Restoration of Out-of-focus Lecture Video by Automatic Slide Matching

Ngai-Man Cheung, David Chen, Vijay Chandrasekhar, Sam S. Tsai,
Gabriel Takacs, Sherif A. Halawa, and Bernd Girod
Department of Electrical Engineering, Stanford University, Stanford, CA 94305
{ncheung, dmchen, vijayc, sstsai, gtakacs, halawa, bgirod}@stanford.edu

ABSTRACT

Restoring the fine detail in the slide area of a defocused lecture video is a challenging task. In this work, we propose to use clean images of slides available along with the defocused lecture video to help the restoration. Our proposed method uses local feature descriptors and multiple defocused slide decks to automatically identify the slide that is displayed in the defocused frame. We then use the matching slide as side information to estimate the parameters for deconvolution and bilateral filtering. Experimental results show that the proposed algorithm compares favorably to a computationally-intensive iterative deconvolution algorithm that does not employ any side information. In particular, it can recover small drawings and text that are severely blurred in a poorly focused lecture video.

Categories and Subject Descriptors

I.4.4 [**Image Processing and Computer Vision**]: Restoration

General Terms

Design

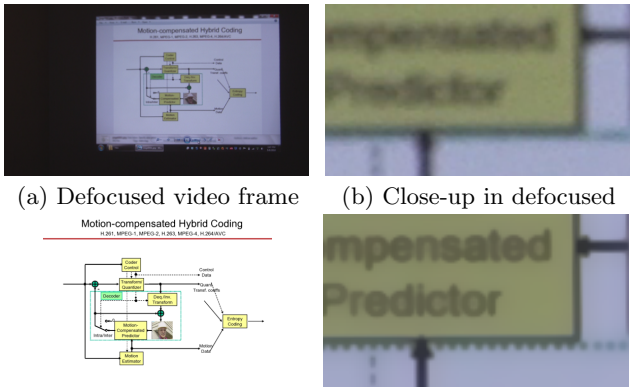
Keywords

Video restoration, slide recognition, local features, deconvolution, bilateral filter

1. INTRODUCTION

Recent years have seen a growing interest in lecture video capturing. Many universities and workplaces are recording classes or training materials on a variety of subjects [1, 8]. Low-cost lecture capturing with automated systems or by non-professionals is essential for large-scale recording, but the captured videos may suffer from different types of degradation, such as inadequate illumination or camera/projector defocusing.

In this work, we investigate automatic restoration techniques for an out-of-focus lecture recording, with an emphasis on the presentation slide area of the video. The slide area of a lecture video is important, as it is often of interest for people watching the video. Restoring the slide area, however, is particularly challenging. First, the slide area often contains fine detail such as text, equations or drawings. Moreover, the slide area usually requires very high quality restoration, or the details in the slide would not be recognizable. Lecture videos also contain a large amount of data, and many computationally-intensive iterative algorithms developed primarily for restoring photos could be impractical for defocused videos.



(a) Defocused video frame (b) Close-up in defocused

(c) Electronic slide (d) Our restoration result

Figure 1: Lecture video restoration with automatic slide matching. (a) Out-of-focus video frame resulting from improper projector focus settings. (b) Close-up region in the defocused frame. Fine details are blurred. (c) Matching electronic slide identified automatically from a slide deck. (d) Our restoration results with assistance from the matching slide. Text is recognizable, as are arrows and dashed lines.

The main novelty of our work is that we consider using the electronic slides that may be available along with the defocused video to aid restoring the slide area. Increasingly, online lectures have images of slides displaying alongside the relevant sections of the videos. Automatic slide matching is becoming an integral part in producing online lectures [1,4]. In this work, we leverage automatic slide matching to restore defocused lectures (Fig. 1). In our proposed algorithm, we first perform automatic slide matching to identify the slide that is displayed in the defocused video frame, based on local feature descriptors and a geometric consistency check [12]. To address descriptor distortion due to out-of-focus, we propose to use multiple defocused slide decks. We then use the best matching defocused slide to estimate the parameters for deconvolution. We also use the structural information in the

MM'10, October 25–29, 2010, Firenze, Italy.

matching slide to enhance the deconvolution output. Our results suggest the proposed algorithm can recover many fine details and faint edges from the blurred videos. We remark that although slide images are available, it may not be possible to display them side-by-side with the lecture videos if the users view them from small terminals (e.g., smartphones). Therefore, it is important to restore the videos properly.

