To appear Int. Conference on Pattern Recognition, Cambridge, UK, August 2004.

High Frequency Component Compensation based Super-resolution Algorithm for Face Video Enhancement

Abstract

This paper proposes a video-based super-resolution algorithm by high-frequency component compensation. Normalized logarithm distance (NLD) minimization on local shape feature is proposed for image registration. Lost high-frequency information are estimated using the registered frames. By compensating the high-frequency component, the high-resolution images are recovered. The algorithm has lower computational cost than the alternatives. Experimental evaluation verified the usefulness of the algorithm.

1 Introduction

Recently there has been considerable interests in high resolution video reconstruction. In general, existing video super-resolution algorithms can be classified into three categories: frequency domain algorithms [1]; spatial domain algorithms from image generative model; interpolation methods. Frequency domain algorithms are limited by the underlying global translation motion assumption. Real world videos usually have multiple motions or non-rigid motion. For such cases the performance will deteriorate. Spatial domain approaches are motivated from the generative degrading model of low resolution videos. Super-resolution reconstruction is modeled as an inverse problem of this generative model. This inverse problem is ill-conditioned. Different priors have been assumed to solve it [2, 3, 4]. Yet the performance is limited by the consistency between the prior and the data. For interpolation methods, registered low-resolution images are mapped onto a unique non-uniform high-resolution grid [5]. Interpolation is used to get the high-resolution image residing on the corresponding uniform grid. Aliasing is a problem for such approaches.

The human face is different from other popular subjects in computer vision area due to its essential non-planarity and non-rigidity. In this paper, a novel algorithm is presented for human face video super-resolution reconstruction. High-resolution frames are reconstructed by compensating the lost high-frequency component. Experiments on substantive face videos verified its usefulness.

The paper is organized as follows. Pixel-wise image registration by NLD minimization on local shape feature is outlined in section 2. Section 3 presents the proposed super-resolution algorithm. In section 4, experimental results and comparisons are reported. Section 5 concludes the paper.

2 Image Registration

Image registration is a crucial pre-processing step for video super-resolution reconstruction, especially for face videos. Face videos have complicated motions. Various local motions and global motion are present. Illumination and reflectance will change locally in respond to these local motions, such as changes in self-shadows. Hence the pixel intensity will be insufficient for image registration. As an alternative, we use shape feature. More specifically, in our algorithm we define the shape feature as:

$$\begin{aligned}\mathbf{x}_i \mapsto \mathbf{f}(\mathbf{x}_i) &= (\frac{\lambda_{11}}{\lambda_{12}}\theta_1, \cdots, \frac{\lambda_{R1}}{\lambda_{R2}}\theta_R), \\ \theta_k &= (\mathcal{R}_{i,k}\mathbf{u}_{k1})^{\mathrm{T}}(\mathcal{R}_{i,k}\mathbf{u}_{k2}), \end{aligned} \tag{1}$$

where $\frac{\lambda_{k1}}{\lambda_{k2}}(\mathcal{R}_{i,k}\mathbf{u}_{k1})^{\mathrm{T}}(\mathcal{R}_{i,k}\mathbf{u}_{k2})$ is a local curvature measurement. Pixel $\mathbf{x}_{i,k}$ is the kth pixel from $\mathbf{x}_i$'s $\sqrt{R} \times \sqrt{R}$ neighborhood. $\mathcal{R}_{i,k}$ is $\mathbf{x}_{i,k}$'s 3×3 neighborhood. λ_{k1} and λ_{k2} ($\lambda_{k1} \leq \lambda_{k2}$) are the first two greatest eigenvalues for $\mathcal{R}_{i,k}$; $\mathbf{u}_{k1}$ and $\mathbf{u}_{k2}$ are the corresponding normalized eigenvectors. The local curvature metric comes from the following observation: for an image patch, the eigenvalues and corresponding eigenvectors determine its dominant orientation. For example, the eigenvector corresponding to the biggest eigenvalue represents the dominant orientation. $(\mathcal{R}_{i,k}\mathbf{u}_{k1})^{\mathrm{T}}(\mathcal{R}_{i,k}\mathbf{u}_{k2})$ gives the angle between the first two dominant directions. The bigger the angle is, more isotropic the neighborhood is. $\frac{\lambda_{k1}}{\lambda_{k2}}$ shows how major orientation dominates over the second one. A cascade of these local curvature features fully describes the shape structure of an image patch. Experiments show that this feature is more robust to local motions in face videos. Pixels' correspondence

