

## AUTOMATED BLUR DETECTION AND REMOVAL IN AIRBORNE IMAGING SYSTEMS USING IMU DATA

C. A. Shah, W. Schickler

Microsoft Corporation, Bing Imagery Technologies, R&D, Boulder, CO 80305, USA  
(chintans, woschick)@microsoft.com

Commission I/5

**KEY WORDS:** Blur Detection; Blur Removal, Inertial Measurement Unit, Automated Quality Control.

### ABSTRACT:

Images acquired by airborne sensors exhibit blur due to turbulent motion experienced by the aircraft. Significant amount of blur renders the images unusable for subsequent visual/automated analysis, requiring a re-flight. This necessitates quantifying the amount of blur in an image. Most approaches to blur quantification use image based methods to estimate the MTF (Modular Transfer Function) that indicates the amount of blur. Their limitations are – (1) MTF calculation requires presence of straight edges in the image scene, which may not always be available, (2) Due to the absence of an ideal edge, the amount of blur estimated is relative, and (3) It is computationally expensive and therefore may not be practical for blur detection in real time applications. Our solution uses the sensor motion as measured by an Inertial Measurement Unit (IMU) mounted in the camera system to calculate the motion experienced by the aircraft and the sensor during the time the shutter was actually open. This motion information together with the blur detection algorithm presented in this paper can provide an accurate quantification of blur in pixel units. Once we identify the images that exceed a given blur threshold, we use a blur removal algorithm that uses the IMU data and a natural image prior to compute per-pixel, spatially-varying blur, and then de-convolves an image to produce de-blurred images.

The presented blur detection and removal methods are currently being used offline to quantify and remove the blur from images acquired by the UltraCam systems within the Global Ortho Program (Walcher, 2012), which generates ortho imagery for all of the continental US as well as Western Europe, from over 2.5 million images. Furthermore, the blur detection method will be incorporated in the camera software of all our operational and forthcoming UltraCam imaging systems for real-time blur quantification.

## 1. INTRODUCTION

### 1.1 Motivation

Many services and applications provide functionality associated with images captured by cameras. An online mapping service may provide users with interactive maps derived from images captured by cameras mounted on aircrafts, vehicles, and/or satellites. A web search engine may provide users with search results comprising one or more images captured by cameras. Unfortunately, many images acquired by cameras may suffer from blur. For example, blur may occur due to motion of a camera while an image is captured (e.g., a camera mounted to an aircraft may experience three-dimensional movement due to turbulent motion experience by the aircraft during image capture). Substantial blur within an image may render the image unusable for certain applications/functions.

### 1.2 Overview

We use data coming from the Inertial Measurement Unit (IMU) integrated into the camera system and our newly developed algorithm, to quantify the amount of blur in an image. IMU measures the angular motion in the three major axis roll, pitch, and heading of the camera at any given instance of time. It measures these parameters at a very high frequency. The IMU system we are using operates at 200Hz. We use this data to estimate the camera orientation at the beginning and end of an exposure event (the time from opening to closing of a shutter during which an image is captured, usually in milliseconds),

from which we can estimate the amount of angular displacement experienced by the camera in a 3 dimensional space. This combined with the camera intrinsic (focal length and the principal point of the camera) parameters, can give us an exact measurement of the amount of blur at any pixel location within an image.

## 2. IMAGE BLUR DETECTION

### 2.1 Definition of terms

#### *Start and end time of exposure*

The effective length of time during which the camera shutter is open for a particular image is referred to as the exposure time. We will call the times at which the camera shutter opens and closes as  $t_s$  and  $t_e$ , respectively.

#### *Inertial Measurement Unit (IMU) measurements*

An IMU measures the angular orientation of a camera in terms of the roll, pitch, and heading angles.

