

Illumination robust change detection with CMOS imaging sensors

Vijay Rengarajan¹, Sheetal B. Gupta¹, A.N. Rajagopalan¹, and Guna Seetharaman²

¹ Department of Electrical Engineering, Indian Institute of Technology Madras

² Information Directorate, AFRL/RIEA

This is a preprint version of the paper presented at SPIE Defence + Security, International Society for Optics and Photonics, 2015.

Citation Vijay Rengarajan ; Sheetal B. Gupta ; A. N. Rajagopalan and Guna Seetharaman “Illumination robust change detection with CMOS imaging sensors,” Proc. SPIE 9473, Geospatial Informatics, Fusion, and Motion Video Analytics V, 947303 (May 21, 2015);

Copyright 2015 Society of Photo-Optical Instrumentation Engineers. One print or electronic copy may be made for personal use only. Systematic reproduction and distribution, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper are prohibited.

DOI link <http://dx.doi.org/10.1117/12.2176816>

Illumination Robust Change Detection with CMOS Imaging Sensors

Vijay Rengarajan^a, Sheetal B. Gupta^a, A.N. Rajagopalan^a, and Guna Seetharaman^b

^aIndian Institute of Technology Madras, Chennai, India;

^bInformation Directorate, Air Force Research Laboratory, Rome, NY

ABSTRACT

Change detection between two images in the presence of degradations is an important problem in the computer vision community, more so for the aerial scenario which is particularly challenging. Cameras mounted on moving platforms such as aircrafts or drones are subject to general six-dimensional motion as the motion is not restricted to a single plane. With CMOS cameras increasingly in vogue due to their low power consumption, the inevitability of rolling-shutter (RS) effect adds to the challenge. This is caused by sequential exposure of rows in CMOS cameras unlike conventional global shutter cameras where all pixels are exposed simultaneously. The RS effect is particularly pronounced in aerial imaging since each row of the imaging sensor is likely to experience a different motion. For fast-moving platforms, the problem is further compounded since the rows are also affected by motion blur. Moreover, since the two images are shot at different times, illumination differences are common. In this paper, we propose a unified computational framework that elegantly exploits the sparsity constraint to deal with the problem of change detection in images degraded by RS effect, motion blur as well as non-global illumination differences. We formulate an optimization problem where each row of the distorted image is approximated as a weighted sum of the corresponding rows in warped versions of the reference image due to camera motion within the exposure period to account for geometric as well as photometric differences. The method has been validated on both synthetic and real data.

Keywords: Change detection, CMOS sensors, rolling shutter, motion blur, illumination variation, aerial imaging

1. INTRODUCTION

Detecting changes between images is an important problem in many research areas such as aerial surveillance, object tracking, cartography, etc. Unlike stationary camera surveillance, aerial image capturing systems provide complex challenges due to the moving nature of the vehicle. As the scene is being exposed, motion of the vehicle, and hence the camera, causes motion blur in the resultant images. These conventional cameras employ CCD sensors in which the whole image is captured during a single global exposure time period, and hence are called global shutter (GS) cameras. Registering images affected by motion blur using either feature-based or photometric-based approaches is not possible without considering the effect of and solving for the camera motion.

Recently, CMOS sensors are being increasingly employed in cameras since the hardware circuitry used is minimal compared to that of used for CCD sensors. The photon acquisition mechanism is modified in CMOS cameras to enable this simplification. Instead of a global exposure of all sensors in the image plane array, each row of the sensor plane is exposed sequentially that enables sharing a single read-out circuit for all rows, thereby reducing power consumption and cost. These cameras that employ row-wise sensor acquisition are called rolling shutter (RS) cameras. This change in the way of image acquisition invites new artifacts in the form of row-dependent motion blur when the camera moves during exposure time. In addition to motion blur and rolling shutter artifacts, illumination variations prove to be an important factor to consider in change detection algorithms. Even in the case of the same scene being captured by the same camera, the time of capture could alter the photometric nature of the recorded image due to atmospheric illumination changes between the two times. A good change detection algorithm for CMOS cameras should thus take into account these three phenomena, *viz.* motion blur, rolling shutter effect, and illumination.

Further author information: (Send correspondence to Vijay Rengarajan). Vijay Rengarajan (E-mail: ee11d035@ee.iitm.ac.in), Sheetal B. Gupta (E-mail: ee13s063@ee.iitm.ac.in), A.N. Rajagopalan (E-mail: raju@ee.iitm.ac.in, Telephone: +91 44 2257 5430), Guna Seetharaman (E-mail: gunasekaran.seetharaman@us.af.mil, Telephone: 1 (315)-330-2414)

1.1 Related Works

We review some of the recent works that deal with motion blur, rolling shutter effect, and illumination in this section.