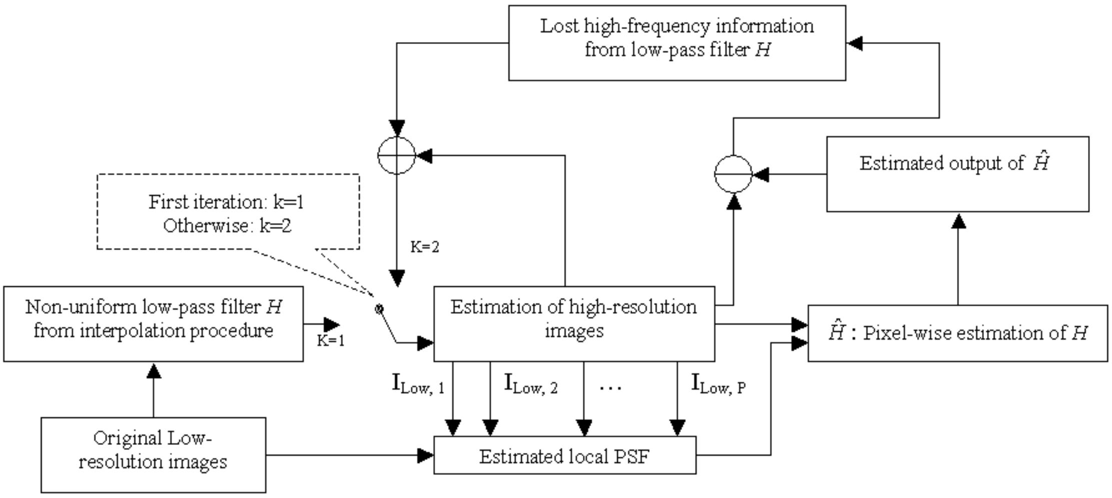

Lost high-frequency information from low-pass filter H
Estimated output of $\hat{H}$
First iteration: k=1
Otherwise: k=2
K=2
Non-uniform low-pass filter H from interpolation procedure
K=1
Estimation of high-resolution images
$\hat{H}$: Pixel-wise estimation of H
$\mathbf{I}_{\text{Low, 1}}$
$\mathbf{I}_{\text{Low, 2}}$
…
$\mathbf{I}_{\text{Low, P}}$
Original Low-resolution images
Estimated local PSF

Figure 1: The flowchart of the algorithm. (All frames are registered.)

are evaluated by the minimum normalized logarithm distance (MNLD):

$$Dist(\mathbf{f}(\mathbf{x}_{1,i}), \mathbf{f}(\mathbf{x}_{1,p})) = \frac{\log(\|\mathbf{f}(\mathbf{x}_{1,i}) - \mathbf{f}(\mathbf{x}_{2,p})\|)}{\|\mathbf{f}(\mathbf{x}_{1,i}) - \mathbf{f}(\mathbf{x}_{2,p})\|} \quad (2)$$

$$\mathbf{x}^c_{2,i} = \arg\min_{\mathbf{x}_{2,p}} Dist(\mathbf{f}(\mathbf{x}_{1,i}), \mathbf{f}(\mathbf{x}_{2,p}))), \quad (3)$$

where pixels $\mathbf{x}_{1,i}$ are from the first frame and $\mathbf{x}_{2,i}$ are from the second frame. $\mathbf{x}_{2,p}$ is the pth pixel from $\mathbf{x}_{2,i}$'s $M \times M$ neighborhood. This distance metric is more robust to pepper and salt type noise. The subsequent procedure is based on the registered frames.

