-term decomposition results for two example frames from four testing sequences. Column F shows the original sequence (after pre-processing) which was decomposed into background (column A), turbulence (column E, absolute value of E is shown), and moving object (column O). Please refer to our website for the complete videos as the results of correcting the deformations are difficult to observe in a single frame.

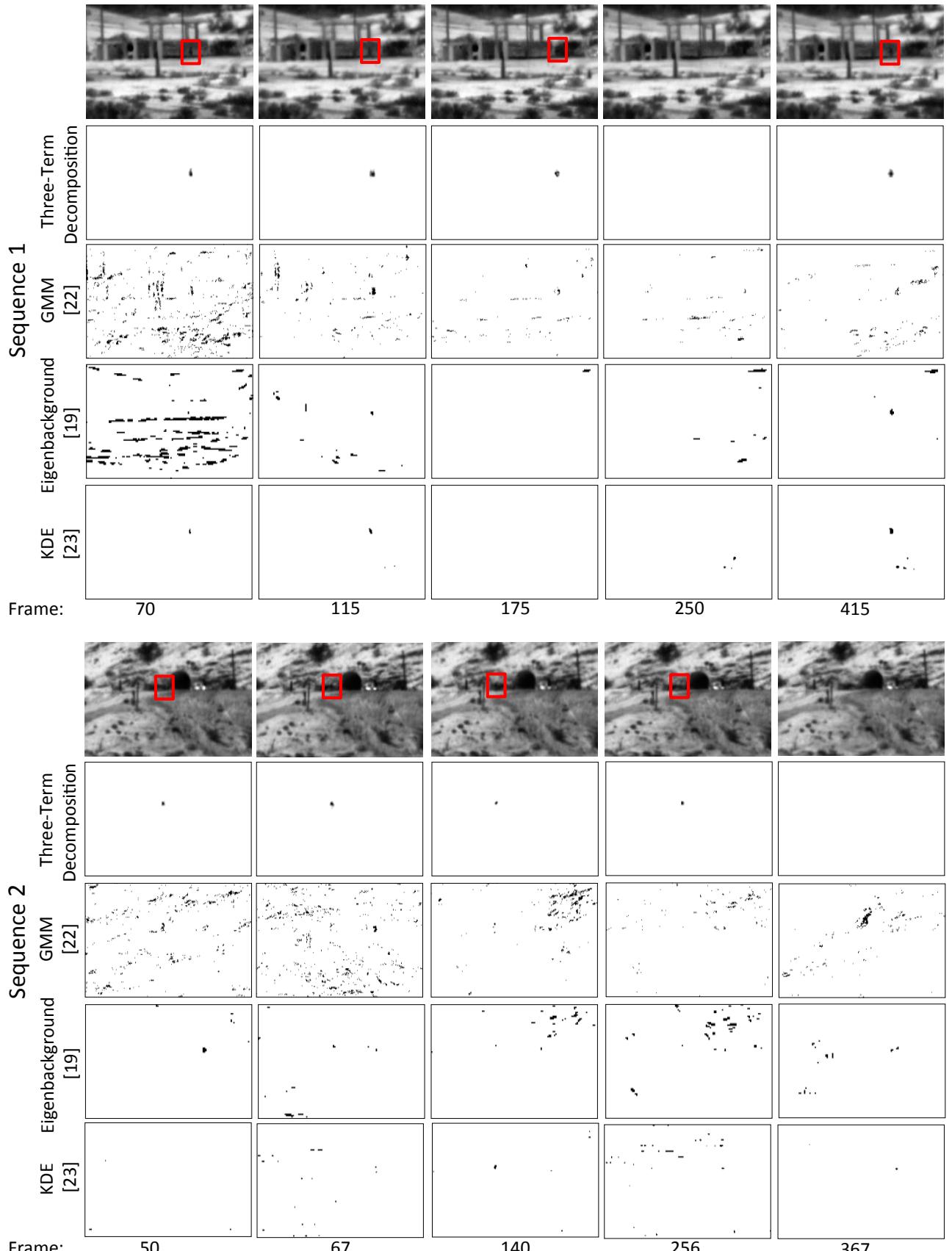


Fig. 4: Our moving object detection results on sample frames from sequences 1 and 2 compared to [22], [19], and [23]. For every sequence, the first row shows the frames, the second row shows the result from our method (taken from matrix O after the decomposition), the third, forth, and fifth rows show the background subtraction result obtained using [22], [19], and [23], respectively. Please zoom in to see the details.

