Resolution enhancement of low quality videos using a high-resolution frame

Tuan Q. Pham[a], Lucas J. van Vliet[a], Klamer Schutte[b]

[a]Quantitative Imaging Group, Delft University of Technology, Lorentzweg 1, 2628 CJ, Delft, the Netherlands

[b]TNO Physics and Electronics Laboratory, P.O. Box 96864, 2509 JG, The Hague, the Netherlands

ABSTRACT

This paper proposes an example-based Super-Resolution (SR) algorithm of compressed videos in the Discrete Cosine Transform (DCT) domain. Input to the system is a Low-Resolution (LR) compressed video together with a High-Resolution (HR) still image of similar content. Using a training set of corresponding LR-HR pairs of image patches from the HR still image, high-frequency details are transferred from the HR source to the LR video. The DCT-domain algorithm is much faster than example-based SR in spatial domain[6] because of a reduction in search dimensionality, which is a direct result of the compact and uncorrelated DCT representation. Fast searching techniques like tree-structure vector quantization[16] and coherence search[1] are also key to the improved efficiency. Preliminary results on MJPEG sequence show promising result of the DCT-domain SR synthesis approach.

Keywords: super-resolution synthesis, DCT, video restoration, low-quality videos, MJPEG.

1. INTRODUCTION

Today's increased demand for electronic devices with audiovisual capabilities has resulted in the presence of a video mode in a variety of handheld devices such as digital cameras and video phones. However, while the race for extra functionalities on a single device is on, the performance of these additions are often not very satisfactory. Due to limited memory and computational power, the video quality of these *convergence* products is not comparable to that of dedicated devices. The image size is rather small (320x240 or 640x480) and the compression is often severe (low quality MJPEG, H263). To bring the quality of these videos to the level of SVCD or DVD for (high definition) television replay, software techniques are needed to increase the image resolution.

A popular approach in the signal and image processing community is super-resolution (SR) reconstruction from a moving image sequence.[10,12,19,31] This approach first registers the low-resolution (LR) input frames to a common reference. A high-resolution (HR) image is then constructed by fusion of the LR samples. This HR image is finally deconvolved to recover high-frequency signals that were suppressed by optical blur and/or sensor integration. As a result, the performance of SR reconstruction very much depends on the Signal-to-Noise Ratio (SNR), optical Point Spread Function (PSF), and fill-factor of the CCD sensors.[20] For example, it has been estimated that under a practical assumption of 100% fill-factor, good signal-to-noise ratio (PSNR* $\approx 34dB$) and small registration error ($\max(\Delta_r) \approx 0.125$ pixel), the super-resolution factor is limited to only 1.6.[14] Furthermore, this factor cannot be improved by adding more LR frames. Although many papers claim two-time upsampling results of real videos,[12,31] the actual resolution gain rarely approaches a factor of two because the reconstructed images are often smooth.

Since SR reconstruction has a fundamental limit, many researches have recently shifted towards a Model-Based approach for SR (MBSR).[2,6,9] This second approach differs from the reconstruction approach in the deblurring step where high-frequency signals are *pasted* into the blurry HR image after fusion. Instead of performing a deconvolution, the model-based approach first learns the correspondence between a sharp image

Corresponding author: Tuan Pham: E-mail: pham@tnw.tudelft.nl , Phone: +31 15 278 6054 , Fax: +31 15 278 6740

*Peak Signal-to-Noise Ratio of an image I: $PSNR(I) = 20\log_{10}(max(I)/\sigma_n)$, where σ_n is standard deviation of noise

Visual Communications and Image Processing 2006, edited by John G. Apostolopoulos, Amir Said,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6077, 607708, © 2005 SPIE-IS&T · 0277-786X/05/$15

patch and its smoothed version from a large number of samples stored in a training dataset. It then pastes plausible high-frequency textures into the blurry HR image following the learned correspondences. The logic behind MBSR is very similar to multi-scale texture synthesis,[1, 18, 28] both of which emerged around the same period of time. Similar to texture synthesis, MBSR only performs favorably if a correct texture source is available. Due to its learning and inference in the spatial domain, MBSR is not robust against noise or changes in illumination.[2] MBSR also performs poorly for compressed images,[6] which frequently occur in practical applications.

To resolve the current limitations of MBSR (i.e. lack of suitable texture and non-robust spatial-based synthesis), we propose a SR texture synthesis approach in the DCT domain using a HR image of the same scene. The DCT domain is chosen because DCT is the basis of most video codecs for consumer devices (e.g., MPEG, H.263 and MJPEG). As a result, the LR video needs not be decompressed for SR synthesis. The HR reference image can be provided by the user together with the video since most handheld devices have dual mode: HR still and LR video. As an alternative, it can be selected from still images captured on the same event in HR mode. Even better the camera can be configured to automatically capture an HR image after every scene transition to guarantee the availability of a good HR texture source.