Motion Blur Modeling a convolutional blur model is common case in most traditional uniform deblurring algorithms. This model does not encompass all kinds of blur but a restrictive form involving only in-plane camera translations. When the camera is free to move during capture, which is especially the case in airborne systems, the motion is not restricted to simple translations. In real a general 6D motion involving rotations and translations is possible resulting in non-uniform blurring across the image. Recent works such as Whyte *et al.*,¹ Hu and Yang,² Paramanand and Rajagopalan³ model the motion-blurred image as weighted instances of the warps of the latent image for planar scenes that the camera had experienced during capture. Tai *et al.*⁴ assume that the blurring function is known apriori and proposed a non-blind space-variant deblurring scheme based on Richardson-Lucy deconvolution. Whyte *et al.*¹ represent the blurring function in a dense 3D grid of camera rotations to perform non-uniform deblurring. Xu *et al.*⁵ propose a single model to perform both uniform and non-uniform deblurring using ℓ_0 sparse representation.

Rolling Shutter Effect Works that deal with RS effect focus on the problem of video stabilization. Liang *et al.*⁶ estimate a global motion between frames of a video that is used to assign motion for every row using Bezier curve interpolation and rectify the RS effect. Baker *et al.*⁷ model the RS removal problem as temporal super-resolution of camera motion. Ringaby and Forssen⁸ model 3D camera rotations across keypoints in frames of a video, fit them to a continuous curve, and rectify the RS effect and stabilize the video from the interpolated motion of each row. Grundmann *et al.*⁹ propose a technique based on homography mixtures for video stabilization. Pichaikuppan *et al.*¹⁰ handle RS effect and motion blur in the application of change detection.

Illumination Modeling Illumination change is an important factor to consider to compare images of a scene taken at different times. An oft-followed approach is to transform images to a common canonical illumination domain, in which the two images would thus have same illumination and could be compared to detect changes. White balancing used by Gijsenij *et al.*¹¹ is one such transformation. Another technique is to normalize the mean and variance of the pixel intensity values of images. Both images could be normalized to a fixed mean and variance, or one of the images could be normalized to the mean and variance of the other image. This technique was used by Dai and Khorram.¹² Albedo is invariant to illumination, and hence its extraction and usage is common in literature to handle illumination variation in image registration. Phong¹³ uses homographic filtering to extract albedo images.

1.2 Contributions

In this paper, we build upon our framework¹⁰ of change detection in the presence of motion blur and rolling shutter effect by additionally handling illumination variations. To our knowledge, this is the first work to deal with all these three phenomena in a single framework. We assume that the captured scene is planar and is fronto-parallel to the image plane. We leverage the sparsity of camera motion during the exposure time within a camera pose space and the sparsity of the spatial coverage of areas of changes. We propose a recursive procedure that adapts itself to local variations of illumination.

2. IMAGE FORMATION MODEL IN CMOS SENSOR CAMERAS

In this section, we first review the image acquisition mechanism in cameras equipped with CMOS sensors, and then model the rolling shutter effect, motion blur, and illumination.

2.1 Image Acquisition in CMOS Cameras

A CMOS camera, similar to a conventional CCD camera, has an array of photosensors with dimensions equal to the resolution of the camera. Each sensor collects photons which help in assigning an intensity to every pixel. To reduce the hardware requirement involved, a common readout circuit is employed across rows. This leads to serial acquisition of photons row-wise as demonstrated in Fig. 1. Let M and N denote the number of rows and columns of the sensor array respectively, and t_e represent the exposure time of a single row. Since the read-out circuit is common to all rows, the read-out times of any two rows cannot coincide. Hence the starting time of the exposure period of a row is delayed by an amount $t_d < t_e$ with respect to that of its previous row. When the camera moves during image acquisition, this row-wise exposure entertains a new image formation model. The values t_d and t_e are the same for all rows during image capture. In Fig. 1, the total exposure time is given by $T_e = (M - 1)t_d + t_e$. If the first row starts its exposure at $t = 0$, then the i th row is exposed during the time interval, $(i - 1)t_d \leq t \leq (i - 1)t_d + t_e$.

