

Image Coding Using Depth Blurring for Aesthetically Acceptable Distortion

Timothy Popkin, Andrea Cavallaro and David Hands

Abstract—We introduce the concept of depth-based blurring to achieve an aesthetically acceptable distortion when reducing the bitrate in image coding. The proposed depth-based blurring is a prefiltering that reduces high frequency components by mimicking the limited depth of field effect that occurs in cameras. To cope with the challenge of avoiding intensity leakage at the boundaries of objects when blurring at different depth levels, we introduce a selective blurring algorithm that simulates occlusion effects as occur in natural blurring. The proposed algorithm can handle any number of blurring and occlusion levels. Subjective experiments show that the proposed algorithm outperforms foveation filtering, which is the dominant approach for bitrate reduction by space-variant prefiltering.

Index Terms—Image coding, depth of field, foveated image coding, space-variant image processing.

I. INTRODUCTION

FOVEATED [1] and region-of-interest [2] coding techniques employ a spatially non-uniform bitrate, resolution or quality across an image or video frame. These techniques, referred to here as space-variant coding techniques, exploit the spatial variation in visual importance (saliency) of regions across the scene and the eccentricity-dependent resolution of the human eye. Foveated coding works by applying an eccentricity-dependent resolution or quality (e.g., using space-variant filtering [3]), aiming for the distortion of the encoding to be perceptually minimal or invisible [4], [5] by matching the density of photoreceptor cells, which is highest at the focal center [6].

As foveated coding requires knowledge of points or regions of interest, whether from eye tracking (gaze-contingent) [7] or estimated (e.g., saliency detection [8]), this might be problematic, as the selection of these priority regions remains an open problem [8]. If the estimated points of interest are wrong, the distortion becomes noticeable. For this reason, we aim at a different space-variant coding approach that,

unlike foveated coding, does not aim for perceptually minimal or invisible distortion, but for distortion of a style that is visually tolerable even when noticed. In other words, we aim for *aesthetically acceptable distortion*, by emulating natural depth of field effects which viewers are accustomed to as a photographic style which they associate with quality, and using this as a prefiltering stage of an image encoder. An important idea here is that if viewers look away from the predicted point of interest (as is possible when eye tracking is not employed), they might judge the degradation more favourably than with foveation filtering. This approach, referred to herein as *depth blurred coding*, has the double benefit of not only providing what might aesthetically be regarded as an enhancement, but also of allowing bitrate reduction due to the increased average blur level (degree of blur) across the view [9]. Moreover, in contrast with the difficulty of estimating human fixation points (for foveation), the proposed approach uses depth information, which is becoming increasingly available, thus allowing it to be employed in a number of possible scenarios. For example, a stereo camera pair can be used in conjunction with a stereo-correspondence technique [10] to extract a map of disparity (i.e., reciprocal of depth). A depth map can also be estimated from scene structure estimation techniques [11], [12] or using depth from motion. Most significantly, a low-resolution time-of-flight camera might be fixed to a high-resolution normal camera, which may be used to enhance the resolution of the depth map [13]. The use of time-of-flight cameras is becoming increasingly widespread [14]. Furthermore, in the simplest case, depth blurring can take a region-of-interest approach, by applying a two-level depth map (Fig. 1) generated for example by background-foreground segmentation [15] or by an object detector such as face detector [16].

In this paper, we propose a novel method[†] for applying realistic depth of field effects to images that simulates occlusion effects as occur at the boundaries of objects. The proposed approach can handle a depth map which is continuous (up to blur level quantization granularity), with a cost of order $\mathcal{O}(N \log(N)^2)$ for an N -pixel image. We also introduce a method-of-adjustment [17] approach for measuring perceived overall image quality in terms of *equivalent JPEG quality*. We evaluate the proposed depth blurring algorithm, providing subjective evidence to support our hypothesis that depth-blurred coding outperforms the dominant approach to space-variant coding, namely foveated coding, for equivalent blur levels and bitrates. We argue that the relative ease of obtaining a depth

Manuscript received April 9, 2010; revised March 9, 2011. This work was supported in part by the U.K. Engineering and Physical Sciences Research Council and in part by BT Innovate & Design. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Lina Karam.

Copyright (c) 2011 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

T. Popkin and A. Cavallaro are with the School of Electronic Engineering & Computer Science, Queen Mary University of London, E1 4NS, UK (e-mail: t.popkin@qmul.ac.uk; andrea.cavallaro@elec.qmul.ac.uk).

D. Hands is with BT Innovate & Design, Martlesham Heath, IP5 3RE, UK (e-mail: david.2.hands@bt.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIP.2011.??????

