Space-variant image deblurring on smartphones using inertial sensors

Ondřej Šindelář, Filip Šroubek
Institute of Information Theory and Automation, ASCR
Pod Vodarenskou vezi 4, 182 08, Prague 8, Czech Republic
Email: sroubekf@utia.cas.cz

Peyman Milanfar
University of California, Santa Cruz
Electrical Engineering Dept., Santa Cruz CA 95064, USA
Email: milanfar@ucsc.edu

***Abstract*—Low-light hand-held photography requires long exposures and leads to space-variant blur degradation. Removing blur without any information about the camera motion is a computationally demanding and unstable process. In this demo system, we use rotational inertial sensors (gyroscopes) to detect the motion trajectory of the camera during exposure and then use it as a base for removing blur from the acquired photographs. The demo is a close-to-real-time deblurring technology, implemented on an Android smartphone.**

***Index Terms*—space-variant deconvolution; gyroscope; mobile phone;**

I. Introduction

Image stabilization (IS) technology, common in modern cameras, can compensate only for motion of a small extent and speed. In this work we target mobile phones which are rarely equipped with IS. Deblurring the image offline using mathematical algorithms is often the only option.

Arbitrary camera motion blur can be modeled by space-variant (SV) convolution and the deblurring process is referred to as SV deconvolution [1]. Camera motion blur is SV for several reasons. First, it is caused by the camera projection itself. Phone cameras are usually equipped with wide-angle lenses (field of view around 60°), which distort objects close to image borders. The blur caused by rotation around x and y axes is therefore different in the image center and borders. The SV blur are particularly noticeable when rotation around z axis is significant. Second, the camera-object distance influences the blur caused by camera translation and the knowledge of depth map is thus necessary. However, phone cameras have a focal length of a few millimeters and the scene projected into the camera image plane moves by less then a pixel if the objects are more then 2m away, so the camera translation in such cases can be neglected [2]. We will thus focus on purely rotational motion of the camera, which has several additional advantages. Gyroscopes are sufficiently accurate for angular speed estimation but drift. We use gyroscope data to estimate rotation and compensate for a drift by either calibration of a still camera or considering accelerometers during motion.

Another reason for SV blur, unrelated to camera motion but intrinsic to camera hardware design, is rolling shutter [3]. In image sensors on mobile devices, contrary to systems with mechanical shutter, values of illuminated pixels are read successively line by line while the sensor is exposed to light. The readout from the CMOS sensor takes several tens of milliseconds, which results in a picture not taken at a single moment, but with a slight time delay between the first and last row of pixels. The rolling shutter effect is therefore another cause of space variance as the blur depends on the vertical position in the image. To model accurately the blur at every position, it is necessary to shift the exposure-time window in which the gyroscope data are fetched according to the vertical position.

Our work demonstrates the use of built-in inertial sensors in smartphones for accurate blur estimation. The proposed solution is simple and practical. It removes blur induced by camera rotation and simultaneously overcomes rolling-shutter effect, which, to our knowledge, has not been considered in the deconvolution problem before. As a testing platform we have chosen a Samsung Galaxy S II smartphone with Android operating system.

A similar system was proposed by Joshi *et al.* in [4] but they have designed an expensive measuring apparatus consisting of a DSLR camera and an external inertial module, and perform image deblurring offline on a computer. Contrary to low-cost cameras, rolling shutter is not present in DSLR cameras. Sindelar *et al.* [2] tested simple deconvolution running on smartphones, but they have considered only space-invariant blur, which limits applicability of their solution.

II. The Demo System

The tested device is equipped with all the apparatus needed for our demo system, namely a relatively high-quality camera, inertial sensors, fast CPU (ARM Cortex-A9) and enough RAM to perform computations. A block diagram of the deblurring application is in Fig. 1.

We first perform offline calibration to obtain camera intrinsic parameters, rolling shutter delay and gyroscope drift.