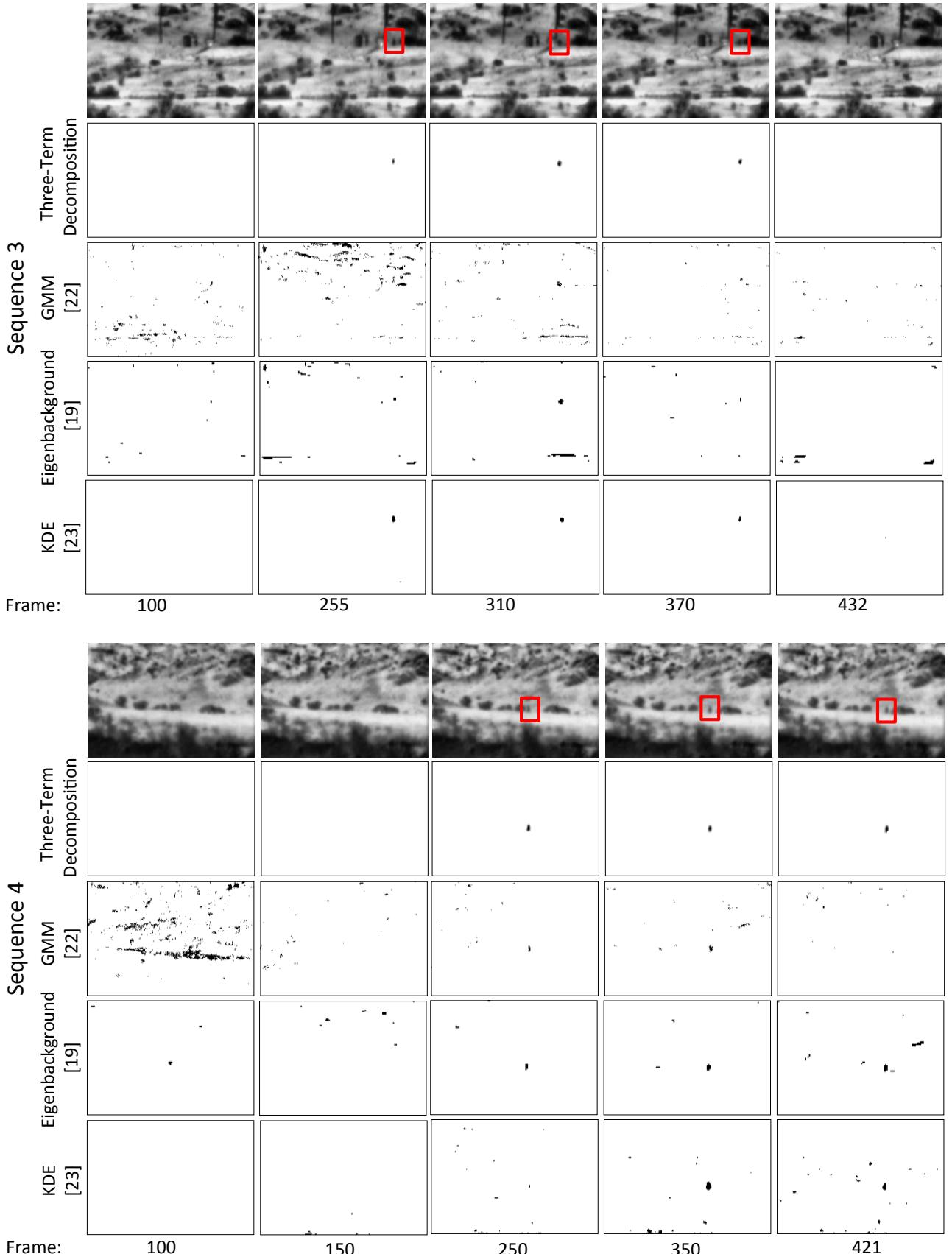


Fig. 5: Our moving object detection results on sample frames from sequences 3 and 4 compared to [22], [19], and [23]. For every sequence, the first row shows the frames, the second row shows the result from our method (taken from matrix O after the decomposition), the third, forth, and fifth rows show the background subtraction result obtained using [22], [19], and [23], respectively. Please zoom in to see the details.

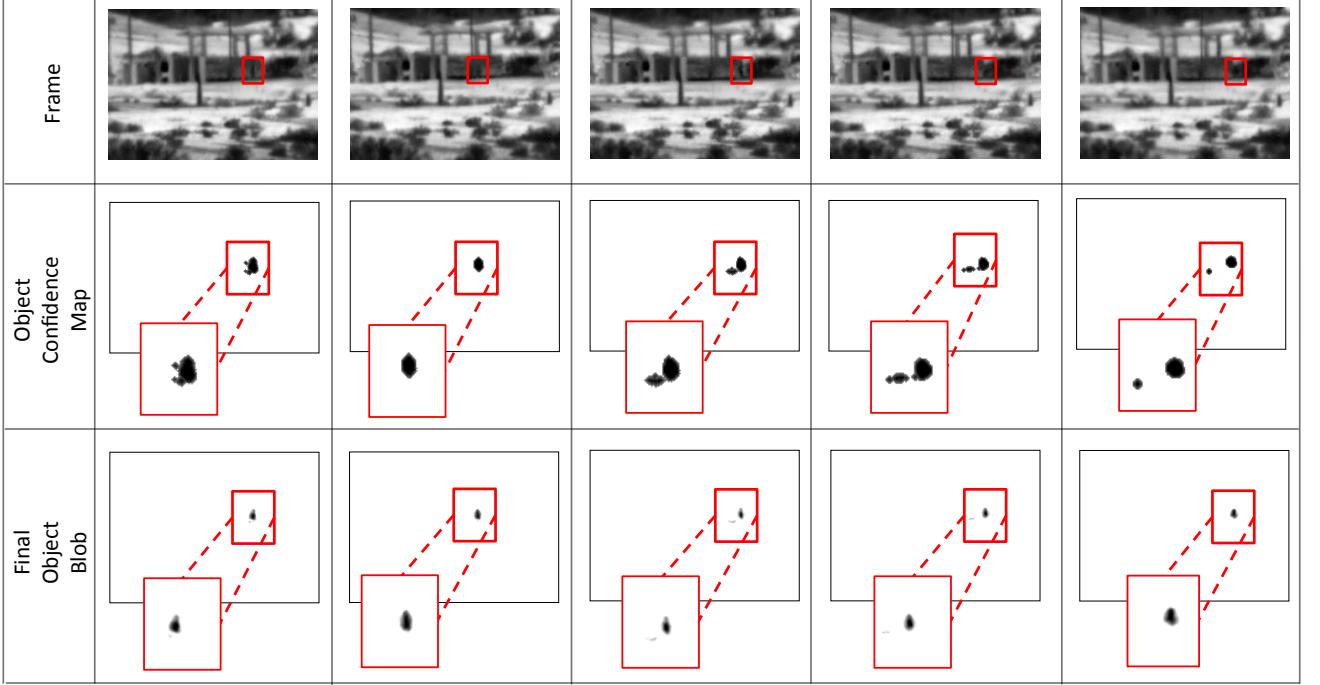


Fig. 6: Example frames illustrating the contribution of the turbulence model and the low-rank optimization in the total moving object detection performance, separately. The first row shows the frames, the second row shows the object confidence map obtained from the turbulence model (confidence values are mapped to $[0 - 255]$, the highest confidence is black), and the third row shows the final object blob after the three-term low-rank decomposition. Clearly, the turbulence model provides a rough estimation of the object location, while the low-rank optimization refines the result to obtain an accurate detection.