Roll ( $\phi$ ): Angular rotation along the axis passing through nose to tail of a plane

Pitch ( $\omega$ ): Angular rotation along the axis passing through the plane from wingtip to wingtip

Heading ( $\kappa$ ): Angular rotation along the vertical axis

#### *Camera Rotation Matrix*

An IMU records the camera orientation angles frequently (~200Hz) during the entire flight path. Using an interpolation technique, we can estimate the respective camera orientation angles at  $t_s$  and  $t_e$  as:  $\theta_s = [\omega_s, \phi_s, \kappa_s]^T$  and  $\theta_e = [\omega_e, \phi_e, \kappa_e]^T$ .

The net change in orientation angles between  $t_s$  and  $t_e$  is then calculated by taking the angular difference  $\theta_d = \theta_e - \theta_s$ . Using which, we can calculate the three dimensional camera rotation matrix  $R(\theta_d)$  as

$$R = R(\theta_d) = \begin{bmatrix} \cos \phi_d \cos \kappa_d & \sin \omega_d \sin \phi_d \cos \kappa_d + \cos \omega_d \sin \kappa_d & \sin \omega_d \sin \kappa_d - \cos \omega_d \sin \phi_d \sin \kappa_d \\ -\cos \phi_d \sin \kappa_d & \cos \omega_d \cos \kappa_d - \sin \omega_d \sin \phi_d \sin \kappa_d & \cos \omega_d \sin \phi_d \sin \kappa_d + \sin \omega_d \cos \kappa_d \\ \sin \phi_d & -\sin \omega_d \cos \phi_d & \cos \omega_d \cos \phi_d \end{bmatrix}$$

### Camera Intrinsic Matrix

A generalized camera intrinsic matrix ( $C$ ) is defined using the camera focal length ( $f$ ) and principal point ( $x_o, y_o$ ), as:

$$C = \begin{bmatrix} f & 0 & x_o \\ 0 & f & y_o \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

A camera intrinsic matrix is used to convert from ground coordinates ( $X, Y, Z$ ) to image plane co-ordinates ( $x, y$ ) as:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & x_o \\ 0 & f & y_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (2)$$

In the above equation,  $f$  is expressed in pixel units and  $x_o$  and  $y_o$  correspond to the co-ordinates of the center pixel of an image.

## 2.2 Modelling image blur caused by angular motion

The intensity of an image at a pixel location ( $x, y$ ) is given as

$$I(x, y) = \int_{t_s}^{t_e} \Gamma(X, Y, Z) dt \quad (3)$$

Where  $\Gamma(X, Y, Z)$  corresponds to the surface reflectance at the ground coordinates ( $X, Y, Z$ ). However, any motion in the camera as a function of time, will cause it to integrate the surface reflectance from continuously changing ground co-ordinates

$$I(x, y) = \int_{t_s}^{t_e} \Gamma(X + \Delta X(\theta_t), Y + \Delta Y(\theta_t), Z + \Delta Z(\theta_t)) dt \quad (4)$$

Here  $\Delta X$ ,  $\Delta Y$ , and  $\Delta Z$  correspond to the change in spatial location as a function of camera orientation at a given time. This leads to an image with blurred appearance, with the blur amount being directly proportional to the amount of spatial displacement. In order to quantify the amount of blur experienced by a pixel, one could calculate the trajectory traveled by a camera starting from initial ground co-ordinates ( $X_{t_s}, Y_{t_s}, Z_{t_s}$ ) to the ground co-ordinates ( $X_{t_e}, Y_{t_e}, Z_{t_e}$ ) at the closing of the shutter. For simplicity, we approximate this trajectory with a straight line, measured between the start and the end co-ordinates. Also, instead of operating in the ground space, we solve this problem in the image space. For a given pixel co-ordinates ( $x_{t_e}, y_{t_e}$ ) in the resulting image, our goal is to find the pixel co-ordinates ( $x_{t_s}, y_{t_s}$ ) at the opening of the shutter. The motion blur experienced by this pixel during the exposure time is then calculated as:

$$b(x_{t_e}, y_{t_e}) = \sqrt{(x_{t_e} - x_{t_s})^2 + (y_{t_e} - y_{t_s})^2} \quad (5)$$