During the photo acquisition, samples of angular velocity are recorded using the embedded gyroscopes, which are afterwards trimmed to match the exposure period. Integrating the position track from the recorded gyroscope data allows us to render a correct blur at every pixel of the image. To perform full image deblurring with SV blur would be computationally very expensive and not feasible on a mobile device. Instead, we split the image into overlapping patches and generate one blur for each patch. We use a division to 6×8 squares with 25%

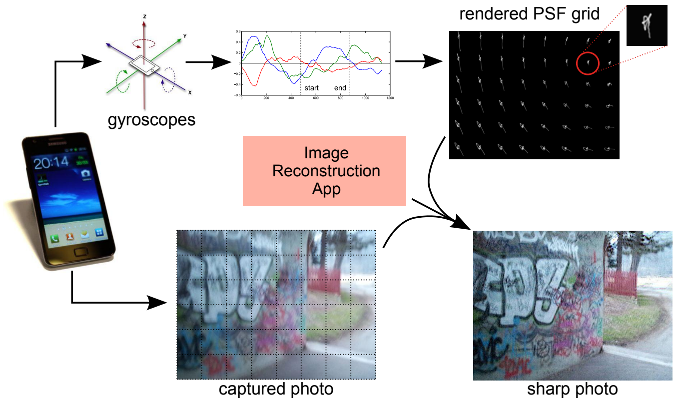

Fig. 1. The block diagram of the smartphone application: During camera exposure, the application records data from the built-in gyroscopes. The data are processed and blurs are estimated. The captured photo is divided into overlapping patches, Wiener deconvolution is performed on every patch and the reconstructed patches are blended to generate the sharp photo. The whole process, entirely done on the smartphone, takes around 10s.

Fig. 2. Examples of captured and reconstructed images using our demo system. Best viewed on a computer screen and zoomed in.

overlap in every directions. Each patch is then reconstructed individually using the Wiener filter and the corresponding blur. To avoid ringing artifacts around patch borders, edge tapering is applied prior to filtering. Due to patch overlaps, we blend the reconstructed patches by weighting them with Hamming windows, which results in virtually seamless images.

The intensity values of the reconstructed image sometimes lie outside the working bit-depth range (0-255), therefore we added optional normalization with clipping of outliers. The normalization is especially useful in the case of larger blurs and scene with high luminance.

For the Fourier transformation, we use the FFTW library ported to ARM CPUs, supporting two cores and a SIMD instruction set (NEON). FFTW proved to be remarkably fast on the tested smartphone.

The acquired images with native camera resolution of 3264×2448 are by default scaled down to 2048×1536 to take advantage of better performance of FFTW when the image size is a factor of small primes.

The Wiener filtering consists of several FFTs: one for the blur and two (forward and backward for inverse) for each color channel. That yields a total of 7 FFT operations for each patch. The deconvolution of the image enlarged by the overlaps takes about 7s; the whole process starting from the camera shutter is done in a little over 10s. This includes image resizing, blur estimation, compressing and saving the original and deblurred image files.

We have identified several issues that hamper our solution. Correct synchronization of camera shutter with the gyroscope samples is critical. Even a few millisecond error can produce annoying artifacts. We managed to find a good synchronization mechanism for our test device, which will be unfortunately hard to port to other models, because Android provides no general aid for precise camera handling. Gyroscope drift is substantial and without any compensation results in a biased blur estimator. Correct calibration is still an open question. Internal image post-processing done by the phone presents another serious problem for deconvolution. Since the original raw data from the image sensor are not available, we are forced to work with JPEG (compressed) images, which are processed by gamma correction and most likely also by unidentifiable image enhancement steps. We have employed the inversion of gamma correction, which indeed improves the results to some degree. Three examples of performance are in Fig. 2. See more examples and a demo video in supplementary materials.

Acknowledgment

This work was supported in part by the Academy of Sciences of the Czech Republic under project M100751201 and by the Grant Agency of the Czech Republic under project GA13-29225S.

References

[1] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce, "Non-uniform deblurring for shaken images," *International Journal of Computer Vision*, vol. 98, no. 2, pp. 168–186, 2012.

[2] O. Šindelář and F. Šroubek, "Image deblurring in smartphone devices using built-in inertial measurement sensors," *Journal of Electronic Imaging*, vol. 22, no. 1, pp. 011 003–011 003, 2013.

[3] P. Forssen and E. Ringaby, "Rectifying rolling shutter video from hand-held devices," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 507–514.

[4] N. Joshi, S. B. Kang, C. L. Zitnick, and R. Szeliski, "Image deblurring using inertial measurement sensors," *ACM Trans. Graph.*, vol. 29, pp. 30:1–30:9, July 2010. [Online]. Available: http://doi.acm.org/10.1145/1778765.1778767