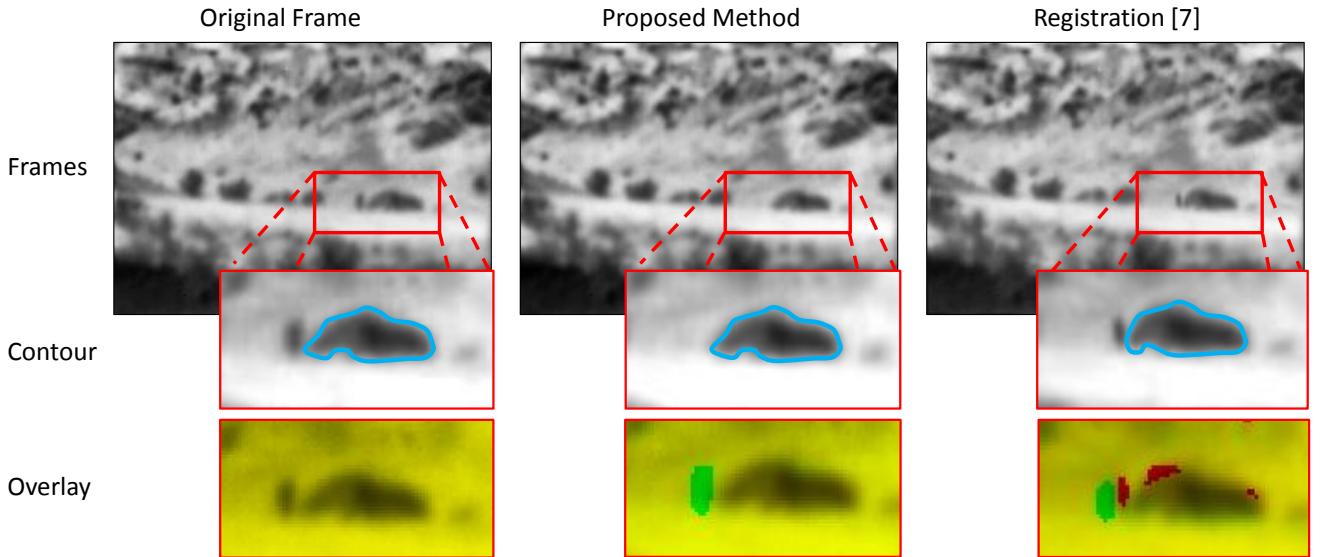


Fig. 7: Our turbulence mitigation results compared to non-rigid registration [7] for an example frame. The first row shows from left to right: The original frame, the recovered background using our proposed method, and the recovered background using [7]. The second row shows a zoomed-in version of the frames with overlaid contours of a vehicle near the moving object. Note how the object's motion near the vehicle caused a deformation in the contour of the vehicle. The third row demonstrates another visualization of the artifacts experienced by registration due to the moving object. We inserted each of the frames into the R color channel with the original frame in the G channel, and set the B channel to zero. It is obvious that, using our proposed method, the object is eliminated from the background (appears as a green blob) and the vehicle's shape is maintained, while using registration methods such as, [7] and [6], do not handle moving objects and accordingly result in deformations in the region surrounding the object (appears as several green and red blobs). The figure is best observed in colors. Please zoom in to see the details and refer to our website for the complete videos.

	Original	Registration [7]	3-Term Decomp.
Sequence 1	26.55	32.20	31.62
Sequence 2	27.49	34.05	34.11
Sequence 3	27.71	31.79	32.25
Sequence 4	27.82	32.72	33.13

TABLE 1: Comparison of the average PSNR in dB for the original sequences, after applying registration, and three-term decomposition.

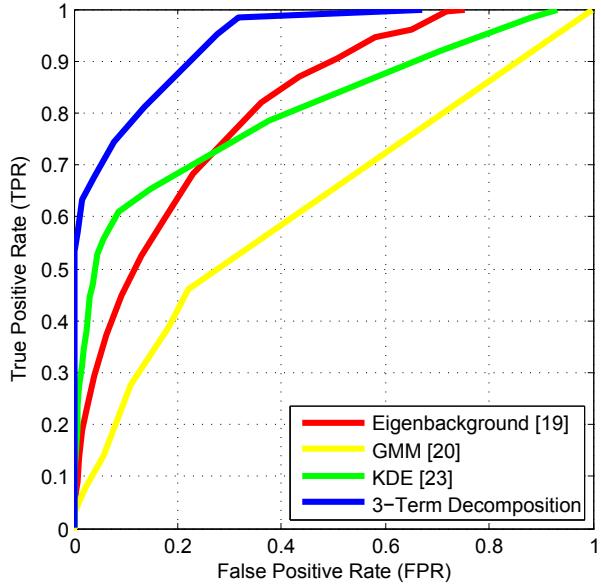


Fig. 8: Performance of our method compared to background subtraction methods.

generated by Algorithm 1 are bounded, and any accumulation point (A^*, O^*, E^*) of (A_k, O_k, E_k) is a feasible solution: $F = A^* + O^* + E^*$.

Refer to the supplementary appendix for an outline of the proof of Theorem 1. Our theorem indicates that the iterations of our algorithm are guaranteed to yield a decomposition of low-rank, sparse, and turbulence components, which is sufficient in our problem. Proving that the solution that we arrive at is globally optimal is, however, still an open problem which we leave for future work. However, our experimental evaluations agree with [41] and suggest the optimality of the alternation scheme.

5 EXPERIMENTS

We experimented extensively on the proposed ideas using four infrared sequences significantly distorted by atmospheric turbulence and also contain a moving human. The sequences and the code are available on our website: <http://www.cs.ucf.edu/~oreifej>. Each frame is 250×180 with 450 frames per sequence. The moving object occupies around 40 pixels in the frame and moves arbitrarily in the FOV. Typically, the object is static at the beginning of the sequence; therefore, we use the first 50 frames to compute the parameters of the intensity and the motion Gaussian

models at every pixel. Our three-term decomposition converges quickly in about 25 – 35 iterations, which takes approximately three minutes on a conventional laptop. Figure 3 shows our decomposition results. Our algorithm is able to decompose all the turbulence sequences generating a clear background and detecting the moving objects. The results are better observed in the videos available on our website.