This calculation can then be done for the four corners of an image, and the motion blur for the entire image is defined as:

$$B = \max_{(x_{t_e}, y_{t_e})} b(x_{t_e}, y_{t_e}). \quad (6)$$

## 2.3 Quantifying image blur caused by angular motion

Let  $C_{t_s}$  define the camera intrinsic matrix at the opening of the shutter. Then for a given image plane coordinates ( $x_{t_s}, y_{t_s}$ ), using equation (2), we can calculate the ground co-ordinates as:

$$\begin{bmatrix} X_{t_s} \\ Y_{t_s} \\ Z_{t_s} \end{bmatrix} = C_{t_s}^{-1} \begin{bmatrix} x_{t_s} \\ y_{t_s} \\ 1 \end{bmatrix} \quad (7)$$

However, due to the rotational motion of the camera defined by  $R$ , the point on the ground imaged at time  $t_e$  is given by:

$$\begin{bmatrix} X_{t_e} \\ Y_{t_e} \\ Z_{t_e} \end{bmatrix} = R \begin{bmatrix} X_{t_s} \\ Y_{t_s} \\ Z_{t_s} \end{bmatrix} \quad (8)$$

Finally, the corresponding ground point will be imaged at the location ( $x_{t_e}, y_{t_e}$ ) given as:

$$\begin{bmatrix} \hat{x}_{t_e} \\ \hat{y}_{t_e} \\ \hat{z}_{t_e} \end{bmatrix} = C_{t_e} \begin{bmatrix} X_{t_e} \\ Y_{t_e} \\ Z_{t_e} \end{bmatrix} \quad (9)$$

Such that  $x_{t_e} = \frac{\hat{x}_{t_e}}{\hat{z}_{t_e}}$  and  $y_{t_e} = \frac{\hat{y}_{t_e}}{\hat{z}_{t_e}}$ .

Thus, combining equations (7), (8), and (9), we obtain:

$$\begin{bmatrix} \hat{x}_{t_e} \\ \hat{y}_{t_e} \\ \hat{z}_{t_e} \end{bmatrix} = C_{t_e} R C_{t_s}^{-1} \begin{bmatrix} x_{t_s} \\ y_{t_s} \\ 1 \end{bmatrix} \quad (10)$$

Also, since the camera intrinsic matrix remains the same at  $t_e$  and  $t_s$  ( $C_{t_e} = C_{t_s} = C$ ), the above equation can be re-written as:

$$\begin{bmatrix} \hat{x}_{t_e} \\ \hat{y}_{t_e} \\ \hat{z}_{t_e} \end{bmatrix} = C R C^{-1} \begin{bmatrix} x_{t_s} \\ y_{t_s} \\ 1 \end{bmatrix} \quad (11)$$

Given an image plane co-ordinates ( $x_{t_e}, y_{t_e}$ ) at  $t_e$ , it is trivial to estimate its corresponding image plane co-ordinates ( $x_{t_s}, y_{t_s}$ ) at  $t_s$  as:

$$\begin{bmatrix} \hat{x}_{t_s} \\ \hat{y}_{t_s} \\ \hat{z}_{t_s} \end{bmatrix} = (C R C^{-1})^{-1} \begin{bmatrix} x_{t_e} \\ y_{t_e} \\ 1 \end{bmatrix} \quad (12)$$

Where  $x_{t_s} = \frac{\hat{x}_{t_s}}{\hat{z}_{t_s}}$  and  $y_{t_s} = \frac{\hat{y}_{t_s}}{\hat{z}_{t_s}}$ .

## 3. IMAGE BLUR REMOVAL

The algorithm for image blur removal has been described in great detail in (Joshi, 2010). The basic principle is to use the IMU data in conjunction with a corresponding blurry input image and a natural image prior to compute per-pixel, spatially-varying blur and then perform blind de-convolution to produce a sharper image.

