

Improved POCS reconstruction of stereoscopic views

Ryszard Stasiński* and Janusz Konrad†

* *Institute of Electronics and Telecommunications*

Technical University of Poznań

Piotrowo 3A, 60-965 Poznań, Poland

e-mail: rstasins@et.put.poznan.pl

† *Department of Electrical and Computer Engineering*

Boston University

8 Saint Mary's St., Boston, MA 02215, USA

e-mail: jkonrad@bu.edu

Abstract

This paper presents an application of the projection onto convex sets (POCS) framework to the reconstruction of intermediate stereoscopic views. Such views are needed in 3-D viewing in order to simulate the so-called “look-around” as well as to adjust the perceived depth (interocular adjustment). The basic problem in the above reconstruction is that of the recovery of a regularly-sampled image from its irregularly-spaced samples due to disparity compensation. This problem also arises in other image processing and coding applications, such as multiple-frame motion compensation or video frame rate conversion. In our POCS-based approach to view reconstruction, two projection operators are used: bandwidth limitation and sample substitution. The bandwidth limitation can be implemented in the original domain by means of lowpass FIR filtering but we opt for a frequency-domain implementation by means of windowing. The results reported here improve our original POCS-based reconstruction method by locally adapting the algorithm to the density of image samples. We also extend the method to color images through an implementation in the luminance-chrominance space.

I. INTRODUCTION

Stereoscopy and multiscopy are two means of delivering depth cues to the human eye. In stereoscopic viewing, a single stereo pair is used thus causing motion/parallax conflict when observer's head changes position (objects are perceived to have rotated). This problem is addressed in multiscopic viewing by presenting a correct stereo pair at each observer position. The numerous stereo pairs needed for a natural (sufficiently continuous) "look-around" need to be either transmitted, thus increasing transmission costs, or computed at the receiver [1], [2], [3], thus requiring sufficient computing power. Such intermediate views must be also computed when interocular distance in a stereo pair needs to be changed; this can be thought of as depth adjustment [4] to accommodate viewers with either excessive (hyper) stereo acuity or with partial stereo blindness.

In either case, new views need to be reconstructed based on views acquired by real cameras. Since, unlike in computer vision applications, in stereo viewing the camera baseline is quite small (typically 6.5–7.0 cm), simple disparity-compensated image interpolation usually suffices [5]. Such interpolation is similar to motion-compensated interpolation used in video compression and in frame-rate conversion. Both in disparity-compensated and in motion-compensated interpolation, a typical problem is that of recovering regularly-spaced image samples (intensity and color) based on irregularly-spaced samples that result from either disparity or motion compensation. An important aspect of such interpolation is its computational complexity; ultimately, the interpolation should be executed in real time in the decoder. Therefore, one of the concerns in this paper is the computational complexity of the proposed method.

The above intermediate view interpolation is a special case of general image interpolation where the grids of known and unknown samples can each be defined as a *regular* (periodic) or *irregular* sampling structure (grid). There are four scenarios for sampling grid combinations of known/unknown samples:

1. *regular* \rightarrow *regular* : simplest case where interpolation filters are space-invariant; for example, interpolating filters in typical image up-conversion (e.g., by 2×2),
2. *regular* \rightarrow *irregular* : more difficult case where interpolation filters are space-variant since samples to be recovered have varying positions with respect to the regularly-spaced known samples (the continuous interpolating kernel has finite support, and the coefficients of each space-variant digital filter are samples from this kernel); for example, bilinear or bicubic interpolation [6] as used in *backward* motion compensation in video coding,
3. *irregular* \rightarrow *regular* : still more difficult case where interpolation filters are space-variant but may not be easily specified in general case; for example, *forward* motion-compensation in advanced video coding/interpolation; this is the case studied here,
4. *irregular* \rightarrow *irregular* : the most general case for which applications have not clearly emerged yet.

The first two cases above have been extensively treated in the literature and have found numerous practical applications in image processing and coding. The *irregular* \rightarrow *regular* interpolation is more difficult and has

been explored to a lesser degree. The primary reason for this difficulty is associated with the extension of Shannon's sampling theory for regular grids to signals defined over irregular grids. Although such theory has been developed in the past [7], [8], and algorithms have been proposed for band-limited [8], [9], and non band-limited spaces [10], [11], it imposes constraints on the sampling grid that are too restrictive for many applications. In particular, the usefulness of this theory and the corresponding algorithms is very limited in the case of disparity-compensated view interpolation or motion-compensated video coding; theoretically derived constraints on the maximum spacing of irregular samples under the perfect reconstruction condition cannot be satisfied by arbitrarily-distributed image samples after disparity or motion compensation. By relaxing the perfect reconstruction condition, other methods have been proposed such as the projection onto convex sets (POCS) [12], polynomial interpolation [13], reconstruction based on biorthogonal spline wavelets [14], or regularization in spline spaces [15].

In this paper, we extend a method for the reconstruction of intermediate images that we had proposed in the past [16]. The method is based on the theory of projections onto convex sets [17] and applied in the context of *irregular* \rightarrow *regular* interpolation. This theory has been successfully used for various image processing/coding applications in the past (e.g., in [18]). The underlying principle of our iterative algorithm is based on the work by Sauer and Allebach [12] and is quite similar to that recently developed by Sharaf and Marvasti [13], however its implementation is quite different. In this paper, we extend our original POCS-based approach implemented on oversampled lattices by locally adapting the algorithm to the density of image samples. Both the lattice oversampling and adaptation to sample density lead to reduced computational complexity and faster convergence. We also extend the method to color images by implementing it in the luminance-chrominance ($Y-U-V$) space.

The paper is organized as follows. In Section II we describe the proposed approach including relaxation coefficient adaptation and frequency-domain implementation of bandwidth limitation. In Section III, we discuss the implementation issues, memory management and filter design. In Section IV, we describe an application of the proposed method to intermediate view reconstruction, while in Section V we present various experimental results for synthetic and natural disparities. We draw conclusions in Section VI.

II. PROPOSED APPROACH

First, we review the basic POCS-based image reconstruction developed by us earlier [16]. Let $g = \{g(\mathbf{x}), \mathbf{x} = (x, y)^T \in R^2\}$ describe intensity of a 2-D projection of the 3-D world onto a continuous image plane and let $g_\Lambda = \{g(\mathbf{x}), \mathbf{x} \in \Lambda\}$ describe intensity of a discrete image obtained from g by sampling over a lattice $\Lambda \subset R^2$ [19]. Let's assume that g is band-limited, i.e., $G(\mathbf{f}) = \mathcal{F}\{g\} = 0$ for $\mathbf{f} \notin \Omega$ where \mathcal{F} is the Fourier transform, $\mathbf{f} = (f_1, f_2)^T$ is a 2-D frequency vector and $\Omega \subset R^2$ is the spectral support of g . If the lattice Λ satisfies the multi-dimensional Nyquist criterion [19], the Shannon sampling theory allows to perfectly reconstruct g from g_Λ . However, in the case of irregular sampling the standard theory is not

applicable. Therefore, the general goal is to develop a method for the reconstruction of g from an irregular set of samples $g_\Psi = \{g(\mathbf{x}_i), \mathbf{x}_i \in \Psi \subset \mathbb{R}^2, i = 1, \dots, K\}$, where Ψ is an irregular sampling grid, such as one obtained by disparity or motion compensation.

A. POCS-based reconstruction algorithm

We use the POCS methodology [17] to reconstruct image g . This methodology involves a set theoretic formulation, i.e., finding a solution as an intersection of property sets rather than by minimization of a cost function. Each property set is constructed in such a way that it contains all solutions to the problem obeying one particular property. Therefore, an intersection of all property sets, if it exists, obeys all of the properties selected. POCS methods are iterative and typically project partial solutions onto consecutive property sets. We use the following sets [16], that are also similar to those proposed in [12]:

- A_0 - set of all images g such that at $\mathbf{x}_i \in \Psi, i = 1, \dots, K$ (irregular sampling grid) $g(\mathbf{x}_i) = g_\Psi(\mathbf{x}_i)$,
- A_1 - set of all band-limited images g , i.e., such that $G(\mathbf{f}) = 0$ for $\mathbf{f} \notin \Omega$.

Assume that the membership in A_0 can be assured by a sample replacement operator \mathcal{R} (to enforce proper image values on Ψ), and the membership in A_1 – by suitable bandwidth limitation (low-pass filtering) \mathcal{B} . Clearly, both \mathcal{R} and \mathcal{B} are projections. The iterative reconstruction algorithm can be then expressed either as

$$g^{k+1} = \mathcal{B}\mathcal{R}g^k = \mathcal{B}\left[g^k + \mathcal{S}_\Psi(g - g^k)\right], \quad (1)$$

or as

$$g^{k+1} = \mathcal{R}\mathcal{B}g^k = \mathcal{B}g^k + \mathcal{S}_\Psi(g - \mathcal{B}g^k), \quad (2)$$

where g^k is the reconstructed image after k iterations and \mathcal{S}_Ψ is a sampling operator that from a continuous-coordinate image extracts values (luminance/color) on the irregular grid Ψ (see [12] for details).

In equation (1), in each iteration first a replacement is applied followed by bandwidth limitation. Therefore, this equation, originally proposed in [12], results in an approximation rather than an interpolation of g_Ψ ; the last step is always low-pass filtering. On the other hand, equation (2) results in an interpolation since the last step in each iteration is that of sample substitution.