Since this problem is novel, there are no directly comparable approaches. Therefore, we consider comparing each of the moving object detection and the turbulence mitigation tasks separately. We compare our moving object detection results with the background subtraction method described in [22], where the background is modelled using a mixture of Gaussians. Moreover, we compare our result with an eigenbackground model similar to [19], where the background basis for every 3×3 patch is found using PCA, then the patch is marked as foreground if it is not well represented by the PCA basis (i.e. its reconstruction error is above a threshold). We also compare our results with [23], which employs a nonparametric kernel density estimation method (KDE) over the joint domain (location) and range (intensity) representation of image pixels. Figures 4 and 5 illustrate the results of the comparison on sample frames from our sequences. It is clear that even state-of-the-art methods suffer in turbulence sequences; therefore, our method outperforms such methods significantly. In figure 6 we demonstrate the contribution of each of the turbulence model and the sparse optimization in the detection performance. As can be observed from the figure, our method leverages multiple constraints (sparsity, motion model, and intensity model) which complement each other to finally determine the object regions, thus significantly reducing miss-detection rates.

To evaluate the results quantitatively, we used a region-based measure where we applied connected components to the binary mask resulting from the background subtraction in order to obtain contiguous detection regions. Consequently, a detection region is considered correct if at least 50% of it is overlapping with the groundtruth, otherwise it is considered a miss-detection. The ROC curve in figure 8 summarizes the obtained results for all sequences.

In addition, we compare our turbulence mitigation results with the robust registration algorithm presented in [7]. The robust registration is an iterative process which recovers the original de-warped sequence by registering the frames to their mean, and then updating the frames and the mean at every iteration. The registration is performed using non-rigid alignment by a means of control points overlaid on the frames. To evaluate the performance, we measured the peak signal-to-noise ratio (PSNR) between the first frame of the sequence and the rest of the frames, and reported the average results for all the frames in table 1. It is clear that both our method and

registration can significantly stabilize the sequences and improve the PSNR. However, the moving object is not explicitly handled in the registration; therefore, it impedes the process by causing the control points to incorrectly shift in the direction of the object's motion, resulting in several artifacts in the surrounding area, as demonstrated in figure 7. In contrast, our three-term decomposition handles such difficulties by separating the moving object; therefore, we recover a clear background without artifacts and with significantly reduced turbulence.

6 CONCLUSION AND FUTURE WORK

We presented a novel method for concurrent turbulence mitigation and moving object detection. Our method leverages the low-rank, the Gaussian, and the sparse properties of the sequence, decomposing it into the background, the turbulence, and the moving objects, respectively.

In the future, we will investigate further applications of our three-term decomposition. In particular, we have recently shown in [39] how RPCA can be employed in action recognition to decompose the trajectories of an action in a moving camera scenario into low-rank components corresponding to camera motion and rigid object motion, and a sparse component corresponding to the articulated object motion. In an extension to that, we will investigate the use of the three-term decomposition to achieve finer decompositions of action sequences, and extract further components including the component corresponding to the Gaussian noise. We are also investigating the potential extension of the three-term-decomposition into a generic n -component decomposition, and its possible applications in clustering and segmentation problems.

7 APPENDIX

Here we first provide the derivation of equation set (8), which is the solution for the minimization problem in equation set (7). Consequently, we provide the proof of theorem 1. In all derivations we refer to the Lagrange function L defined in equation (5).

7.1 Derivation of the update step for A

$$A_{k+1} = \arg \min_A L(A, O_k, E_k, Y_k)$$

Dropping indices k and $k + 1$ for simplicity

$$\begin{aligned} A &= \arg \min_A L(A, O, E, Y) \\ &= \arg \min_A \|A\|_* + \langle Y, -A \rangle \\ &\quad + \frac{\beta}{2} \|F - A - O - E\|_F^2. \end{aligned}$$

The Frobenius norm is induced from the inner product, i.e. $\|X\|_F^2 = \langle X, X \rangle$. Therefore, replacing

the Frobenius norm with an inner product then expanding the inner product and separating A we obtain

$$\begin{aligned} A &= \arg \min_A \|A\|_* + \langle Y, -A \rangle \\ &\quad + \frac{\beta}{2} \{ \|A\|_F^2 - 2\langle F - O - E, A \rangle \} \\ &= \arg \min_A \|A\|_* + \\ &\quad \frac{\beta}{2} \{ \|A\|_F^2 - 2\langle \beta^{-1}Y + F - O - E, A \rangle \} \\ &= \arg \min_A \|A\|_* + \\ &\quad \frac{\beta}{2} \|A - \beta^{-1}Y - F + O + E\|_F^2 \\ &= \arg \min_A \beta^{-1}\|A\|_* + \\ &\quad \frac{1}{2} \|A - (\beta^{-1}Y + F - O - E)\|_F^2. \end{aligned}$$

Using the result from Singular Value Thresholding algorithm [35] we get

$$A = US_{\frac{1}{\beta}}(\Sigma)V^T,$$

where $U\Sigma V^T$ is the SVD of W , $W = \beta^{-1}Y + F - O - E$, and $S_\alpha(\cdot)$ is the soft thresholding operator defined in equation (9).