Previous work has proposed iterative algorithms to restore blurred images, e.g., [2, 5]. Other techniques analyze the edge signals to estimate the blur kernels, e.g., [3]. Image deblurring with the help of another perfectly-aligned image has been investigated in [14]. However, there is no previous work on video restoration with assistance from slides. Winslow et al. propose to compose images of the slides into the video [13]. This enhances readability, but results in a synthetic-looking video. In contrast, our work *restores* the degraded video using clean slide images as side information, so that the result looks natural.

2. PROBLEM FORMULATION

We assume that the clean video frame $a(x, y)$ is convolved with a spatially invariant *defocus kernel* $h(x, y)$, and the convolution result is contaminated by additive noise $n(x, y)$, resulting in the defocused frame $f(x, y)$:

$$f(x, y) = a(x, y) * h(x, y) + n(x, y). \quad (1)$$

Here $*$ is the convolution operator. For defocus distortion, $h(x, y)$ can be well modeled by a symmetric circular 2-D Gaussian with spread σ_h [3]:

$$h(x, y; \sigma_h) = \frac{1}{2\pi\sigma_h^2} e^{-(x^2+y^2)/2\sigma_h^2}. \quad (2)$$

We assume $n(x, y)$ is white zero-mean noise with variance σ_n^2, and is statistically independent of $a(x, y)$. Our goal is to recover $a(x, y)$. In practical situations, σ_h and σ_n are usually unknown. Recovering $a(x, y)$ from $f(x, y)$ blindly with unknown σ_h and σ_n is highly unconstrained and is very challenging [9]. Our work seeks to use the slide deck as side information to help restore $a(x, y)$.

3. RESTORATION ALGORITHM

The proposed algorithm consists of the following steps:

- **Automatic slide matching.** The defocused frame is matched against a slide deck to identify the slide which is displayed in the video frame. The matching is performed automatically by comparing the local features of the video frame with those of the slides. Note that defocus makes it difficult to find the matching slide, as it would affect interest point detection and distort the local feature descriptors. We will discuss solution to address this issue.
- **Wiener deconvolution.** The matching slide is then used to estimate the defocus kernel and the noise variance. With these estimates, the video frame can be restored using low-complexity non-blind deconvolution techniques. In particular, we use Wiener deconvolution in this work [9]. We will discuss how the parameters of Wiener deconvolution can be estimated with the help of the matching slide.
- **Edge preserving filtering.** Deconvolution output usually exhibits some ringing artifacts around sharp edges. We apply a bilateral filter [11] to the deconvolution output to remove these artifacts. We propose to compute some of the filter coefficients from the matching slide instead of from the video frame. As the slide has distinct structural information, our proposed slide-assisted bilateral filter tends to perform better than the standard bilateral filter.

3.1 Slide Matching with Multiple Defocused Slide Decks

We use pairwise image comparison to match the frame against the slides [12]. First, we extract local descriptors from the video frame and slides. We detect local extrema in the difference-of-Gaussian (DOG) filtered images and compute 128-dimensional SIFT descriptors to summarize the local gradients around the interest points [6]. Then, we compare the extracted descriptors from the video frame against those in the slides. We establish correspondences between descriptors that are nearest neighbors in the descriptor space. From these correspondences, we use RANSAC to estimate the geometric transformation between the frame and the slides [12]. We return as the matching result the slide which has the maximum number of correspondences consistent with estimated transformation.

While this pairwise image comparison can achieve very high matching accuracy when the video is well focused, defocus may severely undermine its performance. First, defocus reduces interest point repeatability [7]. Second, even if an interest point can be detected in a defocused video frame, the blur would alter the local gradients around the interest point and distort the descriptor. The distorted descriptor may no longer match its counter-part in the slide deck, resulting in a loss in matching descriptor pairs. We observed that, in some cases, it becomes difficult to find any matching descriptor pairs between the defocused video frame and the slides. As a result, the matching slide cannot be identified.