3 Pixel-wise high frequency component compensation

Figure 1 shows the flowchart of the algorithm. Let the tth original high-resolution frames be $\mathbf{I}_{h,t}$. Suppose we have an estimation of $\mathbf{I}_{h,t}$, denoted as $\mathbf{I}^k_{h,t}$. It is a smoothed version of the high-resolution image: $\mathbf{I}^k_{h,t} = h^k_t * \mathbf{I}_{h,t}$, where h^k_t is the blurring function for current estimate. The error of the estimate is:

$$\varepsilon_t = \mathbf{I}_{h,t} - \mathbf{I}^k_{h,t} = \mathbf{I}_{h,t} - h^k_t * \mathbf{I}_{h,t}.$$

True $\mathbf{I}_{h,t}$ is unknown. We use the following estimate instead:

$$\varepsilon^k_t = \mathbf{I}^k_{h,t} - h^k_t * \mathbf{I}^k_{h,t}. \quad (4)$$

Compensate the error back to $\mathbf{I}^k_{h,t}$, we can get a better estimate $\mathbf{I}^{k+1}_{h,t} = \mathbf{I}^k_{h,t} + \varepsilon^k_t$. $\mathbf{I}_{h,t}$ can be reconstructed iteratively in this way.

Now the next task is to find ε^k_t, or $h^k_t * I^k_{h,t}$. In the proposed algorithm, we predict $h^k_t * \mathbf{I}^k_{h,t}$ without explicitly computing h^k_t.

Suppose every pixel on the original low-resolution grid will be projected onto a $q \times q$ grid of the high-resolution image domain. Different from [2, 3], we assume the point spread function (PSF) is unknown, which is more general for real data. Also, no uniform PSF assumption is made. The degrading model of the low resolution images is:

$$\mathbf{I}_{l,t}(\mathbf{x}_i) = \sum_{m=0}^{q-1}\sum_{n=0}^{q-1} f'_t(\mathbf{x}_{i,m,n}) \times \mathbf{I}_{h,t}(\mathbf{x}_{i,m,n}), \quad (5)$$

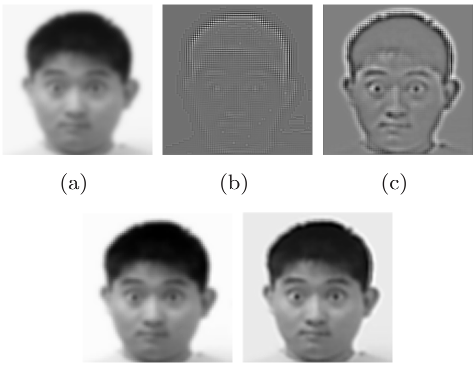
f'_t shows the pixel weight in the $q \times q$ grid of the original high-resolution image. We refer to it as local point spread function (local PSF). h^k_t is closely related with the estimation of f'_t. Later in this section, we will see that $h^k_t * I^k_{h,t}$ is inferred heuristically from f'_t.

This degrading model is solved directly in a simplified way. The current estimate $\mathbf{I}^k_{h,t}$ should also satisfy the degrading model from the MSE sense. Therefore:

$$\mathbf{I}_{l,t}(\mathbf{x}_i) \Leftarrow \sum_{m=0}^{q-1}\sum_{n=0}^{q-1} f'_t(\mathbf{x}_{i,m,n}) \times \mathbf{I}^k_{h,t}(\mathbf{x}_{i,m,n}) \quad (6)$$

It is reasonable to assume that the local PSF keeps unchanged for successive $2p+1$ frames. The optimal function for estimating f'_t on the current $q \times q$ grid is:

$$\begin{aligned} & \mathcal{J}(\mathbf{f'}^k_t(\mathbf{x}_i)) \\ = \; & \sum_{t=-p}^{p} (\mathbf{I}_{l,t} - \sum_{m=0}^{q-1}\sum_{n=0}^{q-1} f'^k_t(\mathbf{x}_i, m, n)\mathbf{I}^k_{h,t}(\mathbf{x}_{i,m,n}))^2 \\ & + \lambda \nabla \mathbf{f'}^k_t(\mathbf{x}_i), \quad (7) \\ & s.t. : \|\mathbf{f'}^k_t(\mathbf{x}_i)\| = 1. \end{aligned}$$