The rest of this paper is presented as follows. Section 2 points out some of the difficulties in applying SR techniques to compressed images. Specifically, there is a slight reduction in the performance of image registration and a severe truncation of the signal bandwidth at medium JPEG compression. Section 3 proposes an example-based SR technique for DCT-based compressed images without the need of decompression. The results of this DCT-based SR synthesis approach is presented in chapter 4. Chapter 5 concludes the paper with potential applications and ideas for further refinement of the algorithm.

2. INFLUENCE OF JPEG COMPRESSION ON SUPER-RESOLUTION

Traditional SR reconstruction methods often involve three sub-tasks: image registration, image fusion, and deconvolution. The level of attainable resolution enhancement depends heavily on the performance of each task. Although there are procedures to measure the performance limit of SR,[14, 20] they were developed for uncompressed data corrupted by space-invariant blur and Gaussian noise. In this section, we analyze the difficulties facing image registration and deconvolution due to a different type of signal degradation: space-variant compression and spectrum-bandwidth reduction by DCT-based compression.

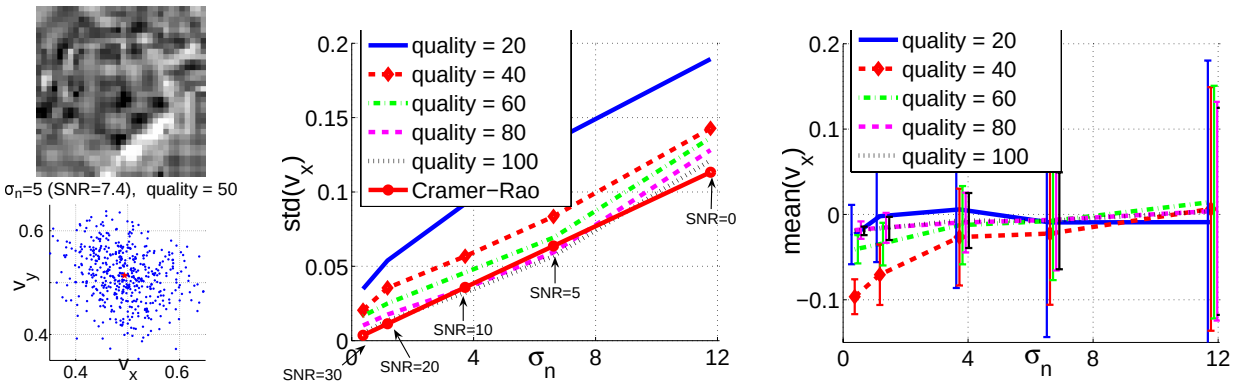
2.1. Performance of registration on compressed images

Image registration is a crucial part in any super-resolution algorithm. The input LR images should be registered with sub-pixel accuracy so that mis-alignment does not become visible even after two- or four-times zoom. Due to the complexity of motion in real videos, registration for video coding is performed as a local shift estimation for every 16×16 macro blocks. If the motion of a block is purely translational, a very precise and unbiased shift estimation can be achieved using an iterative gradient-based method.[15] This shift estimator is shown to be optimal under Gaussian noise,[21] because its variance reaches the Cramer-Rao lower bound:

$$\begin{aligned} \mathrm{var}(v_x) &\geq \sigma^2 \sum_S I_y^2 \Big/ Det(\mathbf{T}) \\ \mathrm{var}(v_x) &\geq \sigma^2 \sum_S I_y^2 \Big/ Det(\mathbf{T}) \end{aligned} \qquad \text{where} \qquad \mathbf{T} = \begin{bmatrix} \sum_S I_x^2 & \sum_S I_x I_y \\ \sum_S I_x I_y & \sum_S I_y^2 \end{bmatrix} \tag{1}$$

where $[v_x \ v_y]$ are the estimated shifts, $[I_x \ I_y]$ are image derivatives along the x- and y-dimensions, S is the region over which the shift is estimated, and $Det(\mathbf{T})$ is the determinant of the gradient structure tensor $\mathbf{T}$. The optimality of the iterative shift estimation can be concluded from figure 1b, where the precision of the shift estimation at JPEG quality 100 (i.e. almost lossless compression except for small quantization errors) is within the proximity of the Cramer-Rao lower bound (dotted line versus continuous line with round marker).

Figure 1 also shows that the performance of our shift estimator does not degrade significantly for compressed images at normal JPEG quality (quality$\geq$60). Figure 1a shows a 16×16 noisy and compressed Pentagon image.



(a) normal spread (b) precision versus Cramer-Rao bound (c) bias of shift estimation

Figure 1. Performance of shift estimation on two compressed noisy 16×16 macro blocks that are [0.5 0.5] pixels apart.