In order to practically implement equation (1) on a computer, a suitable discretization must be applied. In [12], equation (1) was implemented as follows:

$$g_\Lambda^{k+1} = \mathcal{B}\left[g_\Lambda^k + \lambda \mathcal{I}_{\Psi/\Lambda}(g_\Psi - \tilde{g}_\Lambda^k)\right], \quad (3)$$

where the lowpass filtering \mathcal{B} is implemented over Λ and λ is a parameter that allows control of convergence and stability of the algorithm. Clearly, g_Λ^k denotes image g reconstructed on Λ after k iterations. The symbol

\tilde{g}_Λ^k denotes a bilinearly-interpolated image g_Λ^k needed to recover image samples on Ψ . Also, note that, due to the discretization of the problem, an interpolation function $\mathcal{I}_{\Psi/\Lambda}$ replaces the sampling operator \mathcal{S}_Ψ . This function interpolates image samples $(g_\Psi - \tilde{g}_\Lambda^k)$ defined on Ψ in order to recover samples on Λ . Sauer and Allebach have studied three interpolators $\mathcal{I}_{\Psi/\Lambda}$: one derived from bilinear interpolation and two based on triangulation with planar facets [12].

The implementation (3) of the reconstruction algorithm (1) suffers from two deficiencies. First, by processing all images on Λ , i.e., the final reconstruction grid, there is little flexibility in shaping the spectrum of g_Λ^k ; any practical lowpass filtering on Λ must suppress high frequencies since a slow roll-off transition band must be used to minimize ringing at sharp luminance/color transitions (this will be discussed further in Section III-B). Secondly, the interpolation operator $\mathcal{I}_{\Psi/\Lambda}$, especially the one based on triangulation, that delivers better performance, is involved computationally.

In our earlier work [16] an alternative implementation of the algorithm (1) was proposed. Since our goal is the reconstruction of image samples obtained from disparity or motion compensation, a 1/2-, 1/4- 1/8-, or 1/16-pixel precision of motion or disparity vectors is usually sufficient. Therefore, we proposed to implement (1) on an oversampled grid matching that precision:

$$g_{\Lambda_P}^{k+1} = \mathcal{B}[g_{\Lambda_P}^k + \lambda \mathcal{S}_{\Psi/\Lambda_P}(g_{\Psi/\Lambda_P} - g_{\Lambda_P}^k)], \quad (4)$$

where the bandwidth limitation \mathcal{B} is implemented on Λ_P , that is a $P \times P$ -times denser lattice than Λ , and P typically equals 2, 4, 8 or 16 depending on disparity/motion vector precision. Clearly, Λ is a sub-grid of Λ_P , i.e., $\mathbf{x} \in \Lambda \Rightarrow \mathbf{x} \in \Lambda_P$. g_{Ψ/Λ_P} is a nearest-neighbor upsampled version of g_Ψ on Λ_P ; intensities of samples on Ψ are passed to their nearest neighbors on Λ_P , while the remaining samples on Λ_P are zeroed. Mathematically, this can be defined at each $\mathbf{x}_i \in \Psi$ as follows:

$$g_{\Psi/\Lambda_P}(\mathbf{y}) = \begin{cases} g_\Psi(\mathbf{x}_i) & \text{if } \|\mathbf{x}_i - \mathbf{y}\| \leq \|\mathbf{x}_i - \mathbf{z}\|, \\ 0 & \text{otherwise.} \end{cases}$$

for all $\mathbf{y}, \mathbf{z} \in \Lambda_P$. Similarly, $\mathcal{S}_{\Psi/\Lambda_P}$ denotes the nearest-neighbor sub-sampling, i.e., sub-sampling on Λ_P that only retains samples at $\mathbf{y} \in \Lambda_P$ nearest to $\mathbf{x}_i \in \Psi$. This sub-sampling is necessary since $g_{\Lambda_P}^k$ is supported at all positions in Λ_P , and so is the update term $(g_{\Psi/\Lambda_P} - g_{\Lambda_P}^k)$, whereas the sample substitution needs to occur at irregular positions from Ψ but quantized to Λ_P .

In other words, the implementation (4) is performed on a denser lattice Λ_P and the positions of the irregular samples from Ψ are quantized to the nearest position on Λ_P . This allows us to avoid the cumbersome interpolation $\mathcal{I}_{\Psi/\Lambda}$ under the assumption that a suitable value of P is selected. Moreover, the bilinear interpolation to recover \tilde{g}_Λ^k in (3) is not needed any more, as the error $(g_{\Psi/\Lambda_P} - g_{\Lambda_P}^k)$ is computed directly on Λ_P . This brings significant computational savings since the nearest-neighbor interpolation to compute g_{Ψ/Λ_P} is performed only once at the beginning, whereas the bilinear interpolation in (3) is needed at every iteration to compute \tilde{g}_Λ^k . In the final step, $g_{\Lambda_P}^k$ is downsampled to Λ to obtain g_Λ^k .

B. Adaptation of the relaxation coefficient

The choice of the relaxation coefficient λ in equation (4) has a direct impact on the convergence properties of the algorithm; the greater the λ , the faster the convergence, but only up to some λ_{max} above which the algorithm becomes unstable. Experiments with our original algorithm (4) have shown that the value of λ_{max} is closely related to the properties of the irregular sampling grid Ψ . Namely, the algorithm has been most prone to instability in image regions where the irregular sampling grid is most dense; when increasing λ above λ_{max} , the algorithm starts to diverge in those image regions where the number of irregular samples per unit area is the highest. Therefore, we propose here to adapt the correction term in (4) to the density of Ψ as follows:

$$g_{\Lambda_P}^{k+1} = \mathcal{B} \left[g_{\Lambda_P}^k + (\lambda/d_\Psi) \mathcal{S}_{\Psi/\Lambda_P} (g_{\Psi/\Lambda_P} - g_{\Lambda_P}^k) \right], \quad (5)$$

where d_Ψ are samples defined on Ψ of a function d describing local density of known image points. We expect that the algorithm implementations based on (5) will allow higher values of λ_{max} , and, therefore, faster convergence than those based on formulation (4). As it will turn out, due to the introduction of d_Ψ the values of λ_{max} become only marginally dependent on the degree of variation in the local densities of irregular grids Ψ .

In order to be a good descriptor of the local grid density, the function d should equal 1 where the grid Ψ is regular, should be greater than 1 in areas where Ψ is denser than Λ , and no more than 1 when converse is true. Experiments show that the actual definition of the function is not critical; various functions d seem to work almost equally well. We have chosen to compute d by counting occurrences of irregular samples in a $\Delta \times \Delta$ square centered at each point of the regular grid Λ , where Δ is the inter-pixel distance in Λ , and then by smoothing the results with a 5×5 separable lowpass filter. The resulting regularly-sampled density d permits us to compute the irregular-grid density d_Ψ by means of nearest-neighbor interpolation.

C. Implementation

The implementation of equation (5) would, in general, require more memory and be less efficient than that of equation (3), however we opt for an implementation in the frequency domain in order to reduce the computational complexity. Let

$$e^k = (g_{\Psi/\Lambda_P} - g_{\Lambda_P}^k) / d_\Psi$$

be the reconstruction error defined on Λ_P weighted by the inverse of the irregular-grid density function. Then, each iteration of the reconstruction algorithm (5) consists of the following three steps:

1. Fourier transform of the error e^k sampled on Ψ/Λ_P :

$$E^k(\mathbf{f}) = \mathcal{F} \left\{ \sum_i e^k(\mathbf{x}_i) \cdot P^2 \delta(\mathbf{x} - \mathbf{x}_i) \right\}, \quad \mathbf{x}_i \in \Psi/\Lambda_P, \quad (6)$$

where δ is the Kronecker delta. The sampling operators S_Ψ in (1-2) [12] and S_{Ψ/Λ_P} in (5) are both equivalent to the multiplication by identical cylindrical impulses at sampling positions in Ψ . The support of each impulse is sufficiently small that no two impulses overlap. Since the integral of one impulse is equal to 1, its height is inversely proportional to the square of the smallest distance between two samples in Ψ . For the grid Ψ/Λ_P , the smallest distance between irregular samples is $1/P$, and hence the sampling impulses can be represented by $P^2\delta(\mathbf{x} - \mathbf{x}_i)$ (also see [12]).

2. Update and bandwidth limitation:

$$G^{k+1}(\mathbf{f}) = \begin{cases} G^k(\mathbf{f}) + \lambda E^k(\mathbf{f}) & \mathbf{f} \in \Omega, \\ 0 & \mathbf{f} \notin \Omega, \end{cases} \quad (7)$$

with $G^0 = 0$. Note that the smaller the Ω , the fewer the samples of $E^k(\mathbf{f})$ that need to be computed. This permits a significant reduction of memory requirements and of computational complexity by means of a pruned FFT [20]. A simple bandwidth limitation by zeroing parts of the spectrum leads, in the spatial domain, to oscillations at sharp luminance/color transitions. In Section III-B, we discuss the design of a lowpass filter (frequency-domain window) that minimizes these effects.