(a) (b) (c)

(d) (e)

Figure 2: (a) is the initial input of the current frame. (b) is the estimated local point spread function for the first iteration. (c) is the estimated high-frequency components in the first iteration. (d) is the reconstructed high-resolution image from the first iteration. It is also the input for the second iteration. (e) is the final high-resolution reconstruction after three iterations.

where:

$$\nabla \mathbf{f'}_t(\mathbf{x}_i) = \|\partial_x \mathbf{f'}_t^k\| + \|\partial_y \mathbf{f'}_t^k\| + \|\partial_{xy} \mathbf{f'}_t^k\| + \|\partial_{yx} \mathbf{f'}_t^k\|;$$

$$0 \le m \le q; 0 \le n \le q;$$

and $q \times q$ matrix $\mathbf{f'}_t^k(\mathbf{x}_i) = [f_t'^k(\mathbf{x}_i, m, n)]_{q\times q}$ is the local PSF on the current $q \times q$ grid.

The first term of $\mathcal{J}(\mathbf{f'}_t^k(\mathbf{x}_i))$ boosts $\mathbf{h}'(\mathbf{x}_i)$ as an impulse function with non-zero at the pixel most similar to the given low-resolution pixel; while the second term is a smoothing term. It keeps the local PSF as a uniform function. A simplified solution is provided for equation 7, which is a direct trade-off between these two terms. We choose a uniform function as the initial values for $\mathbf{f'}_t^k(\mathbf{x}_i)$, and apply one-step steepest descendent update as follows:

$$\widetilde{f_t'^k}(\mathbf{x}_i, r, l) = \frac{\sum_{t=-p}^{p}\{I_{l,t}(\mathbf{x}_i) - M_{t,r,l}\}}{\sum_{t=-p}^{p} I_{h,t}^k(\mathbf{x}_i, r, l)}, \tag{8}$$

where:

$$M_{t,r,l} = \frac{1}{q^2} \sum_{m=0;m\neq r}^{q-1} \sum_{n=0;n\neq l}^{q-1} I_{h,t}^k(\mathbf{x}_i, m, n);$$

$$f_t'^{k,\star}(\mathbf{x}_i, r, l) = \frac{\widetilde{f_t'^k}(\mathbf{x}_i, r, l)}{\|\widetilde{\mathbf{f'}_t^k}(\mathbf{x}_i)\|};$$

$$0 \le r \le q-1, 0 \le l \le q-1;$$

$$\widetilde{\mathbf{f'}_t^k}(\mathbf{x}_i) = [\widetilde{f'}_t^k(\mathbf{x}_i, m, n)]_{q\times q}.$$

$\mathbf{f'}_t^{k,\star}(\mathbf{x}_i) = [f_t'^{k,\star}(\mathbf{x}_i, m, n)]_{q\times q}$ is used as the estimated optimal local PSF for current $q \times q$ grid at kth iteration. Now we relate $f_t'^{k,\star}$ with the estimation of $h_t^k * \mathbf{I}_{h,t}^k$. Equation 6 can also be written as:

$$\mathbf{I}_{l,t}(\mathbf{x}_i) \Leftarrow \sum_{m=1}^{q} \sum_{n=1}^{q} f_t'(\mathbf{x}_{i,m,n}) \times (h_t^k * \mathbf{I}_{h,t})(\mathbf{x}_{i,m,n}) \tag{9}$$

