Improving Super-Resolution Enhancement of Video by using Optical Flow

Chris Crutchfield
MIT
ccrutch@mit.edu

Abstract

In the literature there has been much research into two methods of attacking the super-resolution problem: using optical flow-based techniques to align low-resolution images as samples of a target high-resolution image, and using learning-based techniques to estimate perceptually-plausible high frequency components of a low-resolution image. Both of these approaches have been naturally extended to apply to image sequences from video, yet heretofore there have been no investigations into combining these methods to obviate problems associated with each method individually. We show how to merge these two disparate approaches to attack two problems associated with super-resolution for video: removing temporal artifacts ("flicker") and improving image quality.

1. Introduction

Super-resolution enhancement of images has been a well-studied topic in the literature, with a wide variety of solutions. All of these methods attempt to solve the same problem: to increase the resolution (the number of pixels) of a given image while also estimating the missing high frequency content of the resized image.

Of these methods, there have been three major approaches for increasing image resolution. The first method involves interpolating a single image to a higher resolution, and then boosting its high frequencies by applying a deconvolution filter [8]. The second method uses several low resolution, aligned images as samples of a high resolution image, which it them attempts to estimate [3]. The third method uses learning-based techniques to infer perceptually-plausible high frequencies for a low resolution image [5].

However, despite the fact that super-resolution for images has been a well-studied topic, there have been comparatively few investigations into applying these techniques to video (in effect, generalizing the problem to three dimensions — two in space and one temporal — where the goal is to increase the spatial resolution by making use of the information provided by the additional temporal dimension). The first investigation into this domain was an extension of Chiang and Boult [3], involving the use of optical flow to align successive frames [1]. The second approach was an extension of the methods of Freeman et al. [5] to apply their VISTA algorithm to each frame individually [2].

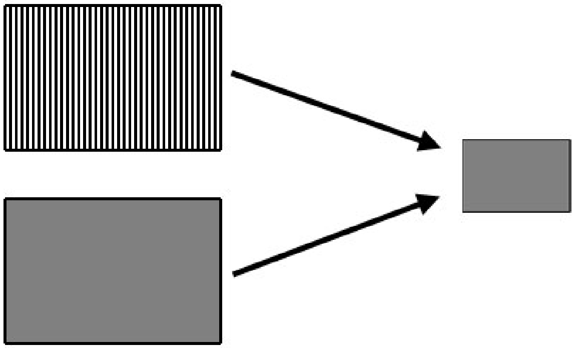

Figure 1. The images on the right are very different in terms of frequency content, yet they both map to the same image when blurred and downsampled.

1.1. Super-Resolution

The task of super-resolution seems nearly impossible at first — to extract information from an image that is simply not present. Although all the methods listed in this section claim to achieve this task, there is no true algorithm for solving this problem exactly. Since many very different high-resolution images may all map to the same low-resolution image, we have no hope for recovering the initial image for all cases (see Figure 1). However we may *estimate* or *infer* what the high-resolution image most likely looks like using several techniques. In this section I will review the three main super-resolution techniques applied in the literature.

Several papers (Schultz et al. [8], Chiang et al. [4]) have addressed the task of boosting high frequencies present in an image by deblurring using a deconvolution filter (typi-

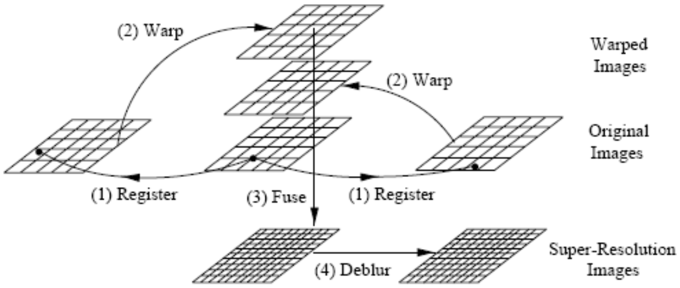

Figure 2. Super-resolution using image sequences.

cally Wiener deconvolution). There are a host of problems associated with this task; however, they are mostly concerned with estimating the blur kernel that has been applied to the image, in order to deconvolve and remove the blur (and therefore boost up the missing high-frequency components). This is certainly no trivial task, since estimating the blur of an image is an inexact science. The conclusion of Chiang et al. [4] was that more robust methods are needed in practice in order for deconvolution alone to be feasible.