3. Computation of g^{k+1} on Ψ/Λ_P :

$$g^{k+1}(\mathbf{x}_i) = \mathcal{F}^{-1}\{G^{k+1}(\mathbf{f})\}|_{\mathbf{x}=\mathbf{x}_i \in \Psi/\Lambda_P}. \quad (8)$$

This operation can be very efficiently implemented by computing the pruned inverse Fourier transform [20]. The final reconstruction g_Λ is computed by $P \times P$ -fold subsampling of g_{Λ_P} . Note that no additional pre-filtering is necessary since the bandwidth limitation \mathcal{B} is designed to obey the Nyquist criterion on Λ (Section III-B).

III. PROGRAMMING CONSIDERATIONS

A. Memory management

The main practical problem for methods operating on oversampled images is the amount of required memory, e.g., for $P = 8$, 64 times more intermediate memory is needed than for the final reconstructed image. On the other hand, the sampling theorem implies that the number of non-zero Fourier-domain samples for an oversampled signal does not depend on the oversampling coefficient P ; only $1/P^2$ DFT coefficients of the oversampled signal lay in the spectral support (2-D bandwidth) of the non-oversampled signal. Since in practice the operator \mathcal{B} (4) is a lowpass filter with a non-zero transition band, the number of non-zero DFT samples after filtering equals $1/P^2 \cdot m^2$ times, where m is the filter bandwidth excess (Section III-B). Therefore, for practical values of m the Fourier-domain operations require much less memory than the space-domain methods with P -fold oversampling.

Note that the zero-valued DFT coefficients need not be computed, and can be ignored at the input of the inverse DFT (step 3). In [20] it was shown that a computation of no more than K consecutive coefficients of an N -point DFT, where K is a divisor of N , requires only $O(\log K)$ operations per sample of the N -point data vector, in comparison with $O(\log N)$ operations needed for the complete transform. Let us assume that $N = K \cdot M$. We start with reorganizing the input and output DFT samples as follows:

- input samples $x(n)$, $n = 0, 1, \dots, N - 1$:

$$x(n) \longrightarrow x(n', n'') = x(n' + M \cdot n''), \quad n' = 0, 1, \dots, M - 1; n'' = 0, 1, \dots, K - 1; \quad (9)$$

- output samples $X(k)$, $k = 0, 1, \dots, N - 1$:

$$X(k) \longrightarrow X(k', k'') = X(K \cdot k' + k''), \quad k' = 0, 1, \dots, M - 1; k'' = 0, 1, \dots, K - 1. \quad (10)$$

These 1-D \rightarrow 2-D data mappings change the DFT formula as follows:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \longrightarrow X(k', k'') = \sum_{n'=0}^{M-1} \left\{ W_N^{k''n'} \sum_{n''=0}^{K-1} x(n', n'') W_K^{k'n''} \right\} W_M^{k'n'}. \quad (11)$$

Assume now that we want to compute K consecutive DFT samples starting with index $k = 0$. In such a case formula (11) simplifies to the following one:

$$X(0, k'') = \sum_{n'=0}^{M-1} W_N^{k''n'} \sum_{n''=0}^{K-1} x(n', n'') W_K^{k'n''}. \quad (12)$$

In [20] it is shown that the same formula is obtained when indices k'' are replaced by a set of more general indices $l(k'')$, for which the only condition that holds is:

$$l(k'') \bmod K = k''. \quad (13)$$

In particular, for the purpose of lowpass DFT windowing, the following set of indices should be taken (K is even):

$$l(k'') = 0, 1, \dots, K/2 - 1, K/2, N - K/2 + 1, \dots, N - 2, N - 1.$$

Clearly, $N - i \bmod K = K - i$, and for real-valued data $X(N - K/2)$ is the complex conjugate of $X(N/K)$.

The computation of equation (12) can be organized in such a manner that only two K -point vectors need to be implemented. The first vector is used for computing the inner summations on data blocks for fixed n' : $x_{n'}(n'') = x(n' + n'' \frac{N}{K})$. These are calculations of K -point DFTs on the blocks, i.e., they can be done in-place by the FFT algorithm. The FFT computations can be combined with multiplications by factors $W_N^{k''n'}$ outside the summations. The second K -point vector is used for accumulating the results of the outer summation. It is particularly important that in our application the vector $x(n', n'')$ is not remembered – only values of the irregularly-positioned samples are stored in a table of lists, one list per block in inner summation, each element

of a list containing the true value of image sample, its recently computed value, and its coordinates. Then, at any time a K -point input block is formed¹, the appropriate list is addressed, read, and irregular positions belonging to the block are filled with data.

Similarly, the inverse DFT algorithm can be pruned. Assume the same 1-D \rightarrow 2-D data mapping as in (9-10), and that only K first DFT samples are nonzero. Then, the following pruned inverse DFT formula is obtained:

$$x(n) = \frac{1}{N} \sum_{k=0}^{K-1} X(k)W_N^{-kn} \longrightarrow x(n', n'') = \frac{1}{N} W_N^{-k''n'} \sum_{k''=0}^{K-1} X(0, k'')W_K^{-k''n''}. \quad (14)$$

Note that the K -point data blocks $x_{n'}(n'')$ consisting of samples $x(n', n'')$ for a fixed n' are computed independently. Then, at any time such a block is computed², the data from irregularly-positioned samples belonging to it are processed and saved in the appropriate list, and the rest of the results is discarded.

The memory requirements for the two-dimensional version of the algorithm are as follows. Let us assume that an $M \times N$ -point vector can hold all samples of the lattice Λ (finite and regular); the oversampled image is of size $(P \cdot M) \times (P \cdot N)$. The vector sizes should be chosen in such a way that there exists an efficient $K \times L$ -point DFT algorithm, where K is a divisor of $P \cdot M$ such that $K \geq m \cdot M$ to accommodate all nonzero DFT samples after filtering, and L is a divisor of $P \cdot N$, $L \geq m \cdot N$. Moreover, as the signal filtering consists of its convolution with the filter's impulse response, a vector storing the data should be large enough to store them all (image after a convolution is somewhat larger than prior to it). Neglecting this fact in the case of DFT-domain filtering may cause the opposite image borders to influence each other, since multiplications in the DFT domain result in a circular convolution in space.

As in the 1-D case, the algorithm requires only two data vectors of the segment size, i.e., of size $K \times L$. Moreover, accumulation of results in the outer summation in (12) can be combined with the data update in (7), and hence we do not need a separate vector for keeping the frequency-domain data. Then, apart from the two vectors of size $K \times L$ we need four types of space-domain data: addresses of irregularly-positioned samples (Ψ/Λ_P), samples of the original image g_Ψ , space-domain results g^{k+1} on Ψ/Λ_P , and the function describing grid's local density d_Ψ . Usually, the amount of space-domain data is even less than four times $M \times N$. For comparison, in space-domain methods each oversampled image requires nearly $P^2 \cdot M \cdot N$ data words. At the end of the algorithm the reconstructed image is obtained by collecting regularly-positioned samples at the output of the inverse DFT.

For one sample of the oversampled image the algorithm requires $O(\log(K \cdot L))$ operations, which is comparable to the computational cost of direct filtering by a short-kernel separable FIR filter. Moreover, for processing $K \times L$ -point data segments we can use specialized DFT hardware [20].

¹Actually, a $K \times L$ -point input block, as the data are two-dimensional; K, L are divisors of the data dimensions.

²In the actual algorithm, blocks are two-dimensional.

B. Filter design

In equation (7), an update is followed with zeroing DFT samples outside of Ω . This is equivalent to “brick-wall” filtering, and results in, often objectionable, ringing at sharp luminance/color transitions in the filtered image. Additionally, since the filtering is applied iteratively, the effect is cumulative and the quality of the filters (e.g., with respect to ringing) is very important. Therefore, we need to design a filter with a narrow transition band, quickly-decaying impulse response and few parameters to adjust. A narrow transition band is needed in order to minimize detail loss in the image without introducing excessive aliasing, but also in order to stay as close as possible to the idempotent nature of \mathcal{B} (i.e., $\mathcal{B}\mathcal{B}=\mathcal{B}$); \mathcal{B} must be idempotent to be a projection. A quick-decay impulse response is needed for perceptual reasons (ringing).

For simplicity, we will use separable 2-D filters with the 1-D prototype obtained by convolving a rectangular window with a Gaussian window. This will assure a smooth transition region. Since the image under reconstruction is defined on Λ while the iterations are performed on Λ_P , the 1-D rectangular window must have the width of $N/(2P)$, with N being the DFT size³. This delimits the 2-D bandwidth at every iteration by a factor of $P \times P$ (Λ_P is $P \times P$ oversampled compared to Λ).

The Gaussian window is defined as follows:

$$w_G(n) = e^{-(2n/N_G)^2}, n = -N_G + 1, \dots, -1, 0, 1, \dots, N_G - 1,$$

under suitable normalization. Although this is a finite-support Gaussian window, the amplitude at its edges is less than 3% of $w_G(0)$ for $N_G > 8$.

Clearly, the sole design parameter to be determined is the ratio of window widths: $\mu = N_G/(N/2P)$. This ratio has a direct impact on the filter transition band since it extends between $(1 - \mu)(N/2P)$ and $(1 + \mu)(N/2P)$. The term $m = 1 + \mu$, that we shall call *filter bandwidth excess*, tells us how much filter bandwidth exceeds the Nyquist limit of $N/2P$.