7.2 Derivation of the update step for O

$$O_{k+1} = \arg \min_O L(A_{k+1}, O, E_k, Y_k)$$

Dropping indices k and $k + 1$ for simplicity

$$\begin{aligned} O &= \arg \min_O L(A, O, E, Y) \\ &= \arg \min_O \tau \|\Pi(O)\|_1 + \langle Y, -O \rangle \\ &\quad + \frac{\beta}{2} \|F - A - O - E\|_F^2 \\ &= \arg \min_O \tau \|\Pi(O)\|_1 + \langle Y, -O \rangle \\ &\quad + \frac{\beta}{2} \{ \|O\|_F^2 - 2\langle F - A - E, O \rangle \} \\ &= \arg \min_O \tau \|\Pi(O)\|_1 + \\ &\quad \frac{\beta}{2} \{ \|O\|_F^2 - 2\langle \beta^{-1}Y + F - A - E, O \rangle \} \\ &= \arg \min_O \frac{\tau}{\beta} \|\Pi(O)\|_1 + \\ &\quad \frac{1}{2} \|O - (\beta^{-1}Y + F - A - E)\|_F^2. \end{aligned}$$

Let $X = \beta^{-1}Y + F - A - E$, then

$$O = \arg \min_O \frac{\tau}{\beta} \|\Pi(O)\|_1 + \frac{1}{2} \|O - X\|_F^2.$$

Using convex optimization theory [47], [48], 0 is in the subdifferential ∂ of the function:

$$\begin{aligned} 0 &\in \partial\left(\frac{\tau}{\beta}\|\Pi(O)\|_1 + \frac{1}{2}\|O - X\|_F^2\right) \\ &= \partial\left(\|\frac{\tau}{\beta}\Pi(O)\|_1\right) + O - X \\ &= \frac{\tau}{\beta}\Pi \operatorname{sign}(O) + O - X, \end{aligned}$$

which can be expressed as:

$$\begin{cases} 0 = \frac{\tau}{\beta}\Pi_{i,j} + O_{i,j} - X_{i,j} & \text{if } O_{i,j} > 0, \\ 0 = -\frac{\tau}{\beta}\Pi_{i,j} + O_{i,j} - X_{i,j} & \text{if } O_{i,j} < 0, \\ 0 \in [-1, 1]\frac{\tau}{\beta}\Pi_{i,j} + O_{i,j} - X_{i,j} & \text{if } O_{i,j} = 0, \end{cases}$$

where $O_{i,j}$ is the (i, j) -element of the matrix O . Rearranging the above equation we get

$$O_{i,j} = \begin{cases} X_{i,j} - \frac{\tau}{\beta}\Pi_{i,j} & \text{if } X_{i,j} > \frac{\tau}{\beta}\Pi_{i,j} \\ X_{i,j} + \frac{\tau}{\beta}\Pi_{i,j} & \text{if } X_{i,j} < -\frac{\tau}{\beta}\Pi_{i,j} \\ 0 & \text{if } |X_{i,j}| \leq \frac{\tau}{\beta}\Pi_{i,j} \end{cases}$$

This piecewise function is equivalent to the soft thresholding operator S defined in equation (9); hence, we can rewrite the above equation as

$$\begin{aligned} O &= S_{\frac{\tau}{\beta}\Pi}(X), \\ &= S_{\frac{\tau}{\beta}\Pi}(\beta^{-1}Y + F - A - E). \end{aligned}$$

7.3 Derivation of the update step for E

$$E_{k+1} = \arg \min_E L(A_{k+1}, O_{k+1}, E, Y_k)$$

Dropping indices k and $k + 1$ for simplicity

$$\begin{aligned} E &= \arg \min_E L(A, O, E, Y) \\ &= \arg \min_E \lambda\|E\|_F^2 + \langle Y, -E \rangle \\ &\quad + \frac{\beta}{2}\|F - A - O - E\|_F^2 \\ &= \arg \min_E \lambda\|E\|_F^2 + \\ &\quad \frac{\beta}{2}\{\|E\|_F^2 - 2\langle \beta^{-1}Y + F - A - O, E \rangle\} \\ &= \arg \min_E \left(\frac{2\lambda}{\beta} + 1\right)\{\|E\|_F^2 - 2 \right. \\ &\quad \left. \langle \left(\frac{2\lambda}{\beta} + 1\right)^{-1}(\beta^{-1}Y + F - A - O), E \rangle\} \right. \\ &= \arg \min_E \|E - \left(\frac{2\lambda}{\beta} + 1\right)^{-1}(\beta^{-1}Y + F - A - O)\|_F^2. \end{aligned}$$

Therefore, the minimum is obtained at

$$E = \left(\frac{2\lambda}{\beta} + 1\right)^{-1}(\beta^{-1}Y + F - A - O).$$

7.4 Proof of Theorem 1

The proof of Theorem 1 in Section 4.3 can be derived from the following lemmas.

Let A_k , O_k , E_k , Y_k , and β_k be as generated by Algorithm 1.

Lemma 1: Let

$$\begin{aligned} a_k &= Y_k + \beta_k(F - A_{k+1} - O_k - E_k), \\ b_k &= Y_k + \beta_k(F - A_{k+1} - O_{k+1} - E_k), \\ c_k &= Y_k + \beta_k(F - A_{k+1} - O_{k+1} - E_{k+1}), \end{aligned}$$

then the sequences $\{a_k\}$, $\{b_k\}$, and $\{c_k\}$ are bounded.

Note that according to Algorithm 1, we have $Y_{k+1} = c_k$, and recall that $L(\cdot)$ is the Lagrange function defined in equation (5).

Lemma 2: Let

$$\begin{aligned} L_{k+1} &= L(A_{k+1}, O_{k+1}, E_{k+1}, Y_k, \beta_k), \\ e_k &= \|F - A_k - O_k - E_k\|_F^2. \end{aligned}$$

Then $e_k \leq c\beta_{k-1}^{-2}$ for some constant $c > 0$, and

$$L_{k+1} - L_k \leq \frac{\beta_k + \beta_{k-1}}{2}e_k, \quad k = 1, 2, \dots$$

At every iteration in Algorithm 1, we set β_{k+1} to $\rho\beta_k$ with $\rho > 1$. Therefore, the above inequality can be rewritten as:

$$L_{k+1} - L_k \leq \frac{1+\rho}{c\beta_{k-1}}e_k, \quad k = 1, 2, \dots$$

Since $\{\beta_k\}$ is an increasing geometric sequence, we see that Lemma 2 implies the boundedness of the sequence $\{L_k\}$ and that $\lim_{k \rightarrow \infty}(F - A_k - O_k - E_k) = 0$, which implies that any accumulation points (if any) of (A_k, O_k, E_k) approaches a feasible solution to the desired decomposition. The following Lemma implies that such accumulation points exist.