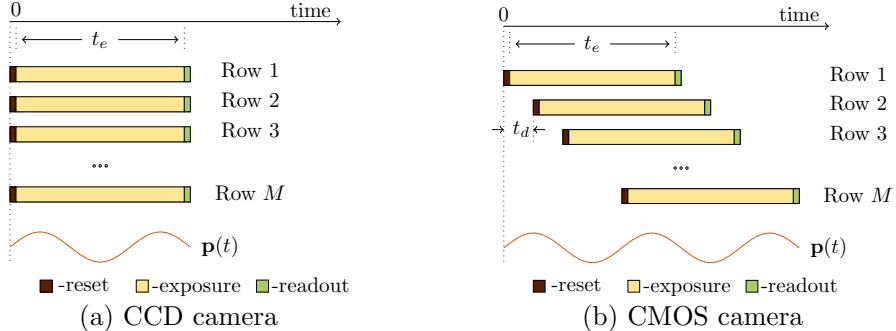


Figure 1. Differences in exposure mechanism of CCD and CMOS cameras.

2.2 Motion Blur Model

Let \mathbf{f} be the image captured by a CMOS camera when it is stationary. We assume that the scene is static. The image thus recorded is clean and contains no artifacts such as blur. Let \mathbf{g} be the image captured by the same CMOS camera when the camera moves during the exposure time. The camera motion is denoted by $\mathbf{p}(t)$, which is a six-dimensional vector corresponding to 3D translations and 3D rotations at a particular time t . In a conventional CCD camera, this results in motion blur in which all pixels of the observed image follow the same camera motion model. In contrast, each row of the CMOS camera experiences a different camera motion due to the sequential exposure. Hence, each row of the observed image embeds a unique motion blur due its own observed camera motion. We model the observed image as

$$\mathbf{g}^{(i)} = \frac{1}{t_e} \int_{(i-1)t_d}^{(i-1)t_d + t_e} \mathbf{f}_{\mathbf{p}(t)}^{(i)} dt, \text{ for } i = 1 \text{ to } M, \quad (1)$$

where $\mathbf{f}_{\mathbf{p}(t)}$ is the warped version of \mathbf{f} due to the camera pose $\mathbf{p}(t)$ at a particular time t . The superscript (i) denotes i th row of an image.

In this work, we assume that the scene is planar and is fronto-parallel to the image plane, i.e. all scene points are at a constant depth from the camera. In projective motion blur model,^{1,3,4} the blurred image is represented as weighted combination of multiple warps of a clean reference image. We adapt this model to suit CMOS cameras such that each row of the distorted image is taken from different weighted combinations of warps of the reference image. We consider a camera pose space $\mathcal{S} = \{\boldsymbol{\tau}_k\}$, where $\boldsymbol{\tau}_k$ represents a single 6D camera pose. We represent the continuous time model of (1) using this pose space as a discrete model as given below.

$$\mathbf{g}^{(i)} = \sum_{\boldsymbol{\tau}_k \in \mathcal{S}} \omega_{\boldsymbol{\tau}_k}^{(i)} \mathbf{f}_{\boldsymbol{\tau}_k}^{(i)} \quad (2)$$

where \mathbf{f}_{τ_k} is the warped version of \mathbf{f} due to camera pose τ_k . Unlike $\mathbf{p}(t)$ that represents the path of the camera motion within a row exposure, the camera pose weight vector $\omega_k^{(i)}$ dissolves the time dependency and provides a non-zero weight for each camera pose proportional to the time for which it had stayed during the exposure time. Since the row exposure times of \mathbf{f} and \mathbf{g} are same, we have $\sum_k \omega_k^{(i)} = 1$ for all i .

3. ILLUMINATION ROBUST CHANGE DETECTION

If there are changes in illumination between images \mathbf{f} and \mathbf{g} , (2) does not hold anymore. We model the illumination change as a general linear operation using factors α and β . We write this general model as

$$\mathbf{g}^{(i)} = \alpha^{(i)} \circ \left(\sum_{\tau_k \in \mathcal{S}} \omega_{\tau_k}^{(i)} \mathbf{f}_{\tau_k}^{(i)} \right) + \beta^{(i)} \quad (3)$$

where \circ is the element-wise multiplication operator. Each pixel of the observed image could have unique multiplicative and additive factors. Here $\alpha^{(i)}$ and $\beta^{(i)}$ denote these factors for the row i . Our model is general and different cases of illumination variations are listed in Table 1.

Table 1. Different cases of illumination variation.

Case	Global/Local	Linear/Multiplicative	$\alpha^{(i)}$	$\beta^{(i)}$
I	Global	Multiplicative	\mathbf{a} for all i	$\mathbf{0}$ for all i
II	Global	Linear	\mathbf{a} for all i	\mathbf{b} for all i
III	Local	Multiplicative	\mathbf{a}_i	$\mathbf{0}$ for all i
IV	Local	Linear	\mathbf{a}_i	\mathbf{b}_i

Here $\mathbf{a} = [a, a, \dots, a]^T$ and $\mathbf{b} = [b, b, \dots, b]^T$ are constant vectors, $\mathbf{a}_i = [a_{1i}, a_{2i}, \dots, a_{Ni}]^T$ and $\mathbf{b}_i = [b_{1i}, b_{2i}, \dots, b_{Ni}]^T$ are non-homogeneous vectors and they vary across rows as well. In this paper, we handle multiplicative illumination change models (cases I and III).