It's clear that h_t^k and f_t' are reciprocally related. For simplification, assume h_t^k has a limited support of 3×3. Then at every pixel, the smoothed output of $h_t^k * \mathbf{I}_{h,t}^k$ is determined only by the intensity and the local PSF of its eight neighborhood. The distribution of $(h_t^k * \mathbf{I}_{h,t}^k)(\mathbf{x}_{i,m,n})$ is modeled by the following Gaussian mixture:

$$Pr((h_t^k * \mathbf{I}_{h,t}^k)(\mathbf{x}_{i,m,n}) | \{\mathbf{I}_{h,t}^k(\mathbf{x}_{i,m,n,j})\}_{j=1,\cdots,8})$$
$$\sim \sum_{i=1}^{8} w_{i,m,n,j} \mathcal{N}(\mathbf{I}_{h,t}^k(\mathbf{x}_{i,m,n,j}); d_j^2); \tag{10}$$

where $\mathbf{x}_{i,m,n,j}(j = 1, \cdots, 8)$ are pixel $\mathbf{x}_{i,m,n}$'s eight neighboring pixels; d_j are Euclidean distance between pixel $\mathbf{x}_{i,m,n}$ and its neighbor $\mathbf{x}_{i,m,n,j}$, which takes value 1 or $\sqrt{2}$. This model actually describes a low-pass procedure characterized by the mixing factor $w_{i,m,n,j}$. The filtered pixel value $(h_t^k * \mathbf{I}_{h,t}^k)(\mathbf{x}_{i,m,n})$ will be as similar as its neighborhood, weighted by $w_{i,m,n,j}$. $w_{i,m,n,j}$ shows the confidence on its neighborhood pixels. This confidence is critical since it captures the characteristics of h_t^k. All the knowledge we currently have for h_t^k is that h_t^k and f_t' are reciprocally related. We use the following heuristic function to model the reciprocal relationship between f_t' and h_t^k.

$$w(\mathbf{x}_{i,m,n,j}) = \exp\{-b_j^2 f_t'^{\star}(\mathbf{x}_i, m, n)^2\}, \tag{11}$$

where b_j is the neighboring pixel's bias from its mean value over the successive p frames:

$$b_j = \mathbf{x}_{i,m,n,j} - \bar{\mathbf{x}}_{i,m,n,j}.$$

By including b_j, less weight will be given to pixels with large deviations, which are most likely outliers contaminated from the data acquisition or inaccurate registration procedure.

Therefore, $(h_t^k * \mathbf{I}_{h,t}^k)(\mathbf{x}_{i,m,n})$ is the solution of:

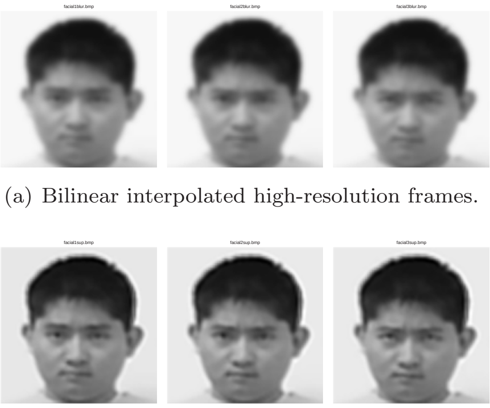
$$\arg\max_{\mathbf{x}^\star} F(\mathbf{x}^\star);$$

$$F(\mathbf{x}^\star) = \sum_{i=1}^{8} w(\mathbf{x}_{i,m,n,j}) \exp\{-\frac{(\mathbf{x}^\star - \mathbf{I}_{h,t}^k(\mathbf{x}_{i,m,n,j}))^2}{d_j^2}\}, \tag{12}$$

which is actually a local MAP estimation. Steepest descendent algorithm is used to solve equation 12.

$$Z = (h_t^k * \mathbf{I}_{h,t}^k);$$

$$Z^{(r+1)}(\mathbf{x}_{i,m,n}) = Z^{(r)}(\mathbf{x}_{i,m,n}) + \mu df \tag{13}$$



(a) Bilinear interpolated high-resolution frames.

(b) Super-resolution frames by the high frequency compensation algorithm.