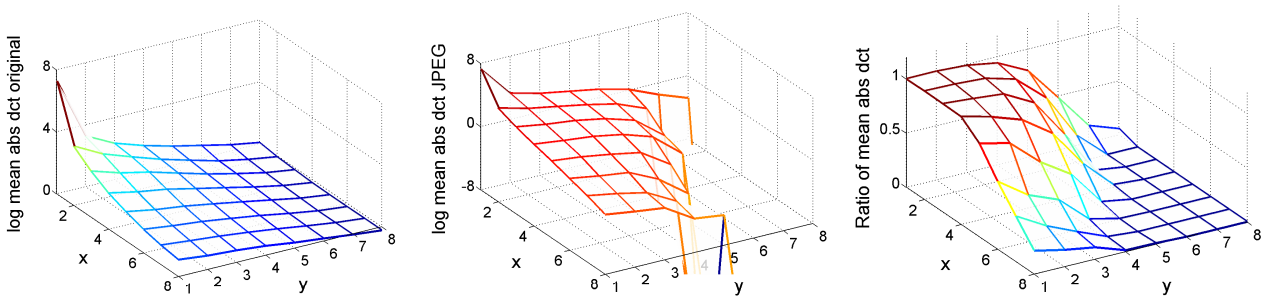
The image is generated from a 1024×1024 original image after a Gaussian blurring ($\sigma = 16$) followed by 32-times downsampling, Gaussian noise addition ($\sigma_n = 5$), and JPEG compression (quality=50). Registration of two such images ([0.5 0.5] pixels apart) with different noise realizations produces a normal distributed result around the true shift (figure 1a). Mean and deviation of the estimated shifts at various noise levels and compression qualities are plotted in figure 1b-c, in which each data point is computed from the statistics of 500 different noise realizations. Figure 1b shows a linear relationship between shift precision and noise as dictated by the Cramer-Rao bound in equation (1). Surprisingly, the precision of shift estimation at normal JPEG quality (quality≥60) does not deviate significantly from the optimal precision. A small bias for normal JPEG quality is observed in figure 1c. All in all, figures 1b-c show that the registration is accurate enough for three-times SR of images at normal SNR[†] ($SNR \geq 10dB$) and normal compression (quality≥60). This is because the registration errors do not exceed 0.15 LR pixel or 0.45 HR pixel in 99% of cases (confidence interval bounded by three times the observed standard deviation). The underlying assumption is that if the registration error is less than 0.5 HR pixel, it does not cause a visible artifact in the HR image after fusion.

2.2. Spectrum reduction by DCT-based compression

Although medium JPEG compression does not significantly degrade the accuracy of registration, it does corrupt the signal in a destructive way. This is the most noticeable drawback of a DCT-based codec because not only small features are suppressed but spurious details are also introduced in the compressed image. Because many high-frequency DCT coefficients are set to zero by quantization, every 8×8 block in the compressed image is essentially low-pass filtered with cyclic border conditions. Coarse quantization levels also mean sudden jumps of quantized DCT coefficients as their values gradually change from one level to the next. This results in noticeable intensity and texture mismatches across block boundaries, also known as blocking artifacts.

Although a single compressed frame is degraded locally within each 8×8 block, the degradation approaches that of global low-pass filter in the HR image constructed by fusion of multiple moving frames. Because all input frames move, each HR pixel receives contributions from multiple LR pixels at different offsets with respect to the 8×8 coded block. As a result, even though the compression error is space-variant within a coded block,[22] the error averages out to a reduced and space-invariant noise over the HR image. Ringing and blocking artifacts are also suppressed by multi-frame fusion.[8] What is not recoverable by multi-frame fusion is the signal blur. As illustrated in figure 2b, DCT quantization causes a sudden attenuation of DCT coefficients at high frequencies. The DCT reduction factor plotted in figure 2c is comparable to the compression's frequency transfer function. Zero values at the tail of this transfer function indicates that all DCT coefficients at high frequencies are truncated to zero. These high-frequency signals are therefore irrecoverable by deconvolution.

[†]Signal-to-noise ratio of an image is $10 \log_{10}(\sigma_I^2/\sigma_n^2)$, where σ_I^2 & σ_n^2 are variances of the uncorrupted image & noise



(a) average original DCT values (b) average quantized DCT values (c) Transfer function as (b)/(a)

Figure 2. Overall attenuation of high-frequency coefficients due to JPEG compression at quality 80.

3. SUPER-RESOLUTION SYNTHESIS

Because compression at low-to-medium quality permanently damages the signal spectrum, deconvolution does not yield a satisfactory SR result.[25] The result suffers from unresolvable details and ringing artifacts around edges. To overcome this limitation, prior knowledge is often incorporated into the restoration process. SR synthesis[2,6,9] uses a set of HR reference images to predict the missing HR information in the LR inputs. In this section, we review an example-based SR technique and propose an adaptation for better handling of input images in the DCT domain.

