

High Resolution and Frame Rate Video through Fusion of Lower Quality Videos

Project Proposal for EE367 Project

Linus Hein
Stanford

1. Motivation

Modern smartphone camera hardware and software have advanced significantly in recent years. One new feature added to these cameras is the ability to capture slow-motion videos. In this mode, the cameras will usually reduce the image resolution (down from a vertical resolution of 2160p to maybe 720p) in exchange for a higher frame rate which can be as high as 240 Hz. I had the idea of downsampling this stream of images in the time dimension to achieve better results in the spatial dimensions, either by deblurring or by superresolution. This idea reduces to fusing multiple images of the same resolution and sampling frequency.

Prof. Wetzstein then suggested also using a second, slow high-resolution camera as an additional source of information to generate a high-frequency high-resolution stream of images. This would enable capturing high-quality video at a high frame rate on hardware that does not support both of these things simultaneously natively.

2. Related Work

Removing motion blur in images is a known problem in the literature [2][4]. Most approaches assume only a single image to be available, making the process of extracting the motion blur kernel quite difficult. One common approach is to estimate the motion blur kernel (e.g., through machine learning) to then apply a deconvolution algorithm on the input image.

Assuming there to be a stream of images/a video, this task should become simpler as one can analyze movements in a video better than in a still image. There have also been publications on using multiple images in a video stream to either achieve a single deblurred image [1] or to deblur each frame individually [3].

Another approach by Tai et al. [5] used a “hybrid camera” which consists of a low-frequency high-resolution camera and a high-frequency low-resolution camera. They used the information from the high-frequency camera to estimate the motion kernel for the images taken by the low-frequency camera and deblur these images. Their output was a low-frequency high-resolution deblurred video.

3. Project Overview

In my project, I want to take a similar approach as Tai et al. [5], with the difference being that I want to arrive at a high-resolution high-frequency video. My basic idea is that the high-frequency video will yield a trajectory (=motion blur kernel) of every part of the image between each of the high-frequency images, similar to the approach by Ren et al. [3]. Since you can use this trajectory to estimate the motion blur kernel for each of the low-frequency frames, I believe it should be possible to also compute the kernels that represent the image that would have been taken by the high-resolution camera in between two frames.

As an example: assume the low-frequency camera takes one image per second, while the high-frequency camera takes four images per second. In this case, I intend to estimate the motion blur kernel for each area for the one-second exposure by using the four images taken in the same second. Reversing the motion convolution for every pixel will result in a deblurred high-resolution image. Then, using the estimated trajectory/motion, instead of a motion blur kernel for the entire one-second frame (which smears the motion across the kernel), I can also create intermediate kernels that represent the picture that the low-frequency camera would have taken 0.25 s, 0.5 s, and 0.75 s into the one-second exposure (using pointwise kernels instead of smeared kernels). Applying these kernels to my deblurred image and concatenating the frames should result in a high-frequency high-resolution video.

4. Milestones and Timeline

- (1 week) Record test data: create a test rig consisting of two smartphones (ideally identical model) taking a video simultaneously. The video should contain a moving feature¹ to synchronize the two video streams manually later on. There should also be a second fast-moving object with a lot of details in the frame. Synchronize the videos from the two phones manually.
- (1.5 weeks) Experiment with extracting the trajectory from videos. I intend to heavily rely on the work by Ren et al. [3]. In case this does not work out, I would attempt feature detection and pairing (possibly SIFT features) to detect movement in images. This should result in a trajectory.
- (0.75 weeks) Write code to deconvolve the motion from the high-resolution image → deblurred high-resolution images.
- (0.75 weeks) Use the trajectory between the deblurred high-resolution low frame rate frames to generate kernels which when convolved with the frames result in the interpolated images.

References

- [1] B. Bascle, A. Blake, and A. Zisserman. Motion deblurring and super-resolution from an image sequence. In *European conference on computer vision*, pages 571–582. Springer, 1996.
- [2] J. Jia. Single image motion deblurring using transparency. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.
- [3] W. Ren, J. Pan, X. Cao, and M.-H. Yang. Video deblurring via semantic segmentation and pixel-wise non-linear kernel. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1077–1085, 2017.
- [4] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. *Acm transactions on graphics (tog)*, 27(3):1–10, 2008.
- [5] Y.-W. Tai, H. Du, M. S. Brown, and S. Lin. Image/video deblurring using a hybrid camera. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

¹I am thinking of using a spinning wheel of a bicycle with a visual mark on the side.