Lemma 3: The sequences $\{A_k\}$, $\{O_k\}$, and $\{E_k\}$ are bounded.

Proofs of The Lemmas

Since we are dealing with finite dimensional Euclidean spaces, all norms are equivalent, and a bounded sequence in one norm is also bounded in any other norms. Therefore, we do not specify the type of the norm unless needed. Additionally, we are assuming the positive sequence $\{\beta_k\}$ satisfies $\sum 1/\beta_k < \infty$.

Proof of Lemma 1:

(i) Proof that $\{a_k\}$ is bounded:

We follow Lin et. al. [33] and note that

$$\begin{aligned} A_{k+1} &= \arg \min_A L(A, O_k, E_k, Y_k) \\ &\Rightarrow 0 \in \partial_A L(A_{k+1}, O_k, E_k, Y_k) \end{aligned}$$

$$\Rightarrow 0 \in \partial\|A_{k+1}\|_* - Y_k - \beta_k(F - A_{k+1} - O_k - E_k)$$

Therefore,

$$a_k = Y_k + \beta_k(F - A_{k+1} - O_k - E_k) \in \partial\|A_{k+1}\|_*.$$

From this, according to Theorem 4 of [33], the sequence $\{a_k\}$ is bounded.

(ii) Proof that $\{b_k\}$ is bounded:

$$O_{k+1} = \arg \min_O L(A_{k+1}, O, E_k, Y_k)$$

$$\Rightarrow 0 \in \partial_O L(A_{k+1}, O_{k+1}, E_k, Y_k)$$

$$\Rightarrow 0 \in \partial(\tau \|\Pi(O_{k+1})\|_1) - Y_k - \beta_k(F - A_{k+1} - O_{k+1} - E_k)$$

Therefore,

$$b_k = Y_k + \beta_k(F - A_{k+1} - O_{k+1} - E_k) \in \partial(\tau \|\Pi(O_{k+1})\|_1).$$

Thus, $b_k(i, j) = 0$ if $(i, j) \notin \text{supp}(\Pi)$, and $b_k(i, j) \in \partial(\tau |Q_{k+1}(i, j)|)$ if $(i, j) \in \text{supp}(\Pi)$. Using Theorem 4 of [33] (for the scalar case), the sequence $\{b_k\}$ is bounded.

(iii) Proof that $\{c_k\}$ is bounded:

First, note that

$$\partial(\lambda \|E_{k+1}\|_F^2) = \{2\lambda E_{k+1}\}.$$

Therefore, using

$$E_{k+1} = \arg \min_E L(A_{k+1}, O_{k+1}, E, Y_k),$$

we have

$$0 \in \partial_E L(A_{k+1}, O_{k+1}, E_{k+1}, Y_k)$$

$$\Rightarrow 0 \in \partial(\lambda \|E_{k+1}\|_F^2) - Y_k - \beta_k(F - A_{k+1} - O_{k+1} - E_{k+1})$$

Therefore,

$$c_k = Y_k + \beta_k(F - A_{k+1} - O_{k+1} - E_{k+1}) \in \{2\lambda E_{k+1}\}.$$

Thus,

$$c_k = 2\lambda E_{k+1}.$$

In Algorithm 1, $Y_{k+1} = c_k$. Hence, $Y_{k+1} = 2\lambda E_{k+1}$. Now, we have obtained

$$\begin{aligned} 2\lambda E_{k+1} &= Y_k + \beta_k(F - A_{k+1} - O_{k+1} - E_k) + \\ &\quad \beta_k(E_k - E_{k+1}) \\ &= b_k + \beta_k E_k - \beta_k E_{k+1}. \end{aligned}$$

Solving for E_{k+1} , we obtain

$$E_{k+1} = \frac{b_k}{\beta_k} \cdot \frac{1}{1 + \frac{2\lambda}{\beta_k}} + E_k \cdot \frac{1}{1 + \frac{2\lambda}{\beta_k}}.$$

Using the fact that $\{b_k\}$ is bounded and $\sum_{k=1}^{\infty} 1/\beta_k < \infty$, we conclude that $\{E_k\}$ must be bounded, and thus, the sequence $\{c_k\} = \{Y_{k+1}\} = \{2\lambda E_{k+1}\}$ is also bounded.

Remark. As a consequence of the proof, we obtained the boundedness of $\{E_k\}$ as well.

Proof of Lemma 2:

Write $L_{k+1} = L(A_{k+1}, O_{k+1}, E_{k+1}, Y_k, \beta_k)$. Then

$$\begin{aligned} L_{k+1} &\leq L(A_{k+1}, O_{k+1}, E_k, Y_k, \beta_k) \\ &\leq L(A_{k+1}, O_k, E_k, Y_k, \beta_k) \\ &\leq L(A_k, O_k, E_k, Y_k, \beta_k) \\ &= \|A_k\|_* + \tau \|\Pi(O_k)\|_1 + \lambda \|E_k\|_F^2 + \\ &\quad \langle Y_k, F - A_k - O_k - E_k \rangle + \\ &\quad \frac{\beta_k}{2} \|F - A_k - O_k - E_k\|_F^2 \\ &= L_k + \\ &\quad \langle Y_k - Y_{k-1}, F - A_k - O_k - E_k \rangle + \\ &\quad \frac{\beta_k - \beta_{k-1}}{2} \|F - A_k - O_k - E_k\|_F^2 \\ &= L_k + \\ &\quad \beta_{k-1} \|F - A_k - O_k - E_k\|_F^2 + \\ &\quad \frac{\beta_k - \beta_{k-1}}{2} \|F - A_k - O_k - E_k\|_F^2. \end{aligned}$$

Therefore,

$$L_{k+1} - L_k \leq \frac{\beta_k + \beta_{k-1}}{2} e_k.$$

Finally, note that

$$e_k = \|F - A_k - O_k - E_k\|_F^2 = \left(\frac{Y_k - Y_{k-1}}{\beta_{k-1}} \right)^2 = O(\beta_{k-1}^{-2}).$$

The last equality is due to the fact that $Y_k = c_{k-1}$ and $Y_{k-1} = c_{k-2}$ are bounded by Lemma 1. This completes the proof of Lemma 2.