3.1. Texture synthesis approach to image restoration

Example-based SR[6] is one of the first attempts to incorporate a strong prior knowledge into the restoration process. From the observation that the high-frequency bands of a LR image corresponds to the medium-frequency bands of an HR image, Freeman proposed a patch-wise synthesis approach that searches for a mid-band frequency match from a database and transfer the corresponding hi-band frequency signal back to the interpolated LR image. The search constraints are such that the intensities of the newly synthesized HR patch agree well with the surrounding patches.

The original example-based SR algorithm, however, is known to perform poorly on compressed images.[5,6] Compression artifacts are often mistaken as signal and are therefore amplified rather than suppressed by the SR synthesis process. Fortunately, this shortcoming can be resolved by using multi-frame fusion to reduce the compression artifacts prior to texture synthesis (see figure 3). The modified example-based SR algorithm can then be seen as a combination of SR reconstruction and texture synthesis.

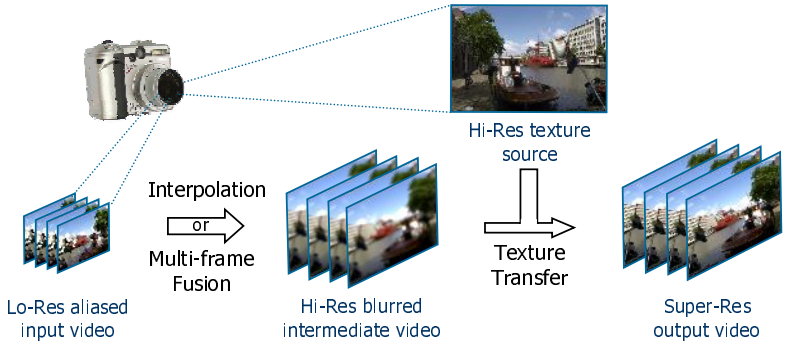


Figure 3. Proposed modification to Freeman's example-based SR[6] algorithm: using multi-frame fusion instead of a single-frame upsampling in the interpolation step to prevent the amplification of compression artifacts by texture synthesis.

3.2. Super-resolution synthesis in the DCT domain

Realizing the fact that band-pass frequency information is available directly in the DCT domain, we choose to do example-based SR directly in the DCT domain. This has a number of advantages. First, no inverse DCT is needed because the input images are already represented in the DCT format. Second, band-pass selection is simply a matter of selecting DCT coefficients. Finally, since DCT is a compact and uncorrelated representation of the spatial signal, computational savings can be achieved by comparing only a subset of the full 8×8 DCT coefficients. In natural images, for example, the first 10 AC coefficients along the standard JPEG zigzag scan path capture 85% of total signal variance. The comparison of high-frequency DCT coefficients should be avoided because these coefficients are often truncated to zero by heavy quantization.

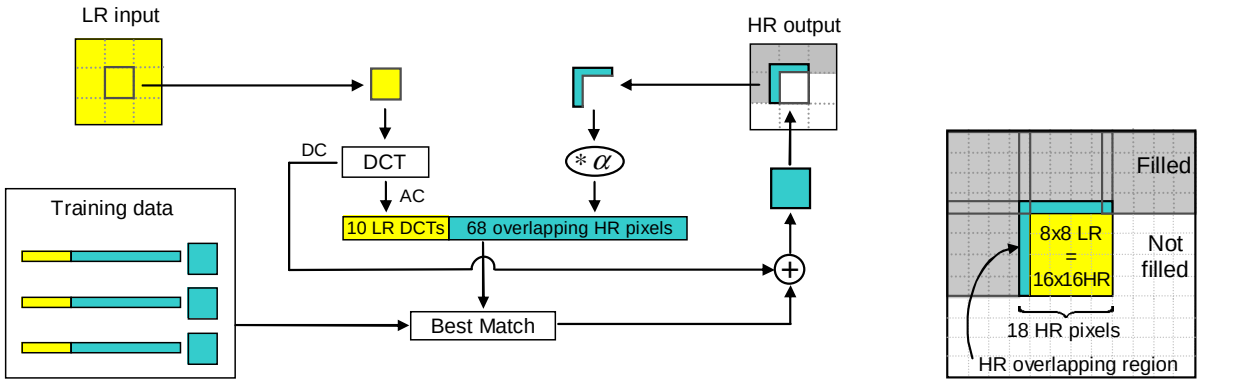
Another important feature of our SR algorithm is a stricter constraint on the HR texture source. Unlike Freeman,[6] who constructs a huge database of training patches for multi-purpose SR, we restrict the texture source to images of similar content captured by the HR still mode of the same camera. Though it sounds very limiting, the assumption is quite practical because most handheld devices have dual video and still capturing modes. The user or, ultimately, the camera system can be trained to capture such a HR still image together with the video. This suitable texture source increases the chance of finding a good match and allows spatial coherence[1] to speedup patch matching in subsequent frames.