Figure 3: Examples of the experimental results.

$$\begin{aligned} df &= -\frac{\partial}{\partial \mathbf{x}^\star} F(\mathbf{x}^\star)|_{Z^{(r)}(\mathbf{x}_{i,m,n})} \\ &= \sum_{i=1}^{8} \frac{w(\mathbf{x}_{i,m,n,j})(\mathbf{I}^k_{h,t}(\mathbf{x}_{i,m,n,j}) - Z^{(r)}(\mathbf{x}_{i,m,n}))}{d_j^2} \\ &\quad \exp\{-\frac{(Z^{(r)}(\mathbf{x}_{i,m,n}) - \mathbf{I}^k_{h,t}(\mathbf{x}_{i,m,n,j}))^2}{d_j^2}\} \end{aligned} \quad (14)$$

The entire procedure is repeated. By compensating the lost high-frequency component step by step, the high-resolution videos are recovered. Figure 2 give an example of one iteration for the whole procedure. Initial input of the algorithm are bilinear interpolation of $(2p+1)$ successive frames (here $p = 2, q = 2$). In our experiment, the iteration times are all set to 3. It is because when $k = 3$, the dynamic range of the obtained high-frequency component has been small enough.

4 Experimental evaluation

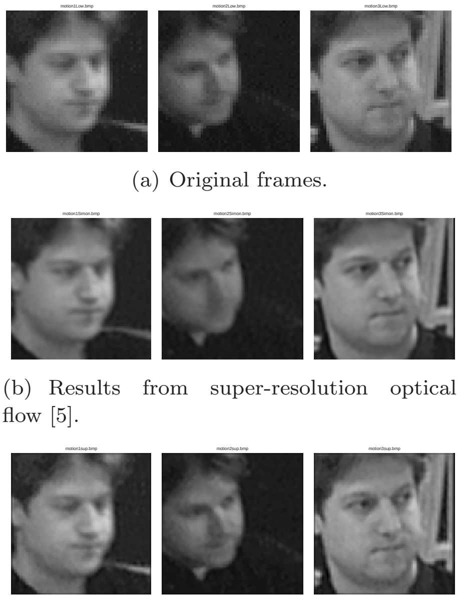
4.1 Videos under different settings

In this section results are shown from our algorithm tested on video sequences with different content and sensors.

1. Videos with changing facial expressions.

The high frequency compensation algorithm enhanced the appearance of subtle changes in facial expression which can be lost in low resolution or blurred sequences. Figure 3 shows results of our algorithm compared to those generated using bilinear interpolation. It is apparent that the bilinear interpolation results in figure 3(a) are severely blurred, whereas our algorithm generates facial features that are significantly clearer, shown in figure 3(b).

2. Videos with large head motions.



(a) Original frames.

(b) Results from super-resolution optical flow [5].

(c) Results from the high frequency compensation algorithm.

Figure 4: Comparison with the super-resolution optical flow. The original video data is courtesy of Dr. Simon Baker. [5]

In figure 4 we compare our results with the super-resolution optical flow algorithm from Baker et. al. using a video sequence from [5]. Our experiments show better performance in certain frames that are problematic for [5]. Specifically, super-resolution optical flow is not able to capture blinking in the first frame of figure 4 and hence introduces blocking artifacts. Our algorithm successfully augments the original image with more perceptually appealing results.

3. Videos from an omni-directional camera.

Omni-directional video cameras are widely used for their 360 degree field of view [6, 7]. However, images from these cameras are typically low resolution and suffer from non-uniform distortion across the image. Our high frequency compensation algorithm can be used to enhance the video quality. Figure 5 shows an example frame from the omni camera used to acquire test sequences. Figures 6(a) and 7(a) show images extracted from indoor and outdoor omni video sequences respectively. The super-resolved images are shown in figures 6(b) and 7(b).

4.2 Quantitative comparison

We compare the results of our algorithm with [3] and [2] using high resolution face videos from [8]. The videos contain substantive facial expressions from various subjects. We sub-sample the original video frames and use them as input. The original video sequences

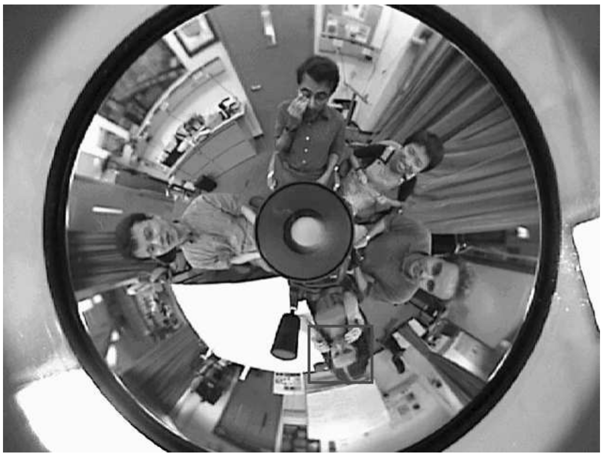


Figure 5: The original frame from omni-directional video camera. The bounding-box shows the detected head area. This area is corresponding to the cropped region in the first image of Fig.6(a).