To overcome these problems, we perform slide matching against multiple defocused slide decks. In particular, we convolve the original slide deck with defocus kernels $h(x, y; \sigma_k)$, i.e., 2-D Gaussian kernels with *defocus scale* σ_k, $k = 1 \ldots K$. This generates a stack of K slide decks with different defocus scales, i.e., different amount of defocusing. We perform pairwise image comparison between the out-of-focus video frame and this stack of defocused slide decks, and declare as the matching slide the one with the maximum number of matching descriptor pairs. Fig. 2 shows an example where the matching slide is found at a certain defocus scale σ_k^*. In the experiment, we use eight different defocus scales $\sigma_k = \frac{1}{2}, 1, \frac{3}{2}, ..., 4$.

While matching with multiple slide decks increases computational complexity, the solution is still practical. First, the size of a slide deck is usually small. A typical one-hour lecture video usually has only tens of slides. Second, the amount of defocusing tends to exhibit only small variations throughout the video. Therefore, matching with multiple defocus scales is required only after a period of time, and can be speeded up by searching only those defocus scales in the proximity of the previous matching scale.

3.2 Wiener Deconvolution with Slide-assisted Parameter Estimation

Using multiple defocused slide decks allows the matching slide to be found. Geometric transformation between the video frame and slide can also be estimated from the matching descriptor pairs using RANSAC. To restore the out-of-focus frame, we use Wiener deconvolution. The spectrum of the sharpened image, $i(x, y)$, is given by [9]:

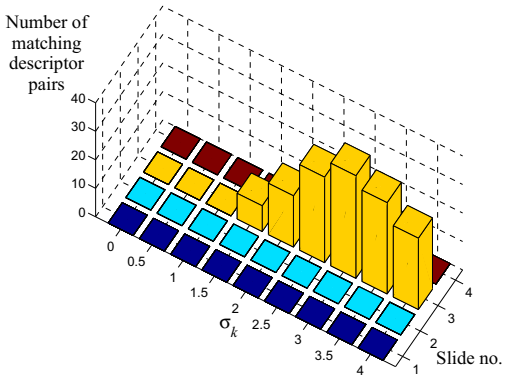


Figure 2: Matching results with multiple defocused slide decks. In this experiment, we blur a clean video frame with a 2-D symmetric Gaussian of $\sigma = 3$. The blurred video frame is matched against the stack of defocused slide decks. As shown in the figure, the number of matching descriptor pairs consistent with the transformation (vertical axis) has the maximum with slide no. 3, which is the correct matching slide, and at the defocus scale $\sigma_k^* = 3$, agreeing with the blur kernel. Note that in this case the system cannot find any matching pair between the Gaussian-blurred video frame and the original slide deck (slides with $\sigma_k = 0$), and by defocusing the slides we recover some matching pairs.

$$I(u,v) = \frac{\Phi_{aa}(u,v)H^*(u,v)F(u,v)}{\Phi_{aa}(u,v)|H(u,v)|^2 + \Phi_{nn}(u,v)}, \quad (3)$$

where $H(.)$, $F(.)$ are the Fourier transforms of $h(.)$, $f(.)$ respectively, $H^*(.)$ is the complex conjugate of $H(.)$, and Φ_{aa}, Φ_{nn} are the power spectral density of $a(.)$ and $n(.)$ respectively. With the defocus model in Section 2, and by approximating Φ_{aa} with that of the electronic slide, we would need to estimate σ_h and σ_n to perform the deconvolution.

We estimate σ_h by the defocus scale σ_k^*, i.e., the scale of the defocus kernel that leads to the maximum number of matching pairs. We select this estimate because, judging from the number of matching pairs, convolving the clean matching slide with a Gaussian of scale σ_k^* results in descriptors very similar to the distorted ones in the video frame. Simulation results suggest that this estimate is usually within ± 1 of the true defocus scale. To estimate the variance σ_n of white noise $n(.)$, we perform the following steps. We project the matching slide onto the defocused frame using the estimated geometric transformation. We partition the projected slide, $l(x,y)$, into non-overlapping regions of size $m \times m$. We detect regions of zero variance from $l(x,y)$, i.e., totally flat regions. As the slide is free from any noise, zero-variance regions can usually be found in the slide's background. We compute the variance of the corresponding co-located regions in the defocused frame $f(.)$. Since these regions should have no texture information, we attribute the pixel variation in these regions to the noise $n(.)$. Therefore, we estimate σ_n^2 by averaging the variance in these regions in $f(.)$.