The procedure for SR synthesis in the DCT domain is described as follows:

1. Rough affine registration of the first LR frame against the HR texture source: this is necessary so that the training LR patches can be produced at the same zoom and orientation of the input images. To avoid a full decompression of the input, the LR-to-HR registration can be performed on the DC component of the compressed input. Once the first LR frame is registered to the HR source, other frames in the sequence can also be registered by concatenation of background motion computed from the MPEG motion field.[29] In most cases, only the scene translation and rotation need to be corrected because consumer cameras and video phones often disable zoom while capturing.

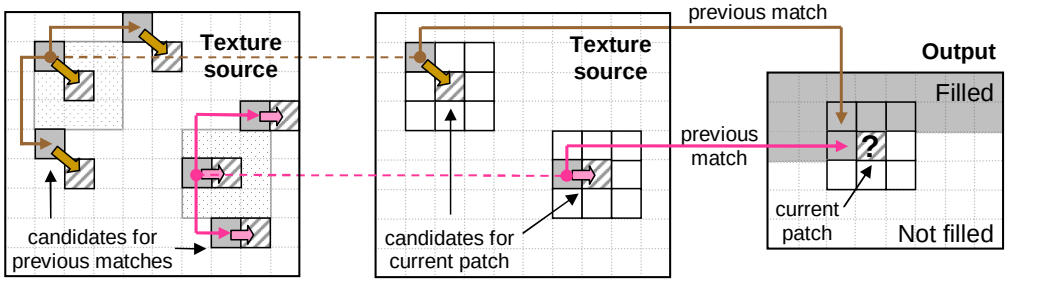
2. LR-HR training pair generation: similar to the spatial-domain example-based SR,[6] HR texture is inferred from training pairs of corresponding LR-HR reference patches. A LR texture image is subsampled from the HR texture source using the registration parameters found in step 1. 8×8 LR training patches are taken from this LR texture image at every integer pixel position. To account for some rotational motions in the input video, four more patches with slight tilts ($\Delta_\theta = -\frac{\pi}{8}, -\frac{\pi}{16}, \frac{\pi}{16}, \frac{\pi}{8}$) are also sampled around each on-grid LR training patch. The corresponding zero-mean HR patches cover a slightly larger neighborhood than that of the LR patches (see figure 4b where an 18×18 HR patch contains an 8×8 LR patch and 68 overlapping HR pixels). The LR patches are encoded using DCT with block size 8. The first ten AC coefficients together with 68 overlapping HR pixels form a 78-dimension vector in the search space (figure 4a).

3. Search for matching texture source: HR texture is synthesized in a raster scan order for every 8×8 DCT block in the LR compressed input. We look for a HR patch that satisfies two constraints: the HR patch must possess similar frequency information contained in the LR data, and its boundary should match the already synthesized patches. The search vector is constructed from the first 10 AC coefficients of the input 8×8 normalized quantized DCT (directly after Huffman decoding and before dequantization[26]) and 68 overlapping pixels from the previously synthesized neighboring blocks. Since the frequency constraint is more important than the spatial coherence constraint, the intensities components of the search vector are multiplied by a small weight ($\alpha = 0.05$, see figure 4a). Euclidean norm is used for the matching and the HR texture linked to the training vector with the smallest norm is selected. For the first frame, a fast brute-force search over the whole dataset is possible using tree-structure vector quantization.[16] For subsequent frames, the synthesized result of the previous frame can be used to reduce the search space by temporal coherence search. Motion vector from the MPEG stream can be used to locate the position of a current DCT block in the previous frame, and hence the position of probable HR patch in the texture source. In addition to temporal coherence, spatial coherence[1] with previously synthesized data within the current frame can also be used. Figure 5 shows how k-coherence search[24] (k=1,2) selects matching candidates for a new patch from spatially coherent HR patches in the texture source.



(a) DCT-domain SR synthesis process (b) partially synthesized HR image

Figure 4. Example-based SR in the DCT domain: synthesizing HR image patches in a raster-order scan.



(a) k-coherence search (k=2) (b) coherence search (k=1) (c) partially synthesized result

Figure 5. Coherence search: candidates for the current block in figure 5c (striped block with a question mark) can be found around the texture sources of previously synthesized blocks. The process can be done recursively, resulting in k-coherence search in figure 5a.

4. RESULTS AND ANALYSIS

We apply the DCT-based SR synthesis algorithm to a low-quality *carphone* sequence. The QCIF sequence (176×144) is compressed using MJPEG codec (each frame is an JPEG image) at quality 50. The HR texture source in figure 6a is the first frame of the same sequence at CIF resolution (352×288). Figure 6b and 6c show frame 20 of the LR input and the synthesized SR output, respectively. Although only two-time zoom is applied, the level of resolution enhancement is actually far greater than two because the input is compressed at very poor quality.