IV. APPLICATION TO STEREOSCOPIC IMAGING

As discussed in the introduction, intermediate views are necessary for continuous “look-around” and for parallax adjustment in stereoscopic/multiscopic viewing. Since the camera baseline in this case is relatively small, as compared to computer vision applications, simple disparity-compensated image interpolation usually suffices [5].

Let I_L and I_R be the left and right images of a stereo pair acquired by two cameras with either epipolar or almost epipolar geometry (no or small vertical disparity, no or small keystone distortion). Assume that an intermediate (virtual) image needs to be reconstructed that could have been captured by a camera:

1. placed on a line connecting optical centers of the two existing cameras, and at the distance $0 \leq \rho \leq 1$ from the left camera ($\rho=0$ corresponds to the left camera, while $\rho=1$ corresponds to the right camera),

³Obviously, the window is duplicated for “negative” DFT indices: $N - 1, N - 2, \dots, N - N/(2P) + 1$.

2. with the optical axis parallel to the optical axes of the other two cameras.

Also, assume that disparity vectors $\mathbf{u} = [u \ v]^T$ pivoted at distance α from the left camera are known (Fig. 1); each disparity vector passes through a lattice point at α . Let this lattice be denoted Γ_α . The complete disparity field then is: $\{\mathbf{u}(\mathbf{x}), \mathbf{x} \in \Gamma_\alpha\}$.

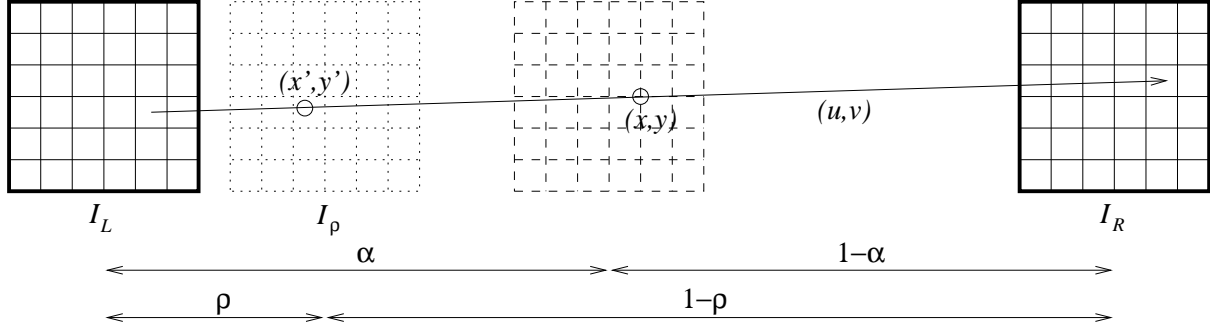


Fig. 1. Disparity compensation needed to reconstruct intermediate image at distance ρ based on disparity \mathbf{u} pivoted at distance α . The depicted disparity vector passes through sampling grid point (x, y) at distance α , while its intersection point (x', y') with the image plane at distance ρ does not, in general, belong to the sampling grid at ρ .

In this small-baseline epipolar-geometry scenario, the intermediate image I_ρ can be quite accurately reconstructed by linear disparity-compensated interpolation as follows [4]:

$$I_\rho(\mathbf{x} + (\rho - \alpha)\mathbf{u}(\mathbf{x})) = (1 - \rho)I_L(\mathbf{x} - \alpha\mathbf{u}(\mathbf{x})) + \rho I_R(\mathbf{x} + (1 - \alpha)\mathbf{u}(\mathbf{x})), \quad \mathbf{x} \in \Gamma_\alpha.$$

Note again that while the image is reconstructed at distance ρ , the disparity field is pivoted at distance α (Fig. 1).

If we require that $\rho = \alpha$, i.e., that the disparity field be pivoted in the plane of the reconstructed image, then

$$I_\rho(\mathbf{x}) = (1 - \alpha)I_L(\mathbf{x} - \alpha\mathbf{u}(\mathbf{x})) + \alpha I_R(\mathbf{x} + (1 - \alpha)\mathbf{u}(\mathbf{x})), \quad \mathbf{x} \in \Gamma_\alpha,$$

and a regularly-sampled intermediate image I_ρ results. However, disparities must be estimated for each α , thus requiring an extreme computational effort.

Alternatively, if we permit $\rho \neq \alpha$, only one disparity field needs to be computed for each stereo pair. Then, however, I_ρ is reconstructed at irregular spatial locations $\Psi = \{\mathbf{x} + (\rho - \alpha)\mathbf{u}(\mathbf{x}), \mathbf{x} \in \Gamma_\alpha\}$. In order to recover I_ρ on lattice Λ , an image reconstruction method from irregularly-spaced samples, such as the one proposed in this paper, is needed. Then, the task is: for a given α , reconstruct a regularly-sampled image $\{I_\rho(\mathbf{x}), \mathbf{x} \in \Lambda\}$ from irregular samples $\{I_\rho(\mathbf{x} + (\rho - \alpha)\mathbf{u}(\mathbf{x})), \mathbf{x} \in \Gamma_\alpha\}$.

V. EXPERIMENTAL RESULTS

The proposed reconstruction algorithm has been tested experimentally on several synthetic and natural images with various regular but non-uniform, and also with irregular sampling grids Ψ .

First, we have tested the behavior of our algorithm on two 364×364 images: *Bouquet* (natural image – Fig. 2.a) and *Zoneplate* (synthetic circular pattern with radially varying frequency – Fig. 3.a), using synthetic displacements that lead to regular but non-uniform grids Ψ . The two particular examples of regular, non-uniform grids that we have studied are those created by zoom and rotation. In order to evaluate the algorithm’s performance numerically, we have processed images twice by first applying a forward, and then – an inverse transformation. In particular, we have carried out the following:

- rotation experiment: rotation by $\pi/8$ followed by rotation by $-\pi/8$,
- zoom experiment: $1.5 \times$ zoom-in followed by $1.5 \times$ zoom-out.

In this experiment, the proposed POCS algorithm has been applied without λ adaptation and with the following parameters: $\lambda=1.0$ (fixed), $P=4$ and $m=1.25$. As the stopping criterion, we used the change in the mean-squared reconstruction error evaluated on Ψ (using bicubic interpolation). Iterations were stopped when the ratio of error change to the average error over the last two iterations was less than 0.005.

Figs. 2 and 3 show the original images along with reconstruction errors for the case of zoom. The reconstruction errors are computed as the difference between the original image and the twice interpolated image (forward and inverse transformation) over a sub-window (pixels that stay “visible” throughout the process), and suitably scaled and offset to accommodate negative error values. Note that despite a significant zoom factor, the reconstruction error is very small for both images and visible only in high-contrast areas. In the *Zoneplate* image, the error amplitude is also correlated with pattern’s local frequency as can be seen in Fig. 4; the higher the frequency, the larger the error. This is to be expected because of the lowpass filtering implemented by the projection operator \mathcal{B} .

This high quality of interpolation has been confirmed numerically; PSNR of the reconstruction error reaches values between 38dB and 42dB after 5-6 iterations, thus making the reconstructed images virtually undistinguishable from the original ones. Fig. 5 shows the evolution of the reconstruction PSNR. Note the quick convergence; already after 4 iterations the PSNR exceeds 30dB. Also, note that the larger number of iterations needed for the zoom-in phase is due to the fact that the interpolation is based on fewer image samples (due to the zoom-in) which leads to a smaller value of allowed λ ’s. Similar PSNR values have been obtained in the rotation experiment.

In order to test the proposed algorithm in a more realistic scenario, we have applied it in the context of disparity-compensated interpolation of stereoscopic views, as discussed in Section IV. First, from the stereoscopic pair *Flowerpot* at ITU-R 601 resolution (Fig. 8.a) we estimated a dense disparity field \mathbf{u} at $\alpha=0.0$ (i.e., passing through regular grid points in the left image), using an algorithm akin to optical flow computation [21], [2], [4]. Clearly, we cannot evaluate our reconstruction algorithm numerically for $0 < \rho < 1$ since no