Proof of Lemma 3:

Since $Y_{k+1} = c_k$, the sequence $\{Y_k\}$ is bounded by Lemma 1. Also the sequence L_k is bounded as implied by Lemma 2. Note that

$$\begin{aligned} \|Y_{k+1}\|_F^2 &= \|Y_k + \beta_k(F - A_{k+1} - O_{k+1} - E_{k+1})\|_F^2 \\ &= \|Y_k\|_F^2 + 2\beta_k \langle Y_k, F - A_{k+1} - O_{k+1} - E_{k+1} \rangle \\ &\quad + \beta_k^2 \|F - A_{k+1} - O_{k+1} - E_{k+1}\|_F^2. \end{aligned}$$

Therefore,

$$\begin{aligned} \frac{\|Y_{k+1}\|_F^2 - \|Y_k\|_F^2}{2\beta_k} &= \langle Y_k, F - A_{k+1} - O_{k+1} - E_{k+1} \rangle + \\ &\quad \frac{\beta_k}{2} \|F - A_{k+1} - O_{k+1} - E_{k+1}\|_F^2. \end{aligned}$$

Since $\{Y_k\}$ is bounded and $\beta_k \rightarrow \infty$, we see that

$$\langle Y_k, F - A_{k+1} - O_{k+1} - E_{k+1} \rangle + \frac{\beta_k}{2} \|F - A_{k+1} - O_{k+1} - E_{k+1}\|_F^2$$

converges to 0 (and thus bounded). Consequently, since

$$\begin{aligned} \|A_{k+1}\|_* + \tau \|\Pi(O_{k+1})\|_1 + \lambda \|E_{k+1}\|_F^2 &= L_{k+1} \\ &\quad - \langle Y_k, F - A_{k+1} - O_{k+1} - E_{k+1} \rangle \\ &\quad - \frac{\beta_k}{2} \|F - A_{k+1} - O_{k+1} - E_{k+1}\|_F^2, \end{aligned}$$

we see that $\{A_k\}$ and $\{E_k\}$ are bounded. This, together with the fact that $\lim_{k \rightarrow \infty} (F - A_k - O_k - E_k) = 0$, yields that the sequence $\{O_k\}$ is also bounded.

ACKNOWLEDGEMENTS

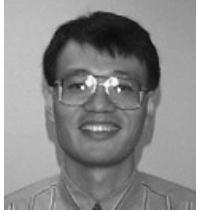
This material is based upon work supported by the Night Vision and Electronic Sensors Directorate (NVESD) under contract number W15P7T-08-D-P417. We thank Teresa Pace from NVESD for introducing us to the problem, providing the dataset, and conducting useful discussions. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NVESD.