In order to detect the changes, *i.e.* the occlusions, between the reference and observed images, we have to account for row-wise motion blur and illumination variations. We propose an optimization problem which simultaneously accounts for motion blur and illumination changes for every row, thereby facilitating the detection of occlusions. We first start with the problem of change detection without any illumination change.

3.1 Solving for Changes

To detect the changes in the scene correctly, we have to segment out the regions of only the changes without picking other areas which are modified due to the presence of motion blur. The registration of images by accounting for blur and detection of changes has to be done jointly. We thus model the changes as an additive factor in (2) and rewrite it as follows:

$$\mathbf{g}^{(i)} = \mathbf{F}^{(i)} \boldsymbol{\omega}^{(i)} + \boldsymbol{\chi}^{(i)} \quad i = 1, 2, \dots, M, \quad (4)$$

We represent (2) in matrix-vector multiplication form with an additional variable $\boldsymbol{\chi}^{(i)}$ denoting the change vector. Each column of $\mathbf{F}^{(i)} \in \mathbb{R}^{N \times |\mathcal{S}|}$ contains the i th row of a warped version of the reference image \mathbf{f} , for a pose $\tau_k \in \mathcal{S}$, and $\mathbf{F}^{(i)} \boldsymbol{\omega}^{(i)}$ chooses multiple warps and combines them according to the pose weight vector $\boldsymbol{\omega}^{(i)}$. $|\mathcal{S}|$ denotes the number of poses in \mathcal{S} . We can write (4) as $\mathbf{g}^{(i)} = \mathbf{B}^{(i)} \boldsymbol{\xi}^{(i)}$ where $\mathbf{B}^{(i)} = [\boldsymbol{\omega}^{(i)}, \boldsymbol{\chi}^{(i)}]^T$ has two parts, the first part taking care of motion blur and the second taking care of the changes. We simultaneously solve for blur and change by solving for $\boldsymbol{\xi}^{(i)}$. We formulate and solve the following optimization problem to arrive at the desired solution.

$$\tilde{\boldsymbol{\xi}}^{(i)} = [\tilde{\boldsymbol{\omega}}^{(i)}, \tilde{\boldsymbol{\chi}}^{(i)}]^T = \arg \min_{\boldsymbol{\xi}^{(i)}} \left\{ \|\mathbf{g}^{(i)} - \mathbf{B}^{(i)} \boldsymbol{\xi}^{(i)}\|_2^2 + \lambda_1 \|\boldsymbol{\omega}^{(i)}\|_1 + \lambda_2 \|\boldsymbol{\chi}^{(i)}\|_1 \right\} \text{ subject to } \boldsymbol{\omega}^{(i)} \succeq 0 \quad (5)$$

where λ_1 and λ_2 are non-negative regularisation parameters and \succeq denotes non-negativity of each element of the vector. ℓ_1 -norm imposes sparsity constraint on number of camera poses in the whole camera pose space and the number of changed pixels.

3.2 Handling illumination variations

In order to address global illumination changes, we consider that the energy captured by one frame is a fraction of the energy captured by the other and thus instead of summing the weights of camera pose vector to 1, we consider $\sum_{\tau_k \in \mathcal{S}} \omega_{\tau_k}^{(i)} = \gamma$, i.e. we refrain from enforcing a sum-to-one constraint on the pose weight vector in (5).

For non-global illumination variations, we propose a procedure which solves for rowwise blur and illumination to detect the changes. Many a time, while capturing aerial images, light could get partially obstructed due to buildings, clouds, or other structures (not present in the field of view). These appear as shadows in the image and result in scaling down of intensities of the underlying pixels leading to local illumination variations. We handle this iteratively by recursively dividing the row into blocks until the block has almost uniform illumination. We start by registering the entire row and calculate the difference which includes occlusion as well as error due to illumination variations. If the number of elements in the difference vector is greater than a set threshold, we divide the block into two. As illumination variations are more spread out compared to the occlusion, this criterion helps us to resolve the ambiguity between illumination variations and occlusion. We follow the same procedure for the split blocks and continue to evaluate each block as before. In order to prevent occlusion from occupying a major portion of the block, we limit the number of iterations by fixing the minimum block length to B_{min} . However, it is possible that there are local illumination variations even within these blocks (which cannot be further divided) that may be wrongly detected as occlusions. To handle this issue, we revisit the occlusion vector and assuming that the illumination variation is constant in the near neighborhood, we apply the weights of the previous block that has been correctly aligned. This helps aligning the portion that needs illumination correction. In the remaining pixels which are not aligned, i.e. where the difference after applying the previous block's weights is greater than a threshold (say ϵ), there are two cases: if the difference value is high, it is assigned as an occlusion, else we align it by estimating the motion using the following optimization problem:

$$\min_{\omega_r^{(i)}} \left\{ \|\mathbf{g}_r^{(i)} - \mathbf{F}_r^{(i)} \omega_r^{(i)}\|_2^2 + \lambda_1 \|\omega_r^{(i)}\|_1 \right\} \quad \text{subject to } \omega_r^{(i)} \succeq 0 \quad (6)$$

Algorithm 1 lists all the steps for local illumination compensation. Here \mathbf{g}_r^i represents the portion of the reference image which needs to be realigned. The corresponding indices are picked from the row and warped to form the columns of $\mathbf{F}_r^{(i)}$. We use $B_{min} = 32$ and $\epsilon = 10$ in Algorithm 1 for all our experiments.

4. EXPERIMENTS

In order to validate the performance of our technique, we show both synthetic and real experimental results for global as well as local illumination variations. We simulate synthetic experiments over a discrete camera path using a scale factor to introduce illumination variations and estimate pose weight vector and occlusion vector while compensating for illumination variations.

4.1 Synthetic Experiments

In order to simulate the effect of global illumination change, we scale the intensities by a constant which is analogous to capturing the same scene at a different instant. Since identical atmospheric and illumination conditions cannot be ensured we emulate distortions in the form of global illumination variations modelled as $\mathbf{a} \circ \mathbf{f}$ (where \mathbf{a} represents the scaling factor and \mathbf{f} is the clean image). We further introduce the effect of rolling shutter and motion blur on this image. To simulate RSMB effect, we generate a discrete camera path of poses of length $(M - 1)\beta + \alpha$ and assign α consecutive poses to each row with a delay of β with respect to the previous row. This assigns i^{th} row a unique set of poses ranging from $(i - 1)\beta + 1$ to $(i - 1)\beta + \alpha$. Each row of the image is warped and averaged according to

Algorithm 1 Change detection in the presence of local illumination variations for i th row.

```

1: Initialize: block = row
2: Estimate pose weight vector  $\omega_{\text{block}}^{(i)}$  and occlusion vector  $\chi_{\text{block}}^{(i)}$  for block
3: Let  $B$  be the length of block
4: Calculate  $\mathbf{d} = \mathbf{g}_{\text{block}}^{(i)} - \omega_{\text{block}}^{(i)} \mathbf{F}_{\text{block}}^{(i)}$ 
5: Calculate  $k = \|\mathbf{abs}(\mathbf{d}) > \epsilon\|_0$ 
6: if  $k > 0.2B$  or  $p/2 \geq B_{\min}$  then
7:   Split block into two, block_l and block_r
8:   Repeat from Step 2 for block_l and block_r
9: else
10:  if  $\|\chi_{\text{block}}^{(i)} > \epsilon\|_0 > 0.1B$  then
11:     $\omega_{\text{block}}^{(i)} = \omega_{\text{prev\_block}}^{(i)}$ 
12:    Realign less difference region  $r$  using (6)
13:  end if
14: end if

```

these poses to obtain the final RSMB image. The centroid of the poses of each row acts as the actual camera path against which our estimates are compared.

In the first example, we introduce new objects to the reference image, followed by global illumination variation. Since, maintaining a straight path of the vehicle is very difficult while capturing aerial images, unwanted vibrations and drifts in flying directions cause in-plane translations and rotations. We factor these effects into consideration while designing the motion path. The image in Fig. 2(a) contains 550 rows and 750 columns. We add new objects and scale the intensity of each pixel of the image by a factor of 0.8 for illumination variations. The resultant image is shown in Fig. 2(b). We generate a camera path with $\alpha = 20$ and β delay of 3. The centroid of t_x and R_z are plotted as shown in Fig. 2(d). The RSMB image generated is shown in Fig. 2(c).