[†]The working code of the proposed method is available for download at ftp://motinas.elec.qmul.ac.uk/pub/code/tim/db_download.zip and in conjunction with this paper at <http://ieeexplore.ieee.org>.



Fig. 1. Depth/disparity map examples. Left: *Art* disparity map, from Middlebury dataset [22], [23]; right: two-level segmentation of *Foreman*.

map, when compared with the difficulty of predicting human fixation to a sufficient level of certainty for foveation, makes depth-blurring a preferable space-variant coding approach for the non-gaze-contingent scenario.

The remainder of this paper is organized as follows: Section II presents the proposed depth blurring approach. Section III describes the methodology used for subjective evaluation. Section IV discusses the results. Finally, Section V concludes the paper.

II. DEPTH BLURRING

A. Challenges in synthetic depth blurring

In this subsection, we discuss the two main challenges in depth blurring, namely the proper treatment of occlusion effects and the computational complexity.

Techniques for synthesizing depth of field can be classed as either multipass approaches or postfiltering. In multipass approaches, high-accuracy techniques such as ray tracing are repeated from slightly different directions and averaged [18]. Although high quality, multipass approaches generally involve heavy computational cost. In postfiltering, the rendering output itself is subjected retrospectively to synthetic depth blurring [19]. Postfiltering approaches can in turn be grouped into *gather* or *scatter* methods. Techniques which employ the *gather* method approximate depth blurring by taking the local average of pixel values around the desired location. This inherently leads to *intensity leaks* [19] as the intensity from sharp source pixels is spread over surrounding background that they should not influence. Approaches that employ the *scatter* method spread the intensity of each source pixel over an area. However, due to speed, scatter methods are not regarded as the choice for real-time depth blurring [20].

Depth blurring has been used in a coding scenario, employing eye tracking in combination with foveation [21]. This work relied on the eye tracking and did not test the plausibility of removing the eccentricity-dependent foveation aspect altogether along with any assumptions about where the viewer will look. It employs Gaussian pyramid blurring for resolution-reduction purposes, aiming for minimally perceivable distortion rather than photorealistic blurring that is aesthetically acceptable on close inspection as proposed herein.

One key aspect of the style of depth blurring that a human viewer is accustomed to seeing in photographs is the effect that occurs around the boundaries of objects that occlude further-away objects in the scene. This *occlusive* aspect of the blurring is necessary because, for example, when a sharply-focussed object is in front of a blurred distant object, the blur of the distant object stops abruptly at the edge of the

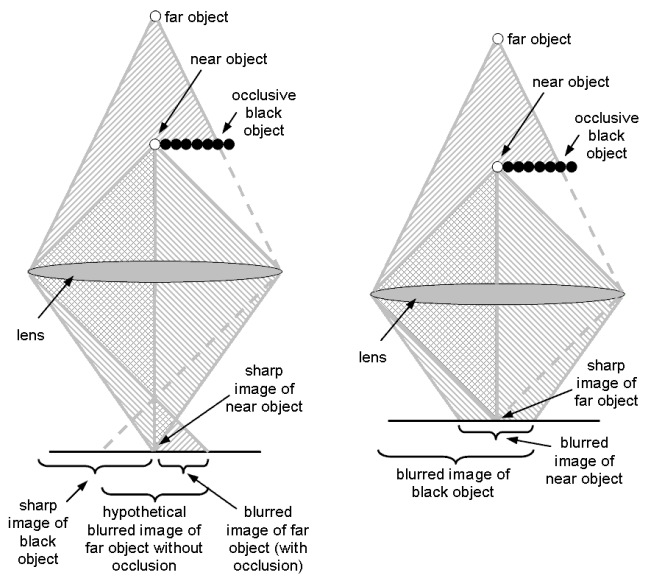


Fig. 2. Examples of occlusive effects in depth blurring. Left: near object in focus; right: far object in focus. In the left diagram, the light from the far (blurred) object is spread over a region which is occluded sharply at the image of the near object. In the right diagram, the light from the near (blurred) object is unaffected by anything beyond it.