This method is fully automatic and has no user-tuned parameters. Also, by incorporating the IMU data, it has shown to outperform purely image-based de-blurring methods as blind de-convolution is an inherently ill-posed problem.

## 4. RESULTS AND EXAMPLES

### 4.1 Blur Detection

The proposed method for blur detection has been applied within Microsoft's Global Ortho Program (Walcher 2012). Over 20,000 images have been processed and verified through manual visual inspection. The success rate for blur detection was 99.99%, the false positive rate was 0%. The false negative rate was 0.01%. All of the cases where motion blur could not be detected correctly have been traced back to corrupt IMU data. As of April 2012 a total of 2.5 million images have been processed successfully with the blur detection algorithm.

The following figure (1) shows the roll and pitch for a time period with three camera events, during a turbulent flight mission. The image shown has image blur of 4.2 pixels due to a roll and pitch angular motion.

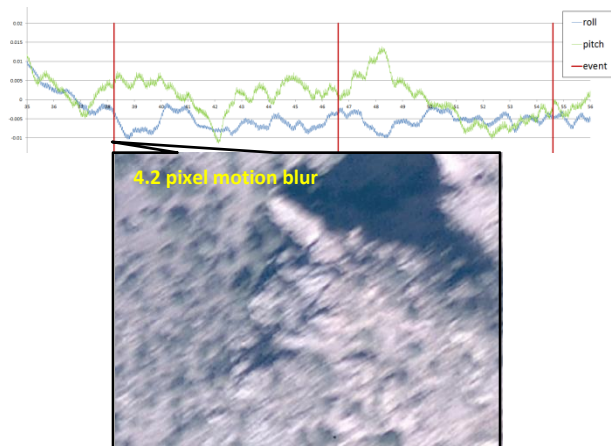


Figure 1 Roll and pitch graph and corresponding image with angular motion blur

### 4.2 Blur Removal

The proposed method for blur removal has been applied within Microsoft's Global Ortho Program (Walcher, 2012) for all images that exceed an angular motion blur of more than 2 pixels. Results of the proposed blur removal algorithm vary depending on the initial amount of blur. Images with 2 – 4 pixel motion blur can be recovered to a quality where the motion blur is almost undetectable. Images with 4 – 7 pixel motion blur can be improved to a point where the blur is still visible but the image meets quality spec. Images with more than 7 pixel motion blur only showed minor improvements.



Figure 2 Subsection of an image with 4.1 pixel motion blur (left) and the de-blurred image result (right)



Figure 3 Subsection of an image with 6.9 pixel motion blur (left) and the de-blurred image result (right). De-blurred image still looks blurry and minor ringing artefacts are visible in the shadow concrete area. Edges have significantly improved.

## 5. CONCLUSIONS

The proposed metrics for automatic blur detection has not only proven to be a crucial part of our automated quality control, but has been instrumental in defining image quality specifications with regards to image blur for our customers. Both are paramount when dealing with large photogrammetric ortho projects such as the Global Ortho Program (Walcher, 2012).

The proposed algorithm for blur detection will be implemented into the UltraCam sensor family. Having the blur detection directly in the sensor will allow detecting circumstances in real time where the acquisition conditions are too turbulent to meet high quality imagery.

The proposed method for blur removal has saved thousands of dollars in acquisition cost, because we were able to recover imagery with motion blur that would have been subject to rejection due to poor image quality otherwise.

## 6. REFERENCES

### Walcher, 2012

Wolfgang Walcher, Michael Gruber, Franz Leberl, *The Microsoft Global Ortho Program*. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Melbourne, Australia.

### Joshi, 2010

Joshi, N., Kang, S.B., Zitnick, L., and Szeliski, R., *Image Deblurring with Inertial Measurement Sensors*. ACM SIGGRAPH.