The reference image and the RSMB image are given as inputs to our method, which registers the reference frame to the RSMB image and detects occlusions. To solve (5) we start with the middle row as sufficient information may not be present at the boundaries of the image. We initialize a large pose space around zero motion for the middle row: t_x ranging from $[-12 : 12]$ pixels in steps of 0.5 pixels, $t_y : [-3 : 3]$ pixels and R_z ranging from $[-5 : 5]^\circ$ in steps of 0.5° . The remaining rows consider pose space around the centroid of neighboring rows. The neighborhood around this centroid is predefined; we consider a neighborhood of 3 pixels for t_x , 2 pixels for t_y and 2° for R_z . Within the time period a row is scanned, the camera undergoes very small motion, and hence we consider a small neighborhood around the centroid. The sparsity of camera poses in the entire pose space and occlusion in the image is ensured by setting regularization parameters $\lambda_1 = 5 \times 10^4$ and $\lambda_2 = 10^3$. On solving the camera poses for each row, we get the estimated pose weight vectors and occlusion vectors. We form the registered reference image using $\{\mathbf{F}^{(i)} \tilde{\omega}^{(i)}\}_{i=1}^M$ and the occlusion image using $\{255 \mathbf{I}_N \tilde{\chi}^{(i)}\}_{i=1}^M$. These are shown in Figs. 2(g) and (h), respectively. Fig. 2(i) shows the thresholded binary image. The estimated camera trajectories for R_z and t_y are shown in Figs. 2(e) and (f). Note that the trajectories are correctly estimated by our algorithm.

In the next experiment, we consider the case of change detection in the presence of local illumination variations, rolling shutter effect and motion blur. This effect is simulated by considering a template of size same as that of image. It has value 1 over the entire image except in a patch where the value of the illumination factor is 0.8. The patch considered in this experiment is a sheared rectangle representing shadows of tall buildings falling on the scene where the camera is focused. This template is multiplied element-wise to the occluded image to which we further introduce the effect of rolling shutter and motion blur. We generate a camera path with $\alpha = 25$ and a β delay of 3. The image obtained as a result of these combined effects is shown in Fig. 3(c).

The clean image in Fig. 3(a) and locally illuminated RSMB image in Fig. 3(c) are given as input to the recursive algorithm which detects occlusion while compensating for local illumination variations. As there is no prior information about the camera motion, we consider a large pose space around the

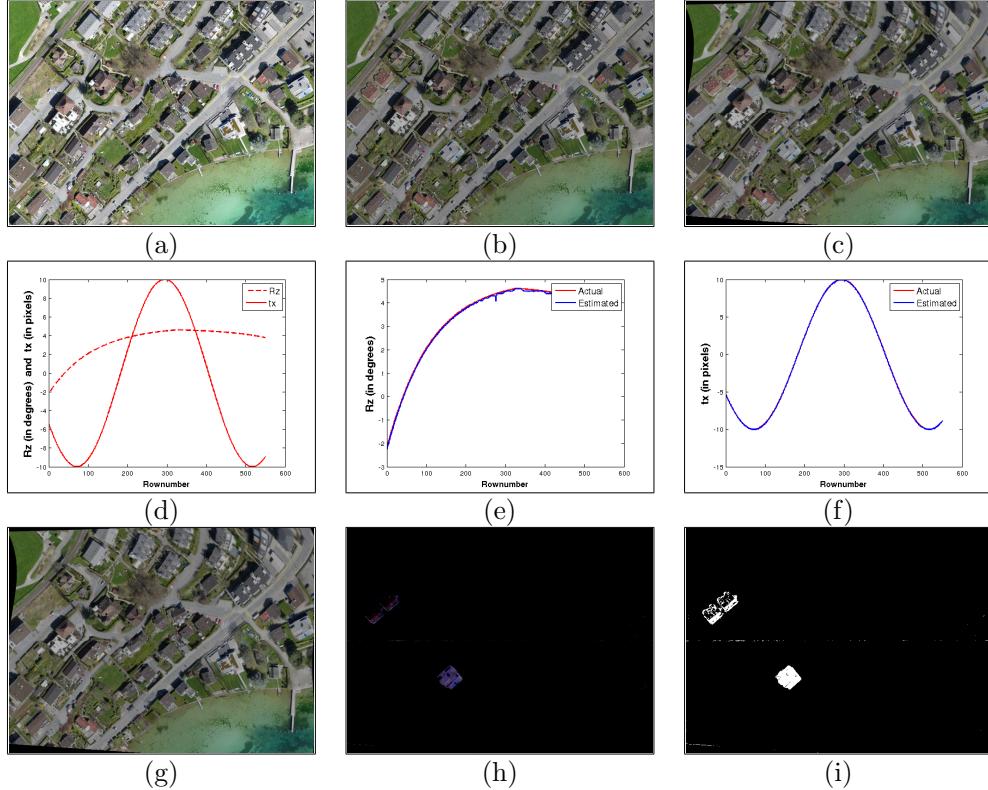


Figure 2. (a) Reference image with no camera motion, (b) Reference image with added occlusions and illumination variation, (c) RSMB image illumination variation, (d) Simulated camera path, (e) Estimated R_z camera path (blue) overlaid on simulated camera path (red), (f) Estimated t_y camera path (blue) overlaid on simulated camera path (red), (g) Registered reference image, (h) Occlusion image, and (i) Thresholded occlusion image.

origin t_x ranging from $[-12 : 12]$ pixels in steps of 0.5 pixels and t_y ranging from $[-5 : 5]$ pixels. For the remaining rows, a pose space defined by a neighborhood of 2 for t_x and 1 for t_y around the centroid of the previous row is considered. We follow the steps in Algorithm 1 to register the row by compensating for illumination through iterative splitting. The registered and illumination compensated image is shown in Fig. 3(d). The occlusion has been correctly detected in Fig. 3(f).