REFERENCES

- [1] M. Roggeman and B. Welsh, "Imaging through turbulence," in *CRC Press*, 1996.
- [2] D. Li, R. M. Mersereau, and S. Simske, "Atmospheric turbulence-degraded image restoration using principal components analysis," in *IEEE Geoscience and Remote Sensing Letters*, 2007.
- [3] J. Goodman, "Statistical optics," in *Wiley-Interscience*, 2000.
- [4] G. C. M. Aubailly, M. Vorontsov and M. Valley, "Automated video enhancement from a stream of atmospherically-distorted images: the lucky-region fusion approach," in *SPIE*, 2009.
- [5] X. Zhu and P. Milanfar, "Image reconstruction from videos distorted by atmospheric turbulence," in *SPIE*, 2010.
- [6] M. Shimizu, S. Yoshimura, M. Tanaka, and M. Okutomi, "Super-resolution from image sequence under influence of hot-air optical turbulence," in *CVPR*, 2008.
- [7] O. Oreifej, G. Shu, T. Pace, and M. Shah, "A two-stage reconstruction approach for seeing through water," in *CVPR*, 2011.
- [8] Y. Yitzhaky, I. Dror, and N. Kopeika, "Restoration of atmospherically blurred images according to weather predicted atmospheric modulation transfer function (mtf)," in *Optical Engineering*, 1997.
- [9] M. Chang, C. Font, C. Gilbreath, E. Oh, E. Distefano, S. Restaino, C. Wilcox, and F. Santiago, "Comparing horizontal path c2n measurements over 0.6 km in the tropical littoral environment and in the desert," in *SPIE*, 2007.
- [10] A. Efros, V. Isler, J. Shi, and M. Visontai, "Seeing through water," in *NIPS*, 2004.
- [11] A. Bjorck, "Numerical methods for least squares problems," in *SIAM*, 1996.
- [12] S. Ali and M. Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis," in *CVPR*, 2007.
- [13] E. J. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" in *IEEE Sensor Array and Multichannel Signal Processing Workshop*, 2009.
- [14] H. Ji, C. Liu, and Y. X. Zuowei Shen, "Robust video denoising using low rank matrix completion," in *CVPR*, 2010.
- [15] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images," in *PAMI*, 2010.
- [16] Y. Yu and D. Schuurmans, "Rank/norm regularization with closed-form solutions: Application to subspace clustering," in *Conference on Uncertainty in Artificial Intelligence*, 2011.
- [17] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, "An augmented lagrangian method for total variation video restoration," in *IEEE Transactions on Image Processing*, 2010.
- [18] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objectss, ghosts and shadows in video streams," in *PAMI*, 2003.
- [19] N. M. Oliver, B. Rosario, and A. P. Pentland, "A bayesian computer vision system for modeling human interactions," in *PAMI*, 2000.
- [20] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *CVPR*, 1999.
- [21] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *ICCV*, 1999.
- [22] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *ICPR*, 2004.
- [23] Y. Sheikh and M. Shah, "Bayesian modeling of dynamic scenes for object detection," in *PAMI*, 2005.
- [24] X. Zhu and P. Milanfar, "Stabilizing and deblurring atmospheric turbulence," in *ICCP*, 2011.
- [25] Y. Tian and S. Narasimhan, "A globally optimal data-driven approach for image distortion estimation," in *CVPR*, 2010.
- [26] Y. Tian and S. Narasimhan, "Seeing through water: Image restoration using model-based tracking," in *ICCV*, 2009.
- [27] A. Buades, B. Coll, and J. Morel, "A non-local algorithm for image denoising," in *CVPR*, 2005.
- [28] C. Liu and W. Freeman, "A high-quality video denoising algorithm based on reliable motion estimation," in *ECCV*, 2010.
- [29] Z. Wen, D. Fraser, A. Lambert, and H. Li, "Reconstruction of underwater image by bispectrum," in *ICIP*, 2007.
- [30] J. Goodman, "Introduction to fourier optics," in *Roberts and Company Publishers*, 2004.
- [31] A. Donat, G. Dahme, and E. Ribeiro, "Classification of textures distorted by water waves," in *ICPR*, 2006.
- [32] N. Joshi and M. Cohen, "Lucky imaging for multi-image denoising, sharpening, and haze removal," in *ICCP*, 2010.
- [33] Z. Lin, M. Chen, L. Wu, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," in *UIUC Technical Report*, 2009.
- [34] D. P. Bertsekas, "Nonlinear programming," in *Athena Scientific*, 2004.
- [35] J.-F. Cai, E. J. Candes, and Z. Shen, "A singular value thresholding algorithm for matrix completion," in *SIAM J. on Optimization, Volume 20, Issue 4, pp. 1956-1982*, 2010.
- [36] H. Branover and Y. Unge, "Progress in turbulence research," in *AIAA*, 1994.
- [37] M. Isichenko and A. Gruzinov, "Isotopological relaxation, coherent structures, and gaussian turbulence in two-dimensional (2-d) magnetohydrodynamics (mhd)," in *Physics of Plasmas*, 1994.
- [38] S. Wu, B. Moore, and M. Shah, "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes," in *CVPR*, 2010.
- [39] S. Wu, O. Oreifej, and M. Shah, "Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories," in *ICCV*, 2011.
- [40] A. Agarwal, S. Negahban, and M. Wainwright, "Noisy matrix decomposition: optimal rates in high dimensions," in *preprint*, 2011.
- [41] M. Tao and X. Yuan, "Recovering low-rank and sparse components of matrices from incomplete and noisy observations," in *SIAM Journal on Optimization*, 2011.
- [42] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximation," in *Computers and Mathematics with Applications*, 1976.
- [43] R. Glowinski and A. Marroco, "Sur l'approximation, par éléments finis d'ordre un, et la résolution, par penalisation-dualité, d'une classe de problèmes de dirichlet non linéaires," in *Revue Française d'Automatique, Informatique et Recherche Opérationnelle*, 1975.
- [44] D. Goldfarb and S. Ma, "Fast multiple splitting algorithms for convex optimization," in *preprint*, 2009.
- [45] B. He and X. Yuan, "Linearized alternating direction method with gaussian back substitutions for separable convex programming," in *preprint*, 2011.
- [46] Y. Zhang, "An alternating direction algorithm for nonnegative matrix factorization," in *preprint*, 2010.
- [47] J. Nocedal and S. Wright, "Numerical optimization (2nd ed.)," in *Springer-Verlag, New York*, 2006.
- [48] G. Magaril-Ilyaev and V. Tikhomirov, "Convex analysis: Theory and applications," in *AMS, Providence, RI*, 2003.



Omar Oreifej received his B.S. degree in computer engineering from the University of Jordan, Amman, Jordan, in 2006, and his M.S. degree in computer engineering from the University of Central Florida, Orlando, Florida, in 2009. He is currently working towards his PhD degree in the Computer Vision Laboratory, University of Central Florida. His research interests include video denoising, low rank optimization, Sparse Representation, object tracking, and object detection.



Xin Li Dr. Xin Li is Professor of Mathematics at the University of Central Florida. He received his B.S. and M.S. degrees from Zhejiang University in 1983 and 1986 respectively and his Ph.D. in Mathematics from the University of South Florida in 1989. His research interests include Approximation Theory and its applications in mathematical image and video analysis.



Mubarak Shah Dr. Mubarak Shah, Agere Chair Professor of Computer Science, is the founding director of the Computer Visions Lab at University of Central Florida (UCF). He is a co-author of three books (Motion-Based Recognition (1997), Video Registration (2003), and Automated Multi-Camera Surveillance: Algorithms and Practice (2008)), all by Springer. He has published extensively on topics related to visual surveillance, tracking, human activity and action recognition, object detection and categorization, shape from shading, geo registration, visual crowd analysis, etc. Dr. Shah is a fellow of IEEE, IAPR, AAAS and SPIE. In 2006, he was awarded the Pegasus Professor award, the highest award at UCF, given to a faculty member who has made a significant impact on the university. He is ACM Distinguished Speaker. He was an IEEE Distinguished Visitor speaker for 1997-2000, and received IEEE Outstanding Engineering Educator Award in 1997. He received the Harris Corporation's Engineering Achievement Award in 1999, the TOKTEN awards from UNDP in 1995, 1997, and 2000; SANA award in 2007, an honorable mention for the ICCV 2005 Where Am I? Challenge Problem, and was nominated for the best paper award in ACM Multimedia Conference in 2005 and 2010. At UCF he received Scholarship of Teaching and Learning (SoTL) award in 2011; College of Engineering and Computer Science Advisory Board award for faculty excellence in 2011; Teaching Incentive Program awards in 1995 and 2003, Research Incentive Award in 2003 and 2009, Millionaires' Club awards in 2005, 2006, 2009, 2010 and 2011; University Distinguished Researcher award in 2007 and 2012. He is an editor of international book series on Video Computing; editor in chief of Machine Vision and Applications journal, and an associate editor of ACM Computing Surveys journal. He was an associate editor of the IEEE Transactions on PAMI, and a guest editor of the special issue of International Journal of Computer Vision on Video Computing. He was the program co-chair of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.