Another approach discussed in Chiang et al. [3] is to reconstruct a high-resolution image from a sequence of low-resolution images that are pre-aligned (see Figure 2 for a pictorial representation). In this “registration” step, each pixel in the high-resolution is assigned a point (which may be a subpixel location) in each low-resolution image. The assumption is that the registration is known *a priori*, so that we then only concern ourselves with combining these images to produce the high-resolution output. In order to combine the images, each low-resolution image is warped into the coordinate frame of the high-resolution image (using the registration information). There are several methods for performing this task, which involves interpolating the values of the subpixel locations in each image using the registration (typically one may use nearest-neighbor, bilinear, or bicubic interpolation). The result of these computations is a stack of high-resolution images, which may then be fused together by taking a robust mean to produce a composite high-resolution image. This image may then be deblurred by applying the Wiener deconvolution filter mentioned above.

In the seminal paper by Freeman, Pasztor and Carmichael [5], a *learning-based* algorithm, VISTA, for solving the super-resolution problem was developed. In this paper, they demonstrate how to extract perceptually-plausible high-frequency components from a low-resolution image. They do so by constructing a training set out of a sequence of high-resolution images. By pairing image patches from the high-resolution image to their low-resolution counterparts (by blurring and downsampling to remove the high-frequency components), one can infer from a given low-resolution image patch what the most likely high-resolution patch would be.

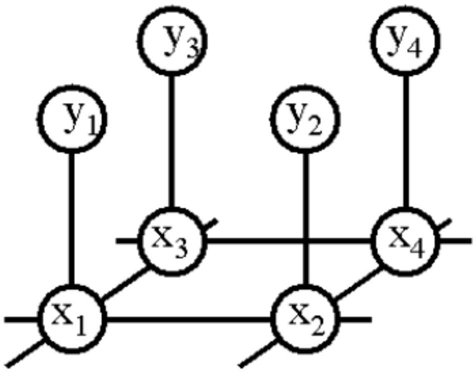

Figure 3. Markov Random Field for images. The observations y_i are the low-resolution patches in the input image. The nodes we wish to estimate x_i represent the high-resolution patches in the output. Each x_i is connected to its associated y_i (enforcing that the medium-frequency components of x_i are “close” to the medium-frequency components of y_i). In addition, each x_i is connected to its neighbor with some compatibility criterion, ensuring that neighboring high-resolution patches “stitch” together well.

In particular, in order to solve this problem for the entire image, they construct a Markov Random Field (see Figure 3) for the image. By applying Bayesian Belief Propagation to this network, they then can reconstruct the maximum likelihood high-resolution output, conditioned on the low-resolution input.

1.2. Super-Resolution for Video

The motivations for applying super-resolution for video are quite apparent. Videos require such a large amount of storage space that they are often of much smaller resolution than the devices used to display them. In particular, in the case of videos streamed over the internet, the space requirements are even more stringent (indeed, they then become bandwidth requirements). This begs the following question: what if we could design an algorithm that allows videos to be streamed to the user at a relatively low resolution, but with some processing we could boost the video to the higher resolution of their display? As mentioned above, there have been several approaches to answering this question.

The work of Baker and Kanade [1] extended the results of Chiang et al. [3] to apply to video. By computing the *optical flow* of the video sequence using Lucas-Kanade [7] or a similar approach, we can then warp the video frames surrounding a particular frame so that they all are aligned in the same coordinate frame. By repeating this process for each frame, we obtain a collection of low-resolution, aligned images for each frame of the video. We can then apply the techniques of Chiang et al. to extract a high-resolution es-