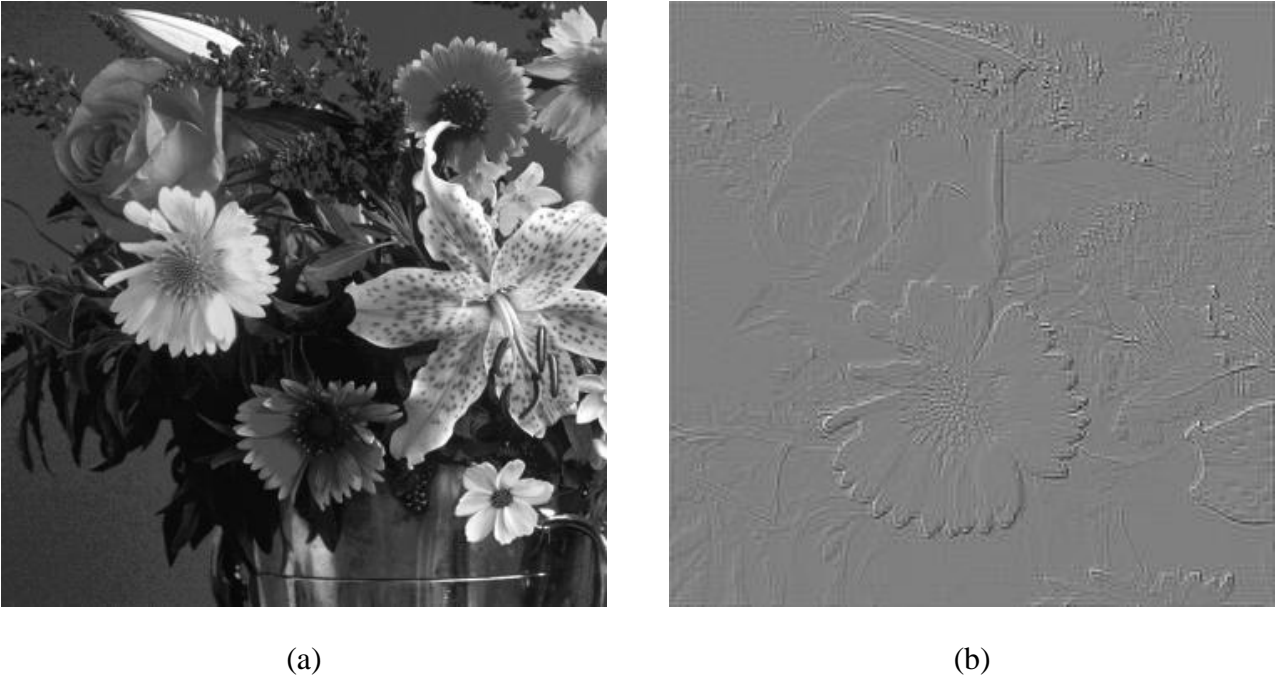


Fig. 2. (a) Original test image *Bouquet*, and (b) reconstruction error ($+128, \times 3.5$) for pixels that stay “visible” throughout the zoom-in/zoom-out (fixed $\lambda=1.0, P=4, m=1.25$).

intermediate images are available (we have a stereo pair only). Therefore, we decided to reconstruct the right image \tilde{I}_R , i.e., image for $\rho=1$, based on the disparity field \mathbf{u} computed at $\alpha=0.0$. However, to avoid mixing disparity estimation errors with image reconstruction errors, we used the estimated disparity field \mathbf{u} *only* in order to define the irregular grid $\Psi = \{\mathbf{x} + \mathbf{u}(\mathbf{x}), \mathbf{x} \in \Gamma_{0.0}\}$ at $\rho=1$. Subsequently, we computed luminance and color on Ψ at $\rho=1$ by means of *regular* \rightarrow *irregular* interpolation ($\Gamma_{1.0} \rightarrow \Psi$); we used bicubic spline interpolation [6] on I_R . We used the resulting image \tilde{I}_R defined on Ψ as the input to our algorithm (i.e., as g_Ψ).

Based on g_Ψ (or \tilde{I}_R), we reconstructed g_Λ using the proposed algorithm and we compared the result with the original image I_R . We tested the algorithm for the oversampling factor P equal to 4, 8 and 16 for different λ values. For each P we established the maximum value of the coefficient λ for which the algorithm was stable.

After some experimentation we selected two filter bandwidth excess values for further tests: $m = 1.22$, and $m = 1.75$. As it turned out, $m=1.22$ was the optimal value from the point of view of the reconstruction error evaluated on Ψ , whereas $m=1.75$ was optimal for the error evaluated on Λ . The optimum for m around 1.22 is broad, and thus the results for $m = 1.2, 1.22, \text{ or } 1.25$ are very similar. Moreover, the optimal m is almost the same for different images, sampling grids, and values of P . Fig. 6 shows PSNR plots for the luminance reconstruction error evaluated on Ψ for $\lambda = 0.7$ and $P = 4$. This error is computed as the luminance difference between the irregularly-sampled image g_Ψ and suitably interpolated g_Λ^k (bicubic

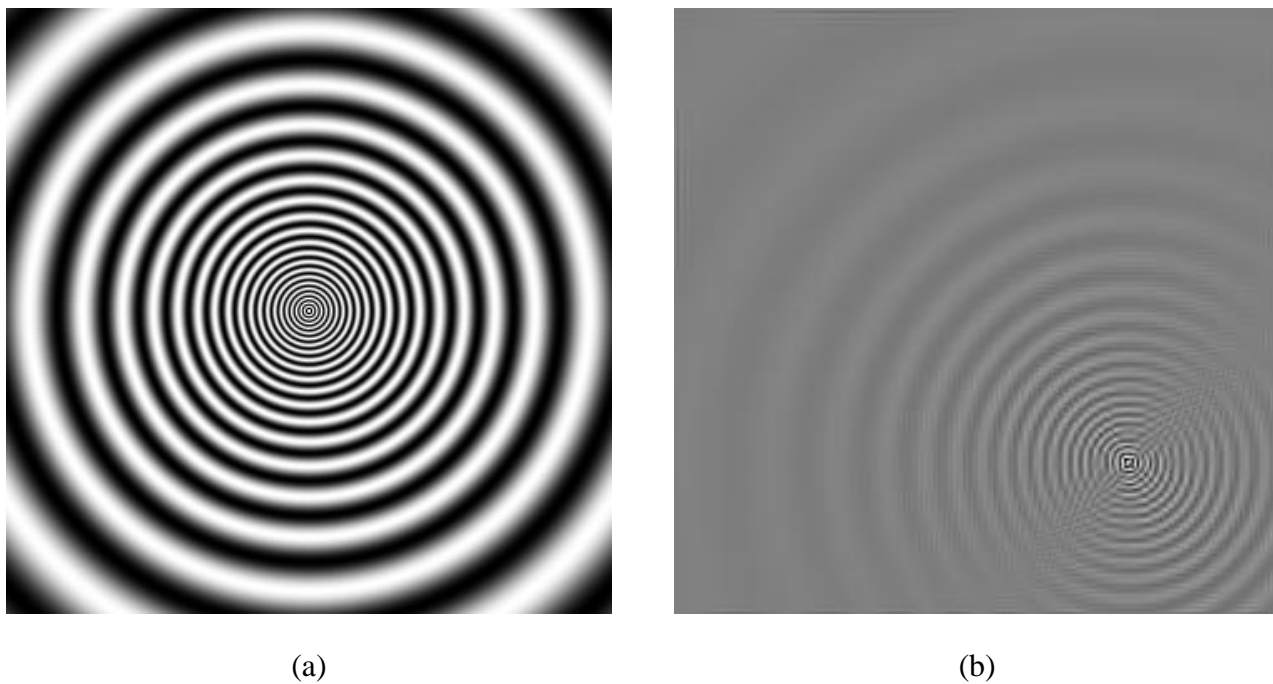


Fig. 3. (a) Original test image *Zoneplate*, and (b) reconstruction error (+128, $\times 1.5$), for pixels that stay “visible” throughout the zoom-in/zoom-out (fixed $\lambda=1.0$, $P=4$, $m=1.25$).

interpolation [6]) after each iteration. As can be seen, the reconstruction PSNR for $m = 1.22$ is clearly higher than that for $m = 1.75$. Note that in the experiments with the *Flowerpot* image iterations were stopped when the ratio of error change to the average error over the last two iterations was less than 0.01.

However, as is shown in Fig. 7.a, the converse is true when the luminance reconstruction error is evaluated on the target (regular) grid Λ ; after 5 iterations PSNR for $m=1.75$ and all P 's exceeds that for $m=1.22$. We have observed that with a growing m , the convergence, the final PSNR, and the stability of the algorithm improve up to approximately $m = 1.75$. Above this value the convergence and stability are getting even slightly better, but the final PSNR diminishes. In the initial experiments with zoom and rotation, however, values of m between 1.2 and 1.25 resulted in the lowest approximation error when evaluated both on Ψ , as well as on Λ .

We have also applied the proposed algorithm to the chrominance components of the test image *Flowerpot* (Figs. 7.b and 7.c). Interpolation in other color spaces is possible as well [22], however this would require additional color space conversions. Since the chrominance is subsampled by 2 horizontally (4:2:2 format) as compared to the luminance, disparity vectors (predominantly horizontal) are twice shorter than in the case of the luminance. This, in turn, results in less irregular Ψ and a better behavior of the algorithm for $m=1.22$. The interpolation is almost perfect; PSNR quickly surpasses 45 dB. In this case, the blurring effect of the narrowband filter with $m=1.75$ is dominant, and makes the algorithm inferior to the case of $m=1.22$ even for reconstruction errors evaluated on Λ . A more detailed discussion of this effect follows below.

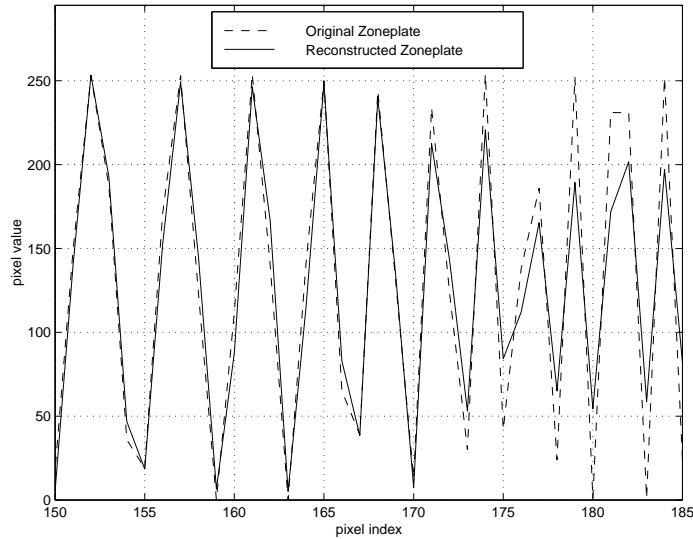


Fig. 4. Comparison of a 1-D intensity profile from the original *Zoneplate* image and from its reconstruction (Fig. 3.b). Note the increasing error for higher frequencies.