4.2 Real Experiments

To capture real images, we use a Nexus 7 mobile phone to capture images affected by rolling shutter effect and motion blur. In the first experiment, we consider a scene looking down from the top of a building. The reference image is captured around noon with a static camera, whereas the second image which contains new objects is captured during evening with an intentional handshake to emulate aerial distortion. The reference image is shown in Fig. 4(a), and the distorted image is shown in 4(b). Observe that the vertical line in the reference is curved in the second image. The distorted image also has motion blur. There is visible illumination variation between the two images due to the time of capture. These two frames are considered as input to our algorithm which compensates for illumination variation. We choose the following pose space for the middle row registration: translations t_x and t_y ranges are $[-12 : 12]$ pixels, and rotation r_z range is $[-2 : 2]^\circ$. The neighborhood for other rows are 2 pixels for translations and 1° for rotation around the centroid of the estimated motion of the neighboring row. Our method successfully estimates the motion and the registered image is shown in Fig. 4(c). The sum of weight vector handles the illumination change row by row and is shown in Fig. 4(e). We can deduce that the illumination change in this case is almost global since the sum of weight vector is almost same (around 0.65) for all rows. The detected occlusions are shown in Fig. 4(d).

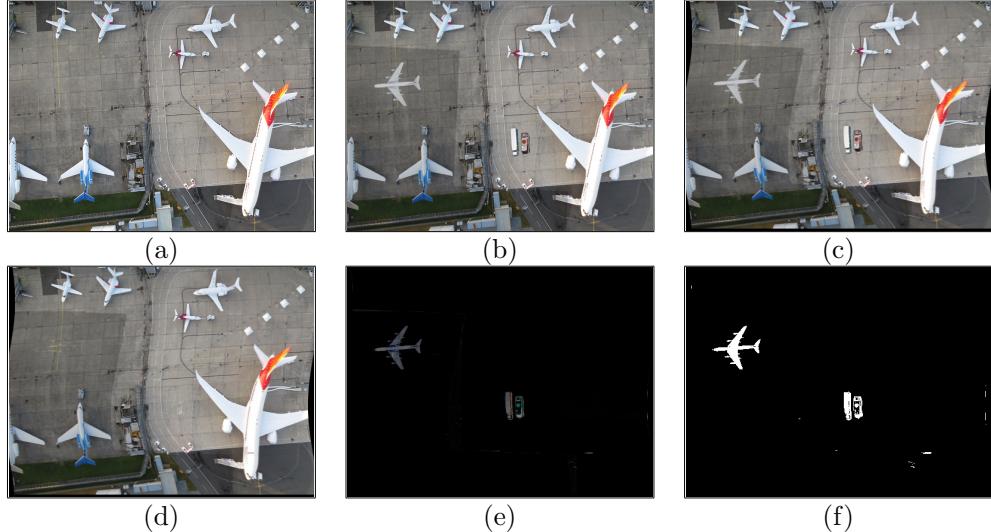


Figure 3. (a) Reference image with no camera motion, (b) Reference image with added occlusions and illumination variation, (c) RSMB image illumination variation, (d) Registered reference image, (e) Occlusion image, and (f) Thresholded occlusion image.



Figure 4. (a) Reference image with no camera motion, (b) Distorted image due to camera motion and illumination variation in the presence of occlusions, (c) Registered image by our algorithm, (d) Detected occlusions, and (e) Adaptation of illumination change through the pose weight vector.

In the second scenario, we consider the case of a local illumination change. The captured reference and distorted images are shown in Figs. 5(a) and (b), respectively. Rolling shutter effect can be observed by the curved edges of the green pen and the blue box. The new red object is present under a shadow region. A simple registration would ignore the presence of shadow and would wrongly detect it as a change too. Our recursive illumination adaptative framework successfully registers the local illumination changes and the camera motion, and the registered image is shown in Fig. 5(c). The detected occlusions are shown in Fig. 5(d) and are verifiably correct yet again.