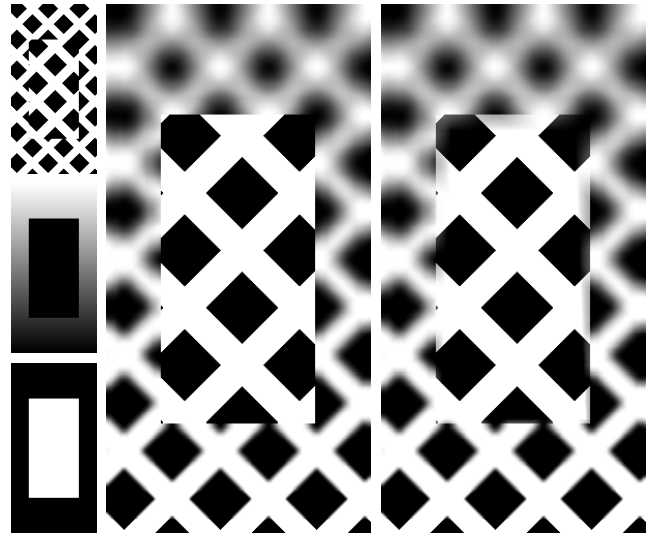


Fig. 3. Examples of occlusion effects using the proposed algorithm on a synthetic image. Left, top to bottom: raw image, blur map (black: unblurred) and occlusion map (white: more occlusive); the blur map and occlusion map may come directly from a depth map. Middle: blurred with occlusion. Right: the same blurring except with occlusion effects switched off. The spread of the background blur over the foreground boundary can be seen in the non-occlusive case, but not in the occlusive case.

nearer object, with no part of the blur overlapping any part of the nearer object. However, when a blurred object is in front of some sharply-focussed background, the edges of the blur of the nearer object spread over the background. This is because the blur goes in all directions and some of the blur will overlap the background (Fig. 2). Our proposed approach caters for this occlusive effect, taking occlusion information from an occlusion map in addition to the blur map. An ordinary, unocclusive selective blurring technique would cause every blurred pixel to be spread over its neighbors regardless of

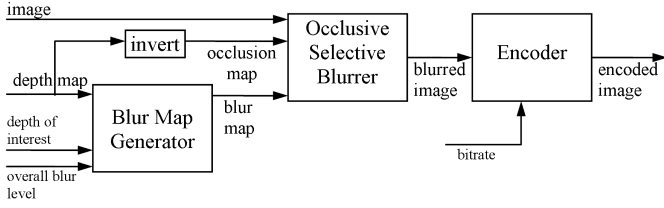


Fig. 4. Block diagram of the overall approach that receives as input a depth map (or equivalent information) and a chosen depth of interest. The Occlusive Selective Blurring block is the core algorithm, as described in section II-B.

whether they are considered nearer to or further from the camera. Examples of occlusive and unocclusive blurring are shown in Fig. 3, where the spread of the background blur over the foreground boundary can be seen in the non-occlusive case, but not in the occlusive case.

B. The blurring algorithm

Fig. 4 summarizes the proposed approach. Given a depth map (generated for example by a time-of-flight or stereo camera, or by a segmentation algorithm), a desired bitrate and overall blur level (e.g. from the rate-control mechanism of an encoder) and a depth of interest chosen to be in sharp focus (such as by taking the nearest-to-camera depth or the depth at a point of interest selected using saliency detection), the proposed depth blurring algorithm takes as inputs a color image \mathbf{C} , a continuously-varying (to one pixel-width quantization granularity) blur map B and an *occlusion map* Ω , all defined over a $W \times H$ image domain $D = \{(x, y) : x \in \{1, \dots, W\}, y \in \{1, \dots, H\}\}$.

The occlusion map gives, in arbitrary units, the occlusion level of each point, thereby providing a ranking of which pixels should or should not be overlapped by the blur regions of which other pixels. Therefore it may be taken directly as the negative of the depth map, so that more occlusive (nearer-to-camera) points have a higher occlusion level. The blur map is taken from the depth map such that the chosen depth will be in sharp focus (zero blur) and the other depths will have gradually increasing blur away from this depth; e.g., each blur level $b(\mathbf{x})$ (at location $\mathbf{x} \in D$) may be taken as

$$b(\mathbf{x}) = k \left| \frac{1}{d(\mathbf{x})} - \frac{1}{d_0} \right| \quad (1)$$

for depth $d(\mathbf{x})$, sharp-focus depth d_0 and constant k chosen to obtain a desired overall blur level. In practice, we obtain each $B(\mathbf{x})$ by rounding $b(\mathbf{x})$ to the nearest integer to avoid the subpixel interpolation of \mathbf{C} that would otherwise be necessary.

Note that here, disparity values ($1/d(\mathbf{x})$) from a stereo-correspondence technique can be used directly and that the accuracy of each blur level will be proportional to the accuracy of the disparity value. Therefore, although stereo-correspondence techniques have difficulty with distinguishing far-away depths due to parallax, this is not an issue for blur map generation as these far-away depths have very similar blur levels. The depth-blurred coding approach can alternatively be adapted to an ordinary single-camera scenario by the use of single-image