3.3 Slide-assisted Bilateral Filtering

Output of deconvolution may exhibit some ringing artifacts around sharp edges of the video frame [9, 14]. High frequency components of these sharp edges were severely attenuated in the defocus process, and are very difficult to recover as they may be embedded in the noise. Loss of these high frequency components results in ringing artifacts around sharp edges. Note that these artifacts tend to be more pronounced in our application, as slides usually have a lot of sharp edges and boundaries in text and drawings. Ringing artifacts are also more visible with the plain backgrounds of many slides.

We use bilateral filter [11] to remove these artifacts in the deconvolution output. We propose to use the matching slide to help compute the weights in bilateral filtering. In particular, bilateral filtering of the deconvolution output $i(.)$ is given by:

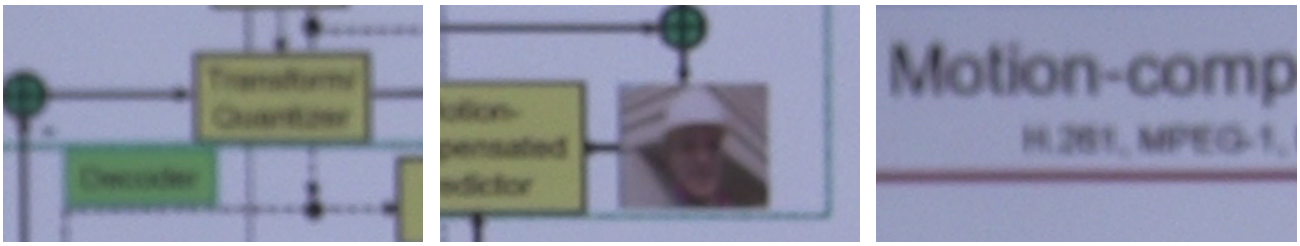
$$BF[i(.)]_{\mathbf{p}} = \frac{1}{w_{\mathbf{p}}} \sum_{\mathbf{q}} G(\|\mathbf{p}-\mathbf{q}\|;\sigma_s) G(|l(\mathbf{p}) - l(\mathbf{q})|;\sigma_r) i(\mathbf{q}). \quad (4)$$

That is, the filter result at pixel location $\mathbf{p}$ is the weighted average of the frame pixel value $i(\mathbf{q})$ at location $\mathbf{q}$, for all $\mathbf{q}$ in the filter window of $\mathbf{p}$. The *spatial weight* $G(.;\sigma_s)$ is the 1-D Gaussian function on the distance between $\mathbf{p}$ and $\mathbf{q}$. The *range weight* $G(.;\sigma_r)$ is the 1-D Gaussian function on the difference between pixel values at $\mathbf{p}$ and $\mathbf{q}$ of the *projected matching slide* $l(.)$. Note that this is different from the standard bilateral filter, where the range weight would be calculated from the frame $i(.)$ in this case. $w_{\mathbf{p}}$ is the normalization factor. We use the projected slide to compute the range weight as the slide has more distinct structural information.

In addition to its edge preserving quality, the proposed slide-assisted bilateral filter also has a computational complexity advantage over the standard bilateral filter. Since the range weights and the normalization factors are calculated based on the matching slide, their values are the same for all the frames matching the same slide and with the same geometric transformation. Therefore, for these frames, the range weights and the normalization factors would need to be computed only once from the slide and can be re-used for all the frames. This compares favorably to the standard bilateral filter, which may compute the range weights and the normalization factors for individual frames.