The discrepancy between reconstruction errors on Ψ and Λ prompted us to analyze the spatial distribution of this error for different m 's. It turned out that in some areas of the image the algorithm fails to reconstruct correctly image samples on the regular grid. In such areas there are high peaks in the approximation error that neither grow nor diminish as the algorithm evolves. The filter for $m = 1.75$ is a narrowband filter; its transition band starts at $0.25/P$ (and therefore its passband is quite narrow). Hence, it interpolates data in problematic areas by more smooth functions than in the case of $m = 1.22$. Outside of these areas appearance of the image for $m = 1.22$ is clearly better since, as a wideband filter (narrow transition band), it blurs images less. Of course, such "false locations" do not appear on regular non-uniform grids, e.g., grids due to image rotation or zooming, hence, wideband filters were clearly better in the experiments with *Bouquet* and *Zoneplate*. It seems that the best results would be obtained by space-variant filters, becoming narrowband in the areas where sampling grid is highly irregular, but we leave this to future work.

Fig. 8 illustrates the above considerations. It contains the original image *Flowerpot* and a 240×160 -pixel window from the original image, as well as reconstructions within this window after 14 iterations with $\lambda=0.7$ and $P=4$. At a first glance, the approximation for $m=1.22$ seems better (especially on a CRT screen); the image is sharper and has more contrast than the one for $m=1.75$, and is very close, in most areas, to the original. A second look reveals, however, small black patches that are false (on the border between the plant and the brick wall), and some distortions (on the wall at lines propagating from this border upwards). The image for $m=1.75$ is not distortion-free, but the "false locations" are much less visible. Note that the plant is much closer to the camera than the brick wall, and hence causes a discontinuity in the displacement field at the border between them. This discontinuity results in more spread out positions in Ψ (lower density) from

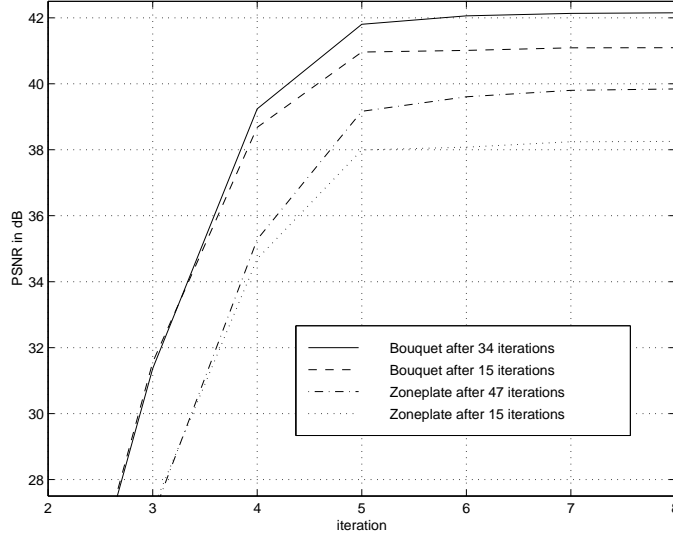


Fig. 5. Evolution of reconstruction error for test images *Bouquet* and *Zoneplate* for zoom-out (fixed $\lambda=1.0$, $P=4$, $m=1.25$). The number of iterations for zoom-in is shown in the legend.

which a better interpolation is achieved by more severe lowpass filtering (smoother interpolating functions), which is the case for $m=1.75$.

In order to further verify the impact of the irregularity of Ψ , we have repeated the above experiment for $\alpha=0.5$. Unlike in the case of $\alpha=0.0$, the disparity vectors are pivoted in this case at the midpoint between the left and right images (Fig. 1). Although the estimated disparities are not identical to the ones estimated for $\alpha=0.0$, they are very similar. Due to the non-zero α , the irregular grid at $\rho=1.0$ is now $\Psi = \{\mathbf{x} + 0.5\mathbf{u}(\mathbf{x}), \mathbf{x} = \Gamma_{0.5}\}$. Clearly, since the disparities estimated for $\alpha=0.0$ and 0.5 are almost identical, Ψ is less irregular in the case of $\alpha=0.5$ since only half of the length and angle of the vectors \mathbf{u} contribute to the (irregular) sample point layout. Therefore, one should expect an improved performance for $\alpha=0.5$ as compared to $\alpha=0.0$. This is confirmed in Fig. 9 where the reconstruction PSNR values for Y , U , and V at $\alpha=0.5$ are shown. As expected, the shorter vectors $(\rho - \alpha)\mathbf{u}$ contributing to Ψ result in less irregularity and a significantly higher PSNR in both luminance and chrominance reconstruction.

In Fig. 10, we show the impact of parameter λ on the convergence of the adaptive algorithm. Note that with the increasing λ , up to a certain value, the convergence speed improves, but for excessive λ values the PSNR may actually deteriorate after certain number of iterations. This demonstrates that even in the adaptive algorithm the value of λ needs to be selected judiciously.

Finally, we have compared the algorithm described here based on the adaptation of the relaxation coefficient λ with our earlier non-adaptive algorithm [16]. Fig. 11 shows PSNR evolution of the luminance reconstruction error for the non-adaptive algorithm. Note that, compared to the adaptive results from Fig. 7.a, in the non-adaptive case the maximum λ was selected experimentally for each P in order to perform the com-

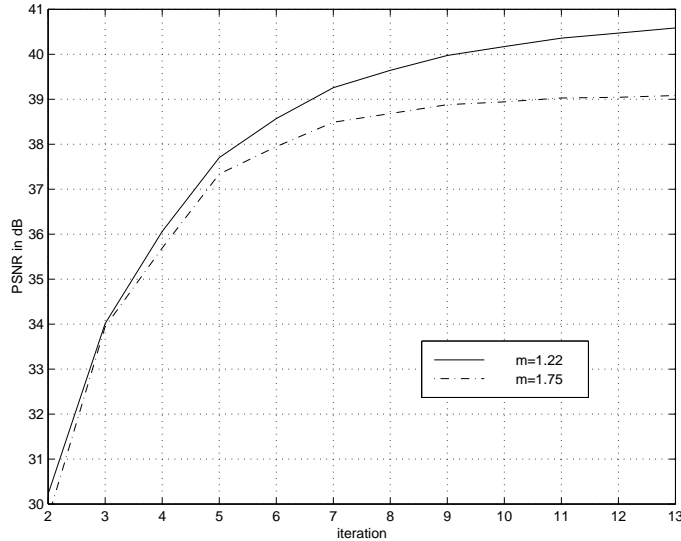


Fig. 6. Evolution of luminance reconstruction error for *Flowerpot* at $\alpha=0.0$ evaluated on Ψ (adaptive $\lambda=0.7$, $P=4$).

parison in the best-case scenario for fixed λ . Clearly, the adaptive method outperforms the non-adaptive one by about 1dB.

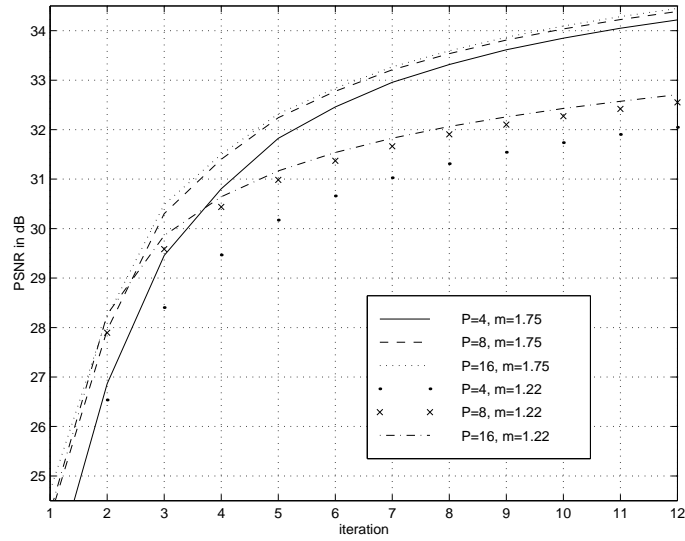
To give the reader an idea of the computational complexity of the proposed method, we have measured the execution times of a *Matlab* implementation on 466 MHz Celeron PC with 256 MB of RAM. On the *Flowerpot* images (720×240 pixels), for $m=1.22$ the execution times per iteration were: 38s, 149s, 611s for $P=4$, 8, and 16, respectively while for $m=1.75$ they were: 42s, 164s, 661s. The slight increase in CPU time for higher m (increased bandwidth excess) is due to larger processing blocks in order to account for more non-zero DFT samples, as discussed in Section III-A. Note the linear scaling of the time with $P \times P$.