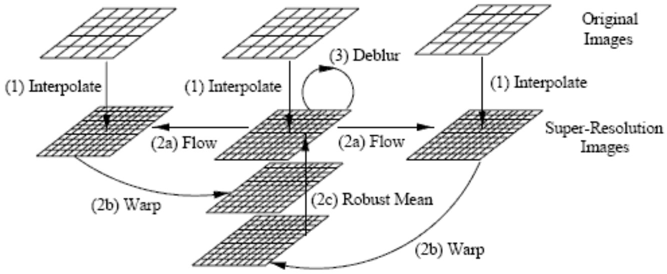

Figure 4. Super-resolution optical flow algorithm: (1) Bilinearly interpolate each frame individually, (2a) Compute optical flow to neighboring frames, (2b) Warp each frame to its neighbors, (2c) Compute a robust mean of the collection of frames, (3) Deblur the result using a Wiener deconvolution filter.

timate of this frame from the collection of aligned images (see Figure 4 for more details). Although this technique works well for carefully chosen examples, when applied to real-world data, optical flow algorithms fail to provide the precision necessary to extract a quality high-resolution output.

The work of Bishop, Blake and Marthi [2] extended the results of Freeman et al. [5] to video. They noticed that if you apply the VISTA algorithm individually to each frame of the video, the result is visually unappealing due to many temporal artifacts. They noted that these “distracting scintillations” were caused by the lack of temporal consistency from applying the VISTA algorithm independently to each frame. Since each frame is processed independently, a high-resolution image patch applied in frame i might not be the same as the high-resolution image patch applied in frame $i + 1$ (even though both patches might be identical or very close in low-resolution). Their solution involved the addition of a regularization parameter β to the cost function for selecting patches, in order to favor re-selecting the same patch for successive frames.

1.3. Our Approach

In this paper we propose new methods for combining the work of Baker et al. [1] and Bishop et al. [2] to use both optical flow-based techniques as well as learning-based techniques for the problem of super-resolution for video. We divide the paper into two sections with two different goals. First, we wish to develop an algorithm that uses optical-flow techniques to reduce the temporal flickering associated with applying the VISTA algorithm to each frame individually. Second, we wish to come up with a method for using optical-flow techniques for video sequences which are exactly low-resolution samples of some high-resolution image, shifted around randomly by some subpixel amounts, in order to extract high-resolution outputs that are of better quality than simply applying the VISTA algorithm to each frame individually.

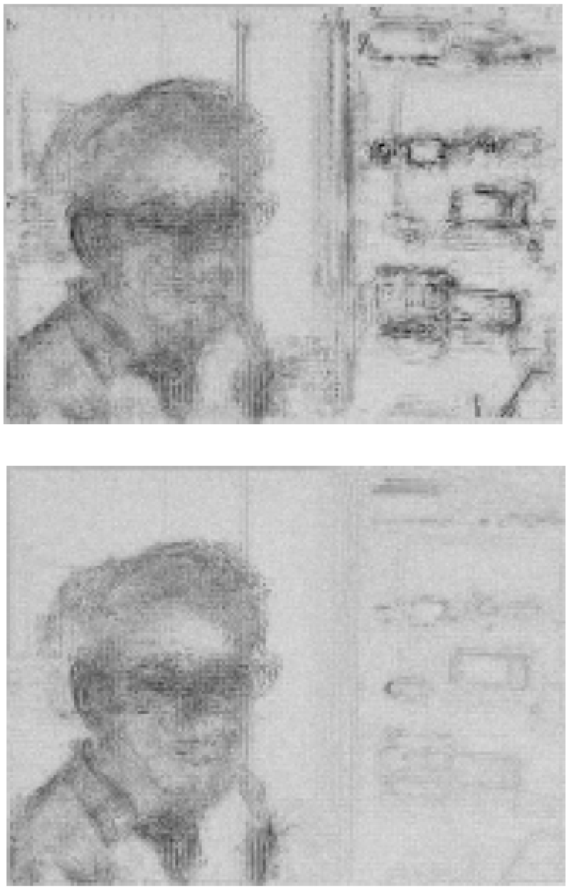

Figure 5. (Above) Average absolute error between the super-resolution video and the ground truth video averaged across all frames, before applying the regularlization parameter. (Below) After.

1.4. Viewing the Results

Since the subject of this paper deals with the perceptual quality of video to the human eye, our results do not lend well to being displayed in a static format (e.g. as figures in this paper). Therefore I have posted screen-captures of each frame of the video, in a side-by-side manner, on my personal webspace for viewing. In addition I have posted MPEG files for each video, but due to the compression associated with this format it does not display very well. These files will be made available at http://people.csail.mit.edu/cyc/6.869/project/superres.html.

2. Reducing Temporal Flicker

The goal of this algorithm is to eliminate the high-frequency flicker present in the approach of applying VISTA individually to each frame, while still retaining the desired perceptually-plausible high frequencies. By applying the optical flow techniques of Baker et al. [1] we hope to gather a collection of VISTA-enhanced samples for each frame. Since the desired high frequencies will be present in each frame of the video, whereas the undesired high-frequencies (the noise) will vary from frame to frame, by taking a robust mean of these collections for each frame the hope is that the desired high frequencies will constructively interfere, whereas the undesired high frequencies will destructively interfere.

2.1. A Super-Resolution Optical Flow Algorithm

This algorithm was adapted from the Super-Resolution Optical Flow algorithm of Baker et al. [1].

1. Apply VISTA [5] individually to each frame of the video (which uses bicubic interpolation to double the resolution of the image, and then adds in perceptually-plausible high frequencies).

2. For each frame of the video, iterate the following steps until convergence (in practice, 5–10 iterations are usually enough). Note that for the first, second, second-to-last, and last frames you may simply leave them be.

   (a) For frame X_i, compute the optical flow from X_i to X_{i-1} and X_{i-2}, as well as to X_{i+1} and X_{i+2}. For the sake of efficiency, we use the Lucas-Kanade optical flow algorithm [7].

   (b) Warp the frames $X_{i-2}, X_{i-1}, X_{i+1}, X_{i+2}$ into the coordinate frame of X_i to create a collection of aligned images.

   (c) Let X_i' be a robust mean of $X_{i-2}, X_{i-1}, X_i, X_{i+1}, X_{i+2}$ (in practice this is usually just the arithmetic mean or the median). Replace X_i with X_i' for the next iteration.

3. (Optional) Deblur each frame using a Wiener deconvolution filter.

Note that in the last step, we may decide to deblur each frame of the resulting high-resolution video to remove any blur that may have been caused by imprecise optical flow. Since the techniques used for estimating optical flow and for warping images according to this flow are not perfect, they may cause the resulting image frame to become blurry as a result of this imprecision (if the optical flow or warping is off by even a subpixel amount, this may cause some perceptible blurriness in the output). Therefore in order to remove this artifact it may be necessary to deblur the image (in practice we use a Wiener deconvolution filter with a Gaussian blur kernel).

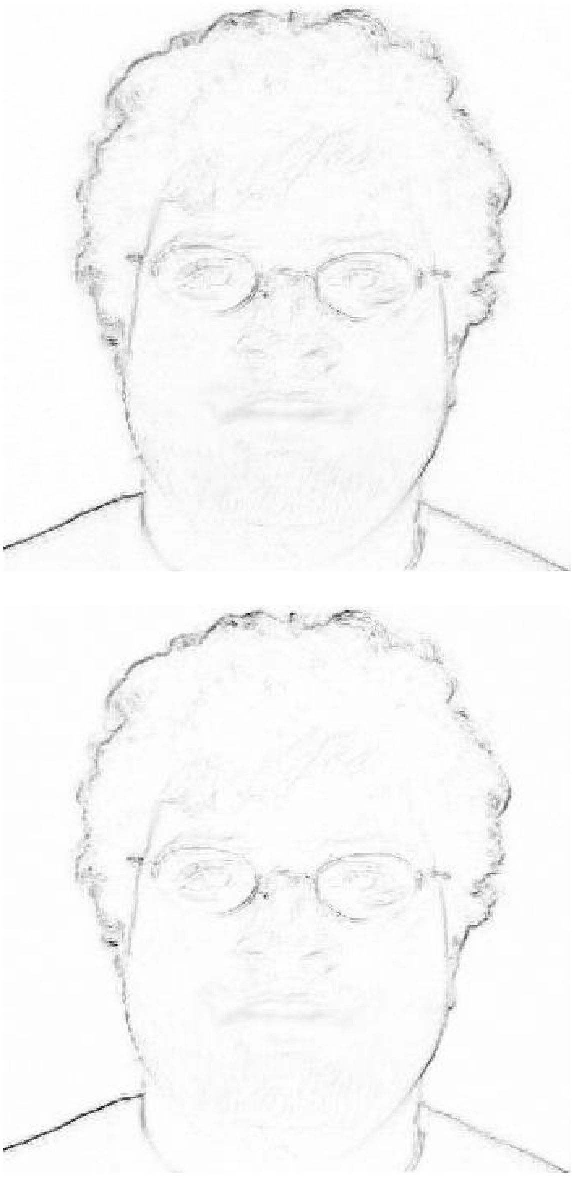

Figure 6. (Above) Average absolute error between the VISTA super-resolution video and the ground truth video averaged across all frames. (Below) Average absolute error between the output of the algorithm in Section 2.1 and the ground truth video averaged across all frames. Note that much of the error density around the eyeglasses (where the flickering is most apparent in the video) has been reduced.

2.2. Results and Analysis

It is difficult to objectively quantify the amount of “flicker” present in a video, since it is largely a measure based on human perception. Thus for the purposes of comparing our algorithm to the output of VISTA applied individually to each frame we need to come up with some quantitative measure of flicker content. In order to do this we create our input video by blurring and downsampling our “ground truth” video. By doing so, we can compare our output to the ground truth to determine our error. For the analysis of this algorithm we used an image sequence created from two pictures: one of a man with a neutral expression on his face, and the other of the same man with a smile on his face. By computing the optical flow of these two images and temporally interpolating the intermediate frames, we created a test dataset of 32 frames of resolution 350×350.

In Figure 6 we compare the output of applying VISTA individually to each frame to the output of our algorithm. If one watches the associated video of the output of VISTA[1], one notices that most of the flickering occurs around the rims of the man’s eyeglasses. This is largely due to the fact that these are the most prominent edges in the image sequence, hence the areas where the VISTA algorithm tries to add the most high frequencies to (in order to preserve the shape of the edge in the high-resolution output). In the output of our algorithm, this is somewhat reduced, as one can see that the dark areas around the man’s eyeglasses are noticeably lighter.

However, Figure 6 only measures the average error between the ground truth and the outputs of the two algorithms, which is not what we desire to measure. We wish to have some quantitative measurement of the “flicker content” of both outputs. Therefore in Figure 7 we compare the high-frequency content of each video — the output of VISTA and the output of our algorithm. The measure of high-frequency content was produced by applying a fifth-order highpass Butterworth filter to each frame of the video, and then averaging over the absolute value of the result. This data was plotted for each frame of the sequence. The reduction in flicker content is modest, yet not ideal (as one can note by the scale of the y-axis).

Despite the reduction in flicker content, the output of our algorithm still maintains the high-frequency detail of the output of VISTA[2].

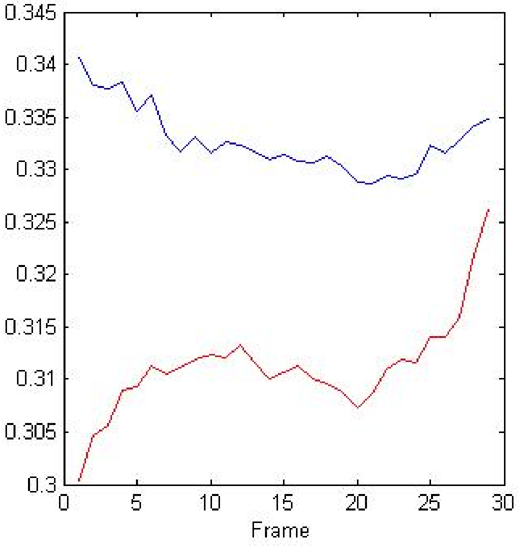

Figure 7. Plot of the measure of flicker averaged over the pixels of the output of VISTA (shown in blue) and the output of the algorithm in Section 2.1 (in red).

2.2.1 Misalignment Due to Poor Optical Flow

The above algorithm relies heavily on having precise optical flow information. In practice this means that the video framerate must be extremely high so that we can reliably compute the Lucas-Kanade optical flow between frames. However, when we apply our technique to a different dataset that does not have such a high framerate, we find that our algorithm fails. As shown in Figure 8, we have an image sequence of a woman walking on a paved road. Since the images were only sampled at a rate of about 15 frames per step, the optical flow algorithm used in our algorithm has a hard time tracking the figure. As a result, it cannot reliably warp together succesive frames into a common coordinate frame, and the result of averaging this mess is a very blurry figure. This highlights the importance of having reliable, precise optical flow for *any* optical flow-based approach to solving the super-resolution problem. As mentioned in Zhao and Sawhney [9], nearly all of the prior optical flow-based approaches relied on near perfect alignment of images, and when tested on actual real-world examples they break down due to misalignment. They conclude that “errors resulting from traditional flow algorithms may render super-resolution[3] infeasible”.

[1] Available at http://people.csail.mit.edu/cyc/6.869/project/flicker_vista.mpg

[2] Due to space constraints, the output cannot be displayed in this paper. However, they may be viewed at http://people.csail.mit.edu/cyc/6.869/project/superres.html

[3] To be clear, they are referring only to optical flow-based methods.

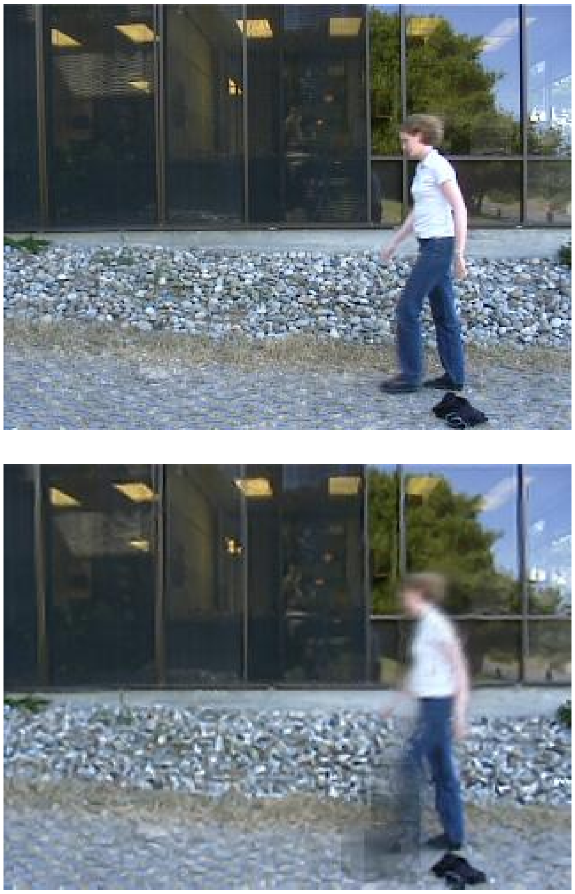

Figure 8. (Above) Correct ground truth image. (Below) Output of the algorithm in Section 2.1. Images taken from the dataset of the PhD thesis of Hedvig Kjellström [6]

3. Using Video Frames as Samples

In this section we consider a modified variant of the algorithm of Section 2.1. The case we wish to consider is when our input video sequence is a randomly-ordered set of shifted downsamplings of a ground truth high-resolution image. This may occur for example, if we have a low-resolution video of a static subject, with random, nearly imperceptible pertubations of the camera (and therefore, the camera reads in a randomly shifted downsampling of the subject for each frame). The rationale behind this is to come up with a contrived example where a set of frames in the video yields a lot of information about the underlying ground truth image (in fact, if one processed all of the frames, one could entirely reconstruct the ground truth image in this model!) The hope is that by applying optical flow techniques to this simpler model, we can come up with higher-quality outputs than just applying VISTA individually to each frame.

In this toy model, we take our ground truth image (see Figure 9) and construct a 16-frame sequence by selecting a random offset and then downsampling by 4 (see Figure 10 for the offsets used in our testcases). Since this is downsampled by 4, we first apply the algorithm of Section 2.1, and then apply VISTA to the results. We compare the result to the output of applying VISTA twice to the downsampled input set.

Figure 9. Test image used for this algorithm. In order to create the image sequence of the video, this image was blurred and (twice) downsampled by random offsets.

8	15	2	16
10	7	4	3
11	14	6	12
9	5	13	1

Figure 10. Offsets used in the creation of the 16-frame input sequence. A high-resolution image was downsampled by picking an offset in the order shown above and selecting every fourth pixel.

3.1. Results and Analysis

In Figure 11 we compare the average absolute difference between the output of VISTA and our approach. We notice that most of the difference seems to occur around the edges in the image, which may be a result of our approach cleaning up much of the high-frequency noise that VISTA includes.

In Figure 12 we compare Frame 3 of the output for both algorithms. It is clear that the output of our approach has much fewer high-frequency artifacts than the VISTA approach, while still maintaining the sharp definition of edges.

Although we have no real way of measuring it, it seems like the approach outlined above gives a better output for this toy model. Though it may simply be due to the ability of our algorithm to supress the high-frequency noise that VISTA produces, it might be the case that our algorithm

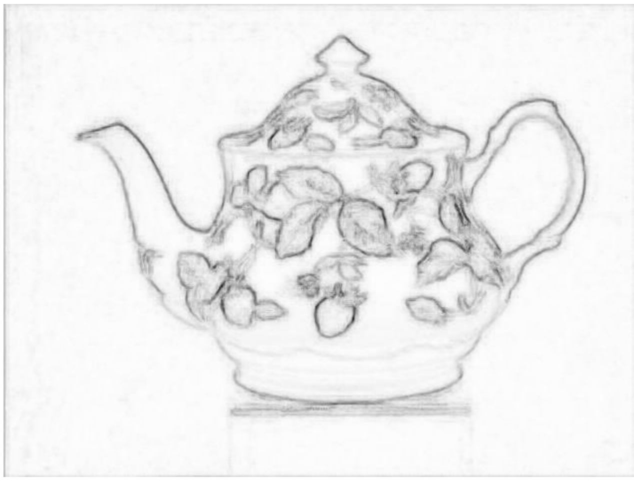

Figure 11. Absolute difference between the output of VISTA and our approach, averaged over all 16 frames.

is extracting some additional information about the ground truth image from neighboring frames in the video. Therefore I feel that this approach definitely may be worth looking into in the future.

4. Conclusion

In this paper we have shown how to use optical flow-based techniques to alleviate some of the problems associated with applying VISTA individually to each frame of the video. Although it relies heavily on the precision of the optical flow algorithm, this seems like a promising approach for producing a better super-resolution algorithm for video. Such techniques may eventually yield better video compression algorithms for websites such as YouTube, for whom bandwidth concerns are paramount.

References

[1] S. Baker and T. Kanade. Super-resolution optical flow, 1999. 1, 2, 3, 4

[2] C. Bishop, A. Blake, and B. Marthi. Super-resolution enhancement of video, 2003. 1, 3

[3] M. Chiang and T. Boult. Efficient image warping and super-resolution. In *Proceedings of the Third IEEE Workshop on Applications of Computer Vision*, pages 56–61, Dec. 1996. 1, 2

[4] M. Chiang and T. Boult. Local blur estimation and superresolution, 1997. 1, 2

[5] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *International Journal of Computer Vision*, 40(1):25–47, 2000. 1, 2, 3, 4

[6] H. Kjellström. *Probabilistic Tracking and Reconstruction of 3D Human Motion in Monocular Video Sequences*. PhD thesis, Dept. of Numerical Analysis and Computer Science, KTH, Sweden. 6

[7] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI81*, pages 674–679, 1981. 2, 4

[8] R. Schultz and R. Stevenson. A bayesian approach to image expansion for improved definition. *Image Processing, IEEE Transactions on*, 3(3):233–242, May 1994. 1

[9] W. Zhao and H. S. Sawhney. Is super-resolution with optical flow feasible? In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I*, pages 599–613, London, UK, 2002. Springer-Verlag. 5

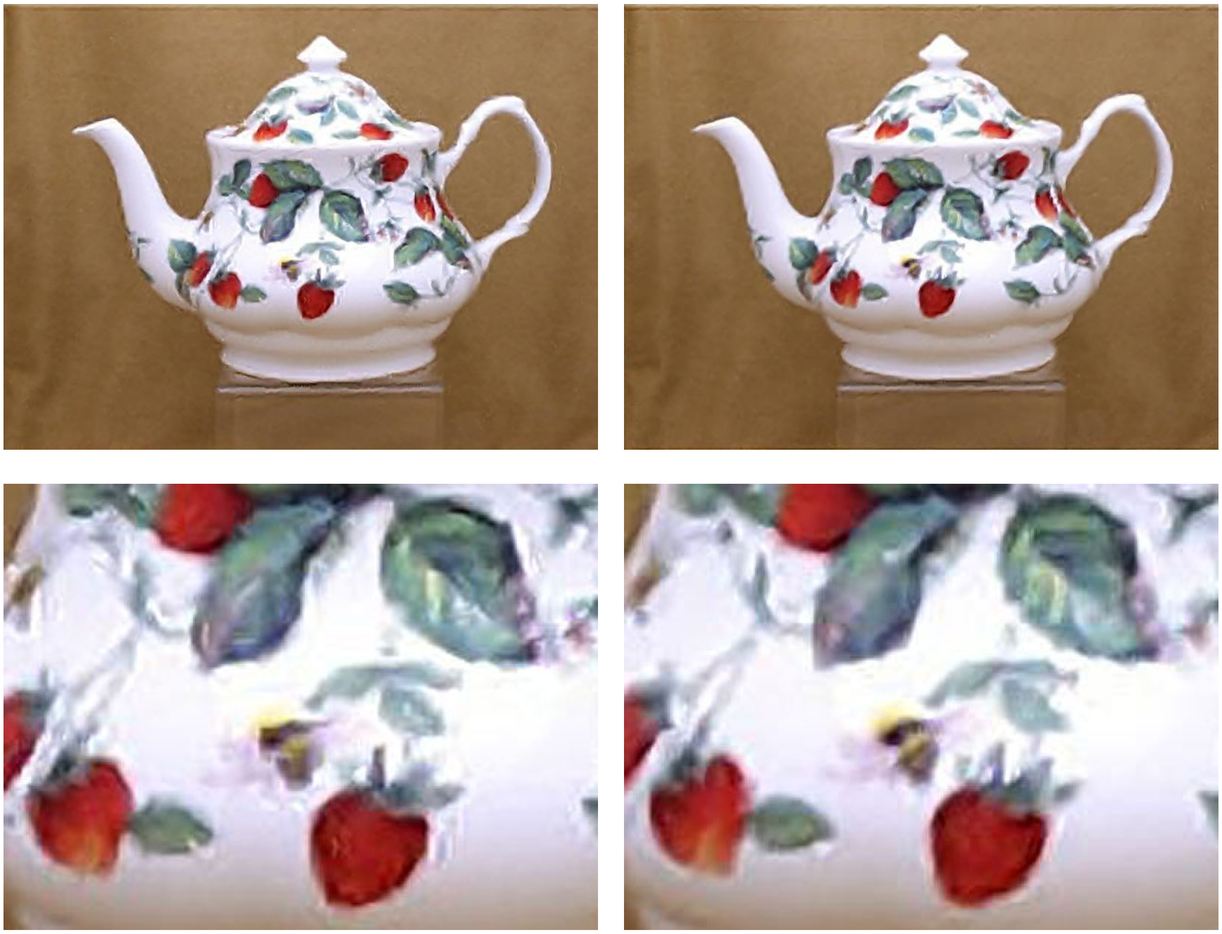

Figure 12. Side-by-side comparison of Frame 3 of the dataset. On the left is the result of applying VISTA twice to a blurred and (twice) downsampled version of Figure 9. On the right is the result of applying our algorithm. Note that in the image, edges are less noisy and features are clearer.