(a) texture source (CIF frame 0) (b) MJPEG input (QCIF frame 20) (c) 2-time SR synthesis

Figure 6. Two-time super-resolution of low-quality video using DCT-domain SR synthesis.

However, the SR result in figure 6c is not without a problem. Block boundaries are visible in several places. In addition, regions outside the car window are not synthesized very well because a suitable texture source is lacking. To alleviate the first problem, some blocking artifact concealment has to be implemented. The boundary mismatch can also be reduced by using a subpixel sampling of the LR training patches and a finer perturbation of patch tilt $\Delta\theta$ in step 2 of the algorithm. The second problem is more difficult to solve for this sequence because the background content keeps changing. The natural view outside the window can be better reconstructed by using an extended set of natural texture. However, since the background is rarely the focus of attention, it could be left unchanged or simply enhanced by a Laplacian filter.

We also compare the result of our DCT-based SR synthesis with that of an improved example-based SR algorithm.[4] This spatial-based SR algorithm uses first and second derivatives at each pixel to search for K-nearest neighbors (K=5) of every 3×3 LR input patch. It then combines the K neighbors using Local Linear Embedding (LLE)[23] to generate a new HR patch that is a linear combination of existing patches in the training set. By this LLE inference, the authors showed an improved mean square error of the new SR image over that of Freeman's nearest neighbor approach.

Images in figure 7 show the same facial region in frame 20 of the *carphone* sequence. The effect of low resolution and severe compression is manifested in figure 7b, whose facial details such as eyes and ears appear blurred compared to that of the original CIF image in figure 7a. JPEG compression also causes ringing artifacts around the face and on the car interior. DCT-domain SR synthesis successfully recovers all facial and hair details. Several LR patches did not have a good enough match from the training set so the high-frequency information is not imported. Occasional mis-registration also causes some edge jaggedness along linear interior structures. However, given all the mentioned shortcomings, DCT-based SR synthesis still outperforms example-based SR using first and second derivatives. The gradient-based SR result in figure 7d shows no sign of detail improvement. On the contrary, the overall sharpness enhancement results from increased ringing and other compression artifacts. As previously stated, this is a typical problem of many other intensity-domain example-based SR algorithms.[5,6]

(a) ground truth (CIF) (b) MJPEG input (QCIF) (c) DCT-domain SR (d) intensity-domain SR[4]

Figure 7. Comparison of two-time SR synthesis of an MJPEG compressed frame in the DCT and intensity domain.

5. FUTURE RESEARCH DIRECTIONS

The experimental results in section 4 clearly show the advantages of DCT-based synthesis over spatial-based synthesis for compressed images. However, the presented algorithm is rather basic and the results can be improved. This section is a collection of ideas for further development of DCT-domain SR synthesis. In particular, four subtasks need improvement: search space pruning, matching criteria, selective synthesis, and error concealment.

5.1. Accurate LR-HR registration for an improved coherence search

Coherence search is an effective way to prune the search space for SR synthesis. Different from texture synthesis, in which the stochastic texture needs not be exactly constructed, SR synthesis requires a plausible reconstruction of patterns well-recognized by the human eyes. HR texture patches should therefore ideally come from the same

actual objects. This can be achieved by incorporating motion tracking in the SR synthesis algorithm. Although motion vectors of macro blocks are available as part of an MPEG compressed stream, they are optimized for coding purposes and may not reflect the true optic flow. Consequently, coherence search should look for the matching position of a LR patch in the texture source around the location given by the optic flow ($\Delta_{x,y} = -1, 0, 1$). In addition, since the orientation of an 8×8 LR block is computable from its DCT coefficients,[13] the tilt Δ_θ of the oriented HR texture patch should also be sampled around this orientation (see the algorithm in section 3.2). In other words, with the zoom corrected in step 1 for the whole sequence, step 2 should find a rigid registration of LR patch against the LR texture source. The add-on registration parameters $\{\Delta_x, \Delta_y, \Delta_\theta\}$ can be refined to subpixel accuracy by fitting a parabola to find local maxima of the presumably smooth error surface. This accurate localization of the coherent texture patch improves the LR input-HR source correspondence and reduces mismatch between HR blocks after texture synthesis.

5.2. Re-weighting the DCT coefficients

Because example-based SR compares two filtered LR patches using their intensities, the weights given to all elements in the search vector are equalled. DCT-based SR synthesis, on the other hand, compares two LR patches based on their AC coefficients in the DCT domain. Since each AC coefficient carries a different signal energy,[17] its weight should be normalized by its variance computed over all coded blocks of the image. However, the problem is not that straightforward because quantization adds further variations to the compressed coefficients. It has been shown[22] that the DCT quantization noise is uniform for non-zero coefficients and it is Laplacian distributed for zero coefficients. Because we only use the first 10 AC coefficients in our matching, these coefficients are most likely non-zero and are therefore uniformly distributed within their quantization interval. To make the variance of this quantization noise equal, we use the quantized DCT coefficients for matching (i.e. DCT coefficient divided by its quantization level). Although the current weighting scheme aims for the same quantization noise across all DCT coefficients, it should be revised to incorporate the different signal energy contained in each coefficient.