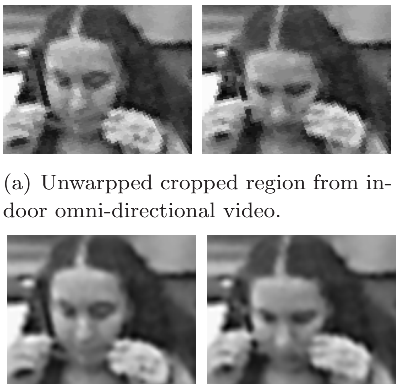
are used as ground-truth for comparison. Examples of the perceptual results are shown in figure 8. Due to legal issues, only the lip patches are shown. Qualitatively, more details are resolved by our algorithm and Borman's algorithm than Zomet's IBP algorithms. However, Borman's algorithm produces blobby images that are perceptually unappealing. The examples in [4] also exhibit the same problems, possibly due to the Huber function prior used leading to excessive constraints on the high frequency components.

The PSNR is computed for each algorithm on a frame-by-frame basis. Figure 9 shows the PSNR curve for the first video sequence in the database. The mean PSNR is computed over all frames and shown in Table 1. The PSNR indicates that our high frequency compensation algorithm exhibits the least distortion.

To better understand the type of information that is well-preserved for the various algorithms, we compute the mean MSE over all frames in different frequency ranges. The results in figure 10 show that the high frequency compensation algorithm has the significantly less low-frequency distortion, however, in the high-frequency range, the distortion is marginally larger. Overall, these comparative results show the effectiveness of our algorithm. Also, our algorithm has a lower computational cost than the others. The quantitative comparison shows that our algorithm is effective

5 Conclusion

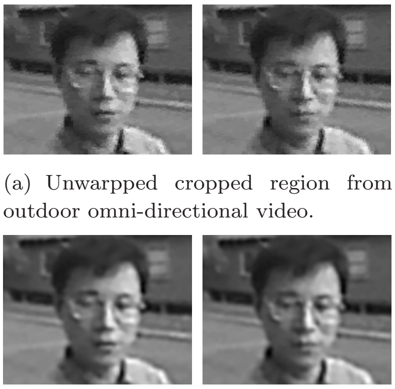
Due to the non-rigidity and non-planarity characteristics of face subject, a lot of existing high-resolution



(a) Unwarpped cropped region from indoor omni-directional video.

(b) Super-resolution recovered results from high frequency compensation algorithm.

Figure 6: Example results from indoor omni-directional video. The super-resolution algorithm removed the blocky and noisy artifacts in the unwarpped omni-directional video.



(a) Unwarpped cropped region from outdoor omni-directional video.

(b) Super-resolution recovered results from high frequency compensation algorithm.

Figure 7: Example results from outdoor omni-directional video. The super-resolution algorithm removed the blocky and noisy artifacts in the unwarpped omni-directional video.

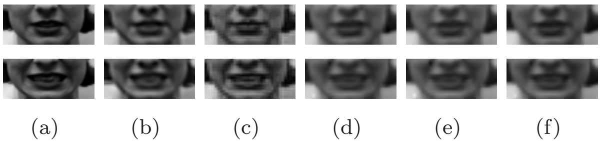


Figure 8: Example of super-resolved lip patches by different algorithms. (a): original images; (b): results from the high frequency compensation algorithm; (c): results from Borman's algorithm; (d): results from IBP algorithm with mean vector updating (no bias detection applied); (e): results from IBP algorithm with median vector updating (no bias detection applied); (f): results from IBP algorithm with median vector updating (with bias detection applied).

compensation	High frequency algorithm algorithm	Borman's algorithm	IBP algorithm with mean	IBP algorithm with median	IBP algorithm with median and bias detection
Mean PSNR	61.7982	61.7161	58.7839	58.8437	59.7318

Table 1: Mean PSNR for different algorithms. Although for our high frequency compensation algorithm and Borman's algorithm, the mean PSNR are similar, our results are perceptually more favorable.