VI. CONCLUSION

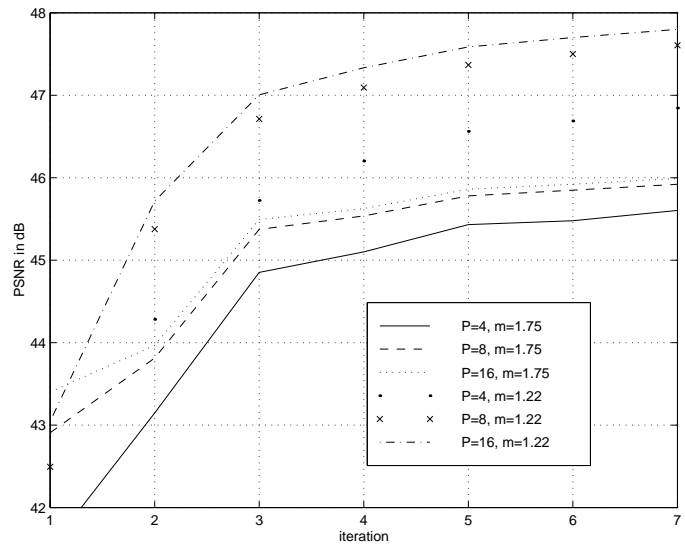
We have presented our earlier POCS-based image reconstruction method, and we have proposed two extensions: adaptation to local density of image samples and incorporation of image chrominances. We have also presented a memory-efficient implementation of the algorithm. We have tested the improved algorithm on various regular but non-uniform as well as irregular sampling grids. The presented results demonstrate very good performance of our improved algorithm, with the attained PSNR values around 40dB.

ACKNOWLEDGMENTS

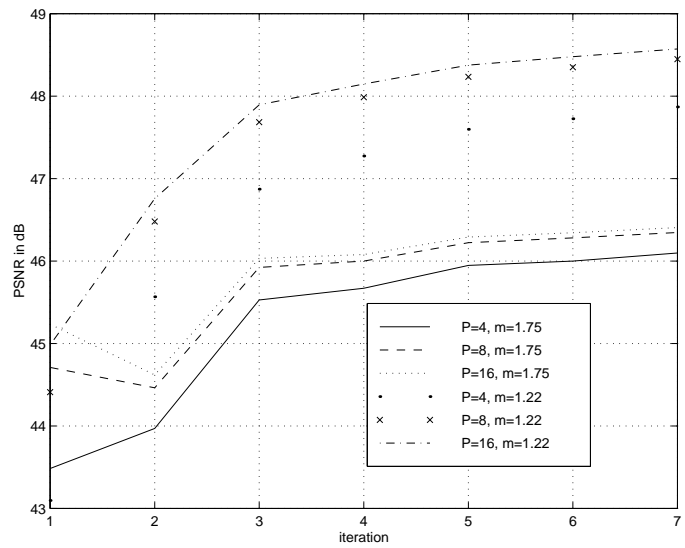
We would like to thank NHK, Japan for providing us with the stereoscopic sequences used in this work.



(a)



(b)



(b)

Fig. 7. Evolution of reconstruction error for *Flowerpot* at $\alpha=0.0$ evaluated on Λ (adaptive $\lambda=0.7$): (a) luminance Y ; (b) chrominance U ; and (c) chrominance V .

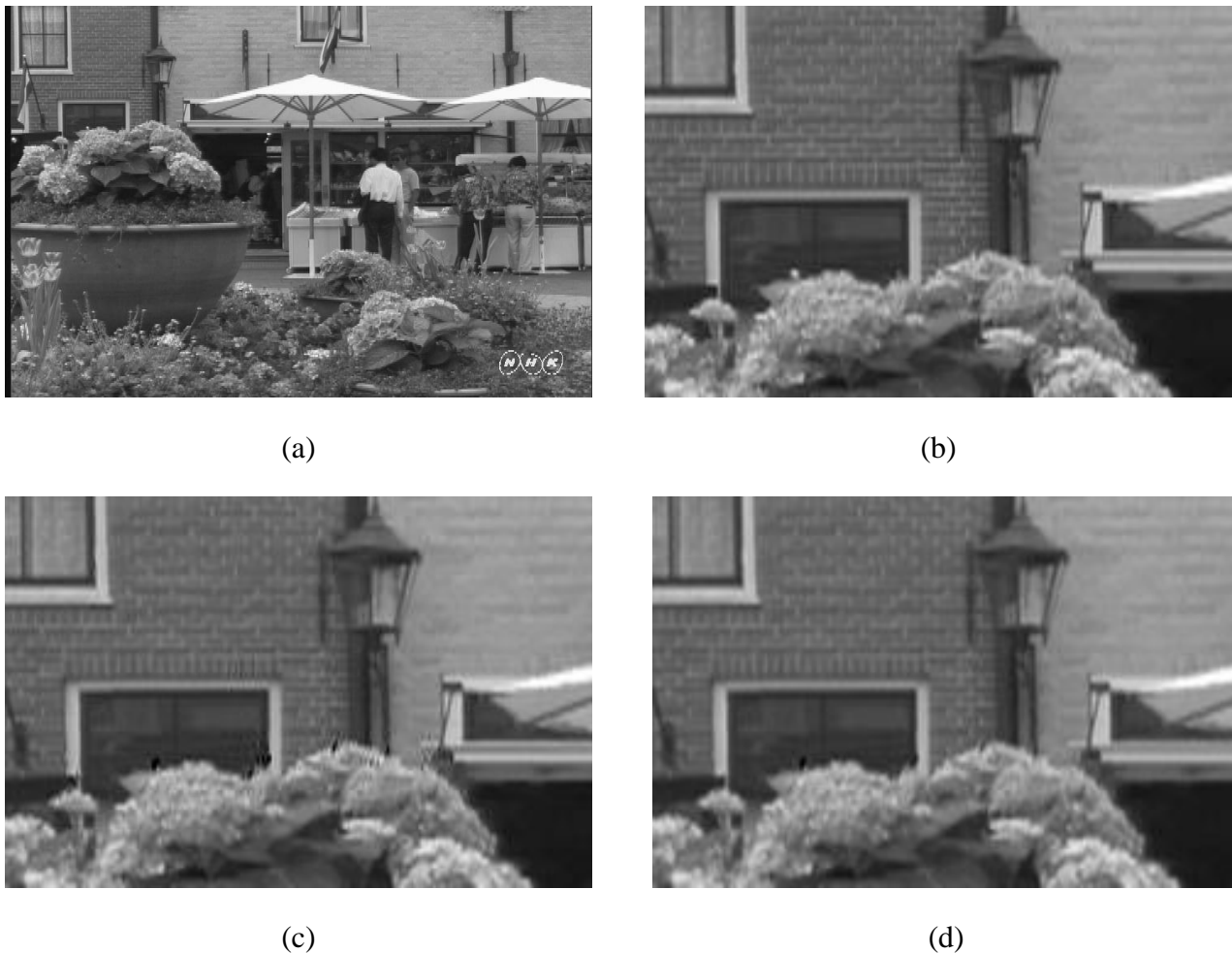
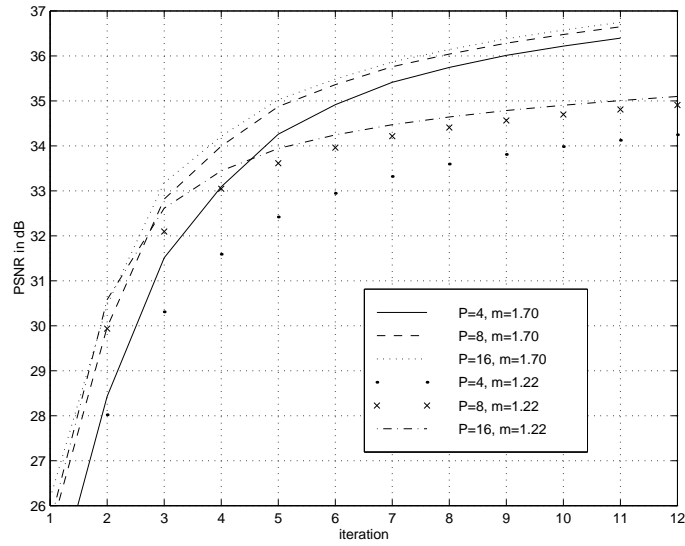


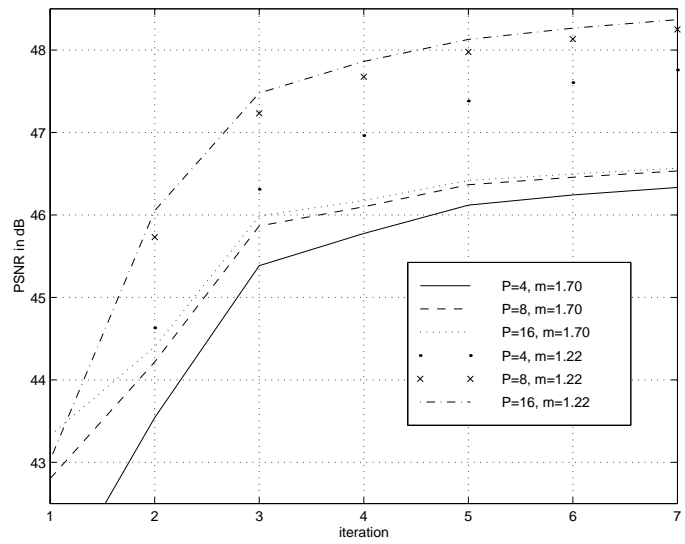
Fig. 8. (a) Original right image from the stereopair *Flowerpot* (full resolution: 720×240), and 240×160 windows: (b) with detail from the top-left part of the original right image, and with corresponding reconstructions using $\lambda=0.7$, $P=4$ after 14 iterations for (b) $m=1.22$, and for (c) $m=1.75$.