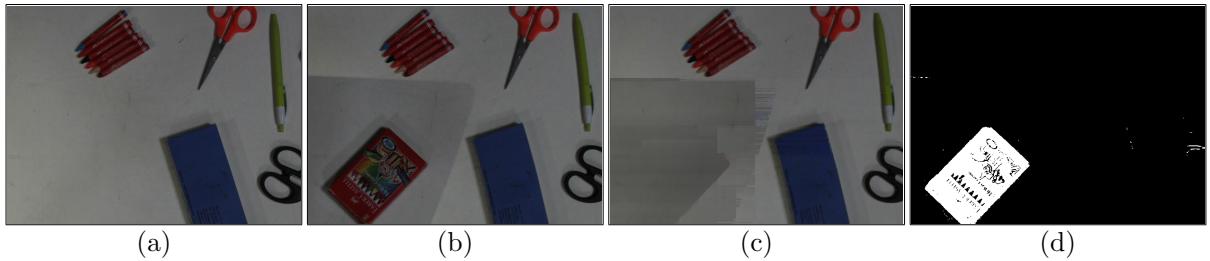


Figure 5. (a) Reference image, (b) Distorted image due to camera motion and local illumination variation in the presence of occlusion, (c) Registered image by our algorithm, (d) Detected occlusions.

5. CONCLUSIONS

In this paper, we considered the problem of change detection of images captured using CMOS cameras. We formulated a joint model to handle motion blur, rolling shutter effect, and illumination variation, and to simultaneously detect occlusions. We also proposed an algorithm to handle local illumination variations such as shadows in the distorted image. We showed experimental results for both local and global illumination variations with rolling shutter and motion blur distortions. Our method was shown to be effective on both synthetic and real data.

ACKNOWLEDGMENTS

A part of this work was supported by a grant from Asian Office of Aerospace Research and Development and Air Force Office of Scientific Research. The support is gratefully acknowledged. The results and interpretations presented in this paper are that of the authors, and do not necessarily reflect the views and priorities of the sponsor, or the US Air Force Research Laboratory.

REFERENCES

- [1] Whyte, O., Sivic, J., Zisserman, A., and Ponce, J., “Non-uniform deblurring for shaken images,” *International Journal of Computer Vision* **98**(2), 168–186 (2012).
- [2] Hu, Z. and Yang, M.-H., “Fast non-uniform deblurring using constrained camera pose subspace,” in [*Proc. British Machine Vision Conference*], 1–11 (2012).
- [3] Paramanand, C. and Rajagopalan, A., “Shape from sharp and motion-blurred image pair,” *International Journal of Computer Vision* **107**(3), 272–292 (2014).
- [4] Tai, Y.-W., Tan, P., and Brown, M. S., “Richardson-lucy deblurring for scenes under a projective motion path,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(8), 1603–1618 (2011).
- [5] Xu, L., Zheng, S., and Jia, J., “Unnatural l0 sparse representation for natural image deblurring,” in [*Computer Vision and Pattern Recognition, IEEE Conference on*], 1107–1114, IEEE (2013).
- [6] Liang, C.-K., Chang, L.-W., and Chen, H. H., “Analysis and compensation of rolling shutter effect,” *IEEE Transactions on Image Processing* **17**(8), 1323–1330 (2008).
- [7] Baker, S., Bennett, E., Kang, S. B., and Szeliski, R., “Removing rolling shutter wobble,” in [*Computer Vision and Pattern Recognition, IEEE Conference on*], 2392–2399, IEEE (2010).
- [8] Ringaby, E. and Forssén, P.-E., “Efficient video rectification and stabilisation for cell-phones,” *International Journal of Computer Vision* **96**(3), 335–352 (2012).
- [9] Grundmann, M., Kwatra, V., and Essa, I., “Auto-directed video stabilization with robust l1 optimal camera paths,” in [*Computer Vision and Pattern Recognition, IEEE Conference on*], 225–232, IEEE (2011).
- [10] Rengarajan, V., Rajagopalan, A., and Rangarajan, A., “Change detection in the presence of motion blur and rolling shutter effect,” in [*Computer Vision - ECCV 2014*], *LNCS* **8695**, 123–137, Springer (2014).
- [11] Gijsenij, A., Gevers, T., and Van De Weijer, J., “Computational color constancy: Survey and experiments,” *Image Processing, IEEE Transactions on* **20**(9), 2475–2489 (2011).
- [12] Dai, X. and Khorram, S., “The effects of image misregistration on the accuracy of remotely sensed change detection,” *Geoscience and Remote Sensing, IEEE Transactions on* **36**(5), 1566–1577 (1998).
- [13] Phong, B. T., “Illumination for computer generated pictures,” *Communications of the ACM* **18**(6), 311–317 (1975).