5.3. Improvements on synthesis

Due to reasons such as noise, lack of input details, or lack of suitable texture, SR synthesis occasionally produces unsatisfactory results as seen in figure 7c. Existing techniques such as Local Linear Embedding (LLE),[23] brute-force search, or Region Of Interest (ROI) processing can be used to improve the yield of SR synthesis. LLE, for example, is not only applicable to spatial-based SR synthesis[4] but also to our DCT-domain SR synthesis. Full database search using fast nearest neighbor search algorithms[16] can be used if a suitable HR texture patch is not found by coherence search. Because we require a good texture source in our problem setting, the number of times the brute-force search is used will generally be small. However, if the success rate of coherence search falls, it is a good time to signal the hardware to capture another HR texture source. Finally, to improve the frame throughput of the SR process, only selected regions of interest will be synthesized. This is especially applicable to SR of video because viewers tends to fixate on a limited number of regions in an image that they deem interesting.[30] These ROIs usually have high spatial gradient or temporal gradient or both. Low contrast background, for example, are not the focus of interest in the *carphone* sequence in section 4.

5.4. Boundary mismatch concealment

Due to a large size of the standard DCT-coded block (usually 8×8 in JPEG and MPEG1/2, but can also be 4×4 or 16×16 in MPEG-4), a small misalignment of the LR input against the HR texture can result in visible artifacts at output block boundaries. One way to alleviate this problem is to synthesize HR texture for the in-between DCT blocks as well. The DCT coefficients of these in-between blocks is computable from those of adjadent blocks.[3] The largely overlapping synthesized HR blocks are then blended together using a reducing weight from the block center.

While HR blocking artifacts can be reduced by the extraneous in-between block synthesis, intensity mismatches may still occur due to poor texture synthesis. This is especially true when the LR image itself has severe blocking artifacts, and the in-between DCT blocks are then not predicted well from their neighbors. A better method for error concealment is required in these cases. Pixels with high intensity mismatch are detected and re-synthesized.[18] Note that this overlap re-synthesis is done in the spatial domain because both the intermediate HR output and the texture source are represented with real intensities.

6. CONCLUSIONS AND DISCUSSIONS

We have presented an example-based algorithm for super-resolution of compressed videos. The input of the algorithm comes directly from the quantized DCT video stream. The SR performance, however, strongly depends on the similarity between the HR texture source and the LR video. As a result, it works best if the HR texture source is captured along with the video.

Apart from the application of video upscaling, the DCT-domain SR synthesis can be used in a number of other applications. The SR scheme can be implemented in hardware (e.g. inside TVs, video players, or capturing devices themselves), and the performance can be significantly increased with an automatic HR image acquisition after a scene change or a time trigger. The SR algorithm is also applicable to multimedia coding. Current video codecs can be modified to encode most frames in LR and only some key frames in HR to serve as a texture source for SR synthesis at the decoding end. Another coding application of SR synthesis is the (re-)compression of image archives of similar content. This technique resembles vector quantization encoding[7] and is especially useful to reclaim disk space or flash memory as the need arises.

While the idea of using a HR frame to enhance video quality is commercially viable, several considerations must be taken into account. Because example-based SR is not a reconstruction method, the correctness of the synthesized output is questionable. Such a software is therefore not suitable for scientific and forensic purposes. If targeted at the entertainment business, on the other hand, the software should have more emphasis on synthesis of high quality and realistic faces. Strong model-based priors such as eigen-faces[27] or tensor faces[11] together with face recognition are useful in this case.