4. EXPERIMENTS

We recorded presentation videos with blur caused by improper projector focus settings. The videos are of 1080p format (1920×1080 pixels per frame). This video format is used in several automated lecture capturing projects [1, 8]. Fig. 1(a) shows an example frame. The defocused video frames were matched against a deck of 16 slides. In this experiment, all the matching slides could be correctly identified. The matching slides were then used to restore the video frames following the proposed algorithm. Block size in noise variance estimation is 16×16. Bilateral filter's window size is 9×9, with parameters $\sigma_s = 3$, $\sigma_r = 0.1$. Fig. 3 depicts the close-up regions of several frames. Also shown in the figure are the restoration results with Matlab's `deconvblind` program, which implements the blind deconvolution algorithm in [2]. Note that `deconvblind` is iterative and is computationally-intensive. We initialized `deconvblind` with a Gaussian blur kernel with window size estimated by examining the blur artifacts in the frames. As `deconvblind` output exhibited some ringing artifacts, we also applied the standard bilateral filter to the blind deconvolution output. As shown in Fig. 3, while this blind deconvolution algorithm can repair some strong edges and main text, it is inadequate to address the blurred fine details. These fine details were



(a) Defocused video frames

(b) Restoration by blind deconvolution

(c) Slide-assisted restoration

Figure 3: Comparison results (We encourage close-up views in the electronic version. Additional results available at [10]). (a) Close-up regions in the out-of-focus video frames. Fine details are severely blurred. (b) Restoration results using the blind deconvolution program implemented in Matlab, followed by the standard bilateral filter. While there are improvements in the strong edges and main text, details are not recognizable. (c) Our restoration results. Fine details are much more visible. Note that video frames may have rather low contrast compared to photos, and we did not perform any enhancement to the results.

severely attenuated in the defocus process, and are very difficult to distinguish from the noise. Our proposed algorithm makes use of the matching slide to achieve much better recovery of these details in the slide area, as demonstrated by the results.

5. CONCLUSIONS

We have discussed a lecture video restoration algorithm that leverages automatic slide matching. We proposed to use local features and multiple defocused slide decks to enable automatic recognition of slides under out-of-focus distortion. We utilized the matching slide to estimate the parameters for a computationally-efficient non-blind deconvolution. We also used the structural information of the slide to assist bilateral filtering of the deconvolution output. Experimental results suggest that our algorithm can recover many fine details that are severely blurred in a defocused lecture video.

6. REFERENCES

[1] ClassX: Stanford University Online Lecture Project. `http://classx.stanford.edu/` .

[2] D. S. C. Biggs and M. Andrews. Acceleration of iterative image restoration algorithms. *Applied Optics*, 36(8):1766–1775, 1997.

[3] J. H. Elder and S. W. Zucker. Local scale control for edge detection and blur estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(7):699–716, 1998.

[4] Q. Fan, K. Barnard, A. Amir, A. Efrat, and M. Lin. Matching slides to presentation videos using SIFT and scene background matching. In *MIR '06: Proc. ACM International Workshop on Multimedia Information Retrieval*, pages 239–248, New York, NY, USA, 2006.

[5] R. Lagendijk, J. Biemond, and D. Boekee. Identification and restoration of noisy blurred images using the expectation-maximization algorithm. *IEEE Trans. Acoustics, Speech and Signal Processing*, 38(7):1180 –1191, July 1990.

[6] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, pages 91–110, 2004.

[7] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *Int. J. Comput. Vision*, 65(1):43–72, 2005.

[8] T. Nagai. Automated lecture recording system with AVCHD camcorder and microserver. In *SIGUCCS '09: Proc. ACM SIGUCCS fall conference on User services conference*, pages 47–54, New York, NY, USA, 2009.

[9] W. K. Pratt. *Digital Image Processing*. John Wiley & Sons, 2nd edition, Apr. 1991.

[10] Supplementary material. `http://msw3.stanford.edu/~ncheung/mm2010.html` .

[11] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, pages 839–846, Jan. 1998.

[12] S. Tsai, D. Chen, J. Singh, and B. Girod. Rate-efficient, real-time CD cover recognition on a camera-phone. In *Proc. ACM International Conference on Multimedia*, 2008.

[13] A. Winslow, Q. Tung, Q. Fan, J. Torkkola, R. Swaminathan, K. Barnard, A. Amir, A. Efrat, and C. Gniady. Studying on the move - enriched presentation video for mobile devices. In *Proc. IEEE Workshop on Mobile Video Delivery (MoViD)*, 2009.

[14] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum. Image deblurring with blurred/noisy image pairs. In *SIGGRAPH '07: Proc. ACM SIGGRAPH*, New York, NY, USA, 2007.