REFERENCES

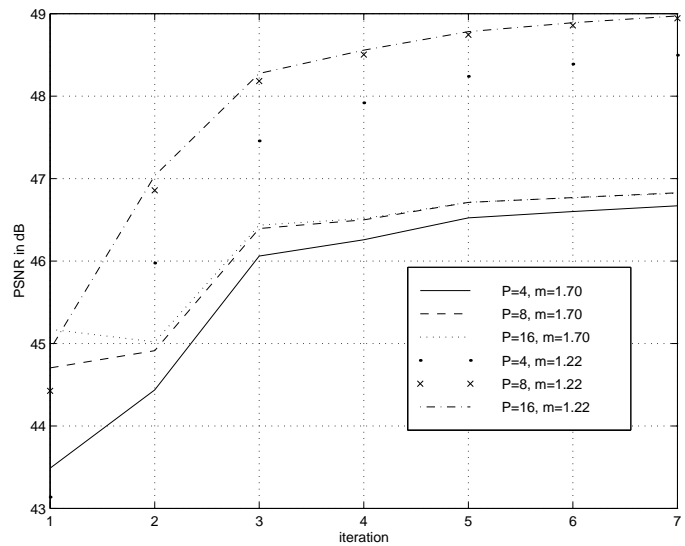
- [1] N.L. Chang and A. Zakhor, "View generation for three-dimensional scenes from video sequences," *IEEE Trans. Image Process.*, vol. 6, no. 4, pp. 584–598, Apr. 1997.
- [2] A. Mancini and J. Konrad, "Robust quadtree-based disparity estimation for the reconstruction of intermediate stereoscopic images," in *Proc. SPIE Stereoscopic Displays and Virtual Reality Systems*, Jan. 1998, vol. 3295, pp. 53–64.
- [3] A.-R. Mansouri and J. Konrad, "Bayesian winner-take-all reconstruction of intermediate views from stereoscopic images," *IEEE Trans. Image Process.*, vol. 9, no. 10, pp. 1710–1722, Oct. 2000.
- [4] J. Konrad, "Enhancement of viewer comfort in stereoscopic viewing: parallax adjustment," in *Proc. SPIE Stereoscopic Displays and Virtual Reality Systems*, Jan. 1999, vol. 3639, pp. 179–190.
- [5] J. Konrad, "View reconstruction for 3-D video entertainment: issues, algorithms and applications," in *Proc. Int. Conf. on Image Process. and its Applications*, July 1999, pp. 8–12.
- [6] R.G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 29, no. 6, pp. 1153–1160, Dec. 1981.



(a)



(b)



(c)

Fig. 9. Evolution of reconstruction error for *Flowerpot* at $\alpha=0.5$ evaluated on Λ (adaptive $\lambda=0.7$): (a) luminance Y ; (b) chrominance U , and (c) chrominance V . The PSNR plots for the luminance and $m=1.70$ stop after iteration 11 since the algorithm reached convergence.

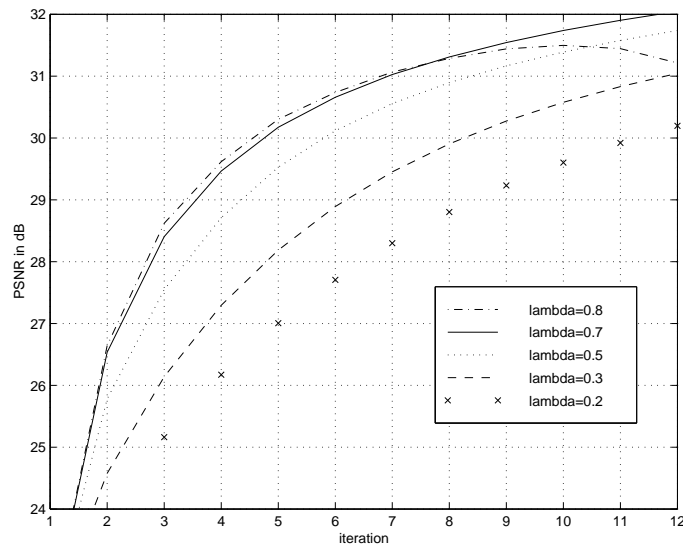


Fig. 10. Evolution of luminance reconstruction error for *Flowerpot* at $\alpha=0.0$, and with $P=4$, $m=1.22$, for different values of λ .

- [7] John J. Benedetto, "Frame decompositions, sampling and uncertainty principle inequalities," in *Wavelets: Mathematics and Applications*, J. Benedetto and M. Frazier, Eds., chapter 7, pp. 247–304. CRC Press, Boca Raton FL, 1993.
- [8] H. Feichtinger and K. Gröchenig, "Theory and practice of irregular sampling," in *Wavelets: Mathematics and Applications*, J. Benedetto and M. Frazier, Eds., chapter 8, pp. 305–363. CRC Press, Boca Raton FL, 1993.
- [9] Farokh A. Marvasti, Chaunde Liu, and Gil Adams, "Analysis and recovery of multidimensional signals from irregular samples using nonlinear and iterative techniques," *Signal Process.*, vol. 36, pp. 13–30, 1994.
- [10] Wen Chen, Shuichi Itoh, and Junji Shiki, "Irregular sampling theorems for wavelet subspaces," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 1131–1142, May 1998.
- [11] A. Aldroubi and K. Gröchenig, "Non-uniform sampling and reconstruction in shift-invariant spaces," *SIAM Review*, vol. 43, no. 4, pp. 585–620, Oct. 2001.
- [12] K.D. Sauer and J.P. Allebach, "Iterative reconstruction of band-limited images from nonuniformly spaced samples," *IEEE Trans. Circuits Syst.*, vol. 34, no. 12, pp. 1497–1506, Dec. 1987.
- [13] A. Sharaf and F. Marvasti, "Motion compensation using spatial transformations with forward mapping," *Signal Process., Image Commun.*, vol. 14, pp. 209–227, 1999.
- [14] C. Vázquez, J. Konrad, and E. Dubois, "Wavelet-based reconstruction of irregularly sampled images: Application to stereo imaging," in *Proc. IEEE Int. Conf. Image Processing*, Sept. 2000, vol. II, pp. 319–322.
- [15] C. Vázquez, E. Dubois, and J. Konrad, "Reconstruction of irregularly-sampled images by regularization in spline spaces," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2002 (to appear).
- [16] R. Stasiński and J. Konrad, "POCS-based image reconstruction from irregularly-spaced samples," in *Proc. IEEE Int. Conf. Image Processing*, Sept. 2000, vol. II, pp. 315–318.
- [17] P. Combettes, "The foundations of set theoretic estimation," *Proc. IEEE*, vol. 81, no. 2, pp. 182–208, Feb. 1993.
- [18] H. Chen, M. Civanlar, and B. Haskell, "A block transform coder for arbitrarily-shaped image segments," in *Proc. IEEE Int. Conf. Image Processing*, Nov. 1994, pp. 85–89.

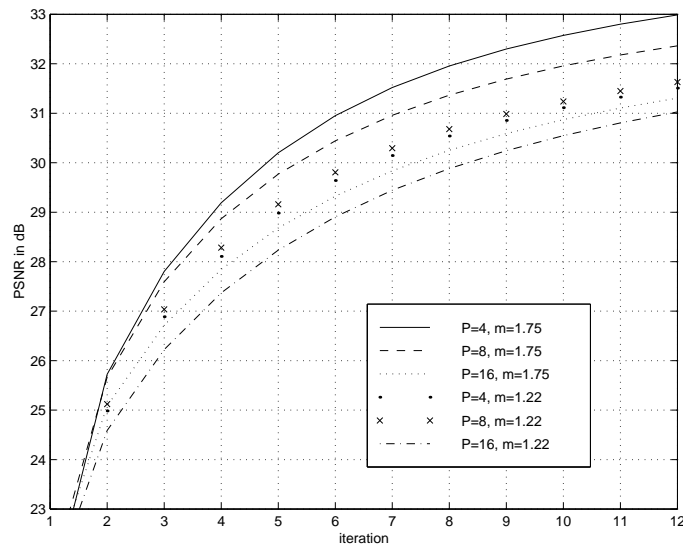


Fig. 11. Evolution of luminance reconstruction error for *Flowerpot* at $\alpha=0.0$ for the non-adaptive algorithm (fixed λ equal to 0.4, 0.3 and 0.2 for $P=4$, 8 and 16, respectively).

- [19] E. Dubois, "The sampling and reconstruction of time-varying imagery with application in video systems," *Proc. IEEE*, vol. 73, no. 4, pp. 502–522, Apr. 1985.
- [20] R. Stasiński, "FFT pruning - a new approach," in *Signal Process. III: Theories and Applications (Proc. Third European Signal Process. Conf.)*, 1986, pp. 267–270.
- [21] R. March, "Computation of stereo disparity using regularization," *Pattern Recognit. Lett.*, vol. 8, pp. 181–187, Oct. 1988.
- [22] G. Sharma and H.J. Trussell, "Digital color imaging," *IEEE Trans. Image Process.*, vol. 6, no. 7, pp. 901–932, July 1997.