REFERENCES

1. M. Ashikhmin. Synthesizing natural textures. In *Proc. of ACM Symposium on Interactive 3D Graphics*, 2001. pp. 217–226.
2. S. Baker and T. Kanade. Limits on super-resolution and how to break them. In *IEEE PAMI*, 24(9):1167–1183, 2002.
3. S.-F. Chang and D.G. Messerschmitt. Manipulation and compositing of MC-DCT compressed video. In *IEEE J. of Selected Areas in Comm.*, 13(1):1–11, 1995.
4. H. Chang, D.Y. Yeung, and Y. Xiong. Super-resolution through neighbor embedding. In *Proc. of CVPR04*, 2004. pp. 275–282.
5. R. Destobbeleire. Super-Resolution. IMPA, Brasil, May-June, 2002. Training report available at `http://www.visgraf.impa.br/Projects/super-res/`
6. W.T. Freeman, T.R. Jones and E.C. Pasztor. Example-based super-resolution. In *IEEE Computer Graphics & Applications*, 22(2):56–65, 2002.
7. A. Gersho and R.M. Gray. Vector quantization and signal compression. Kluwer Academic Publishers, 1992.
8. B.K. Gunturk, and Y. Altunbasak, and R.M. Mersereau. Multiframe blocking-artifact reduction for transform-coded video. In *IEEE Trans. Circuits and Systems for Video Technology*, 12(4):276–282, 2002.
9. A. Hertzmann, C.E. Jacobs, N. Oliver, B. Curless and D.H. Salesin. Image analogies. In *Proc. of SIGGRAPH'01*, 2001. pp. 327–340.
10. M. Irani and S. Peleg. Improving resolution by image registration. *CVGIP*, 53:231–239, 1991.
11. K. Jia and S. Gong . Multi-Modal Tensor Face for Simultaneous Super-Resolution and Recognition. In *Proc. IEEE ICCV*, 2005.
12. Z. Jiang, T.-T. Wong and H. Bao. Practical super-resolution from dynamic video sequenes. In *Proc. of IEEE CVPR'03*, Wisconsin, USA, 2003. pp. 549–554.
13. P. Ladret and A. Guérin-Dungué. Categorisation and retrieval of scene photographs from a JPEG compressed database. *Patt. Anal. & Appl.*, 4:185–199, 2001.
14. Z. Lin and H-Y. Shum. Fundamental limits of reconstruction-based superresolution algorithms under local translation. *PAMI*, 26(1):83–97, 2004.
15. B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *DARPA81*, pages 121–130, 1981.

16. C. Merkwirth, U. Parlitz and W. Lauterborn. Fast nearest-neighbor searching for nonlinear signal processing. *Phys. Rev. E*, 62(2):2089-2097, 2000.
17. F. Müller. Distribution shape of two-dimensional DCT coefficients of natural images. *Electronics Letters*, 29(22):1935-1936, 1993.
18. A. Nealen and M. Alexa. Hybrid texture synthesis. In *Proc. of 14th Eurographics Workshop on Rendering*, 2003. pp. 97–105.
19. A. Patti, M. Sezan and A. Tekalp. Superresolution video reconstruction. with arbitrary sampling lattices and nonzero aperture time. In *IEEE TIP*, 6(8):1064–1078, 1997.
20. T.Q. Pham, L.J. van Vliet, and K. Schutte. Influence of signal-to-noise ratio and point spread function on limits of superresolution. In *Image Processing: Algorithms and Systems IV*, Proc. of SPIE Electronic Imaging, Vol. 5672, San Jose, January 2005. pp. 169–180.
21. T.Q. Pham, M. Bezuijen, L.J. van Vliet, K. Schutte, and C.L. Luengo Hendriks. Performance of optimal registration estimators. In Z. Rahman, R.A. Schowengerdt, and S.E. Reichenbach, editors, *Visual Information Processing XIV*, volume 5817 of *SPIE*, pages 133–144, 2005.
22. M.A. Robertson and R.L. Stevenson. DCT quantization noise in compressed images. In *IEEE Trans. Circuits Syst. Video Techn.*, 15(1):27–38, 2005.
23. S.T. Roweis, and L.K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.
24. X. Tong, J. Zhang, L. Liu, X. Wang, B. Guo, and H.-Y. Shum. Synthesis of bidirectional texture functions on arbitrary surfaces. In *Proc. of SIGGRAPH'02*, 2002. pp. 665-672.
25. C. Andrew Segall, Aggelos K. Katsaggelos, Rafael Molina, and Javier Mateos Super-resolution from compressed video. In *Super-Resolution Imaging*, S. Chaudhuri (ed.), chapter 9, 211-242, Kluwer Academic Publishers, 2001
26. G.K. Wallace, The JPEG still picture compression standard. In *Commun. ACM*, 34(4):30–44, 1991.
27. X. Wang and X. Tang. Hallucinating face by eigentransformation. *IEEE Trans. on Systems, Man and Cybernetics, Part C* 35(3):425-434, 2005.
28. L.-Y. Wei and M. Levoy. Fast texture synthesis using tree-structured vector quantization. In *Proc. of SIGGRAPH'00*, 2000. pp. 479–488.
29. T. Wiegand, E. Steinbach and B. Girod. Affine multi-frame motion-compensated prediction. In *IEEE Trans. Circuits Syst. Video Techn.*, 15(2):197-209, 2005.
30. A.L. Yarbus. *Eye movement and vision.* Plenum Press, New York, 1967.
31. W. Zhao and H.S. Sawhney. Is super-resolution with optical flow feasible? In *Proc. of ECCV 02*, Springer-Verlag, 2002. pp. 599-613.