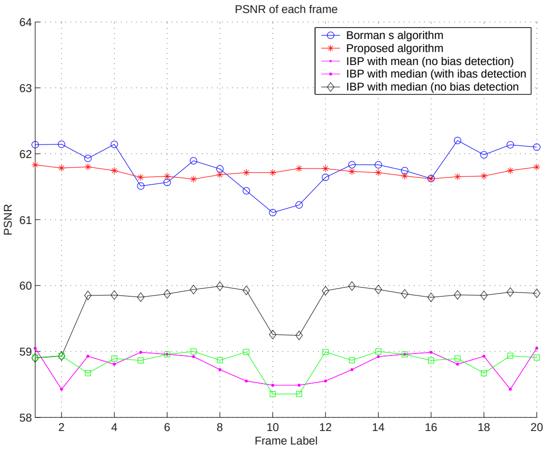


Figure 9: PSNR of different algorithms.

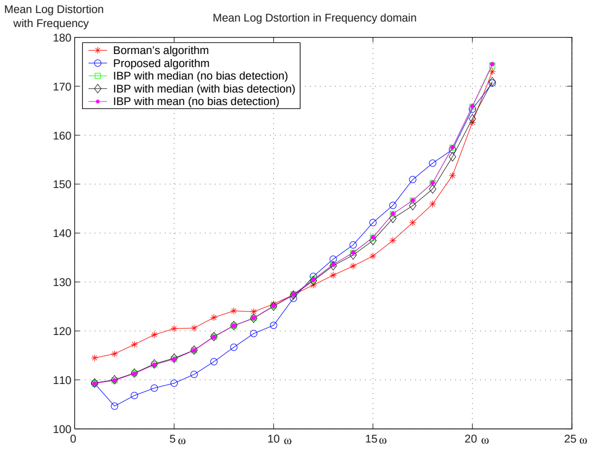


Figure 10: Mean MSE over all frames in different frequency range.

algorithms are not applicable for face video high-resolution reconstruction. This paper proposed a video based super-resolution algorithm based on high frequency component compensation. The frames are registered by minimizing the NLD on the proposed shape feature. The lost high-frequency information is estimated by local MAP criteria, using the registered frames. By compensating the lost high-frequency information, the high-resolution frames are recovered. The computational cost for the algorithm is much lower than the alternatives. Also, although the start point for the algorithm is for face subjects, the algorithm is not limited to faces. One drawback of the high frequency compensation algorithm is that the overall brightness of the image may be altered. We are currently exploring ways to resolve this issue. Also, we are working on combining this promising algorithm with omni-directional face recognition [9] to get a better recognition rate.

References

[1] S. Borman and R. Stevenson. Super-resolution from image sequences - a review. In *Proceedings of Midwest Symposium on Circuits and Systems.*, 1998.

[2] A. Zomet, A. Rav-Acha, and S. Peleg. Robust super resolution. In *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, December. 2001.

[3] S. Borman and R. Stevenson. Simultaneous multi-frame map super-resolution video enhancement using spatio-temporal priors. In *Proceedings of IEEE International Conference on Image Processing.*, October. 1999.

[4] D. P. Capel and A. Zisserman. Super-resolution from multiple views using learnt image models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.*, 2001.

[5] S. Baker and T. Kanade. Super-resolution optical flow . Technical report, Carnegie Mellon University., 1999.

[6] Self reference.

[7] Self reference.

[8] T. Kanade, J. F. Cohn, and Y. Tian. Comprehensive database for facial expression analysis. In *The 4th IEEE International Conference on Automatic Face and Gesture Recognition (FG'00).*, March 2000.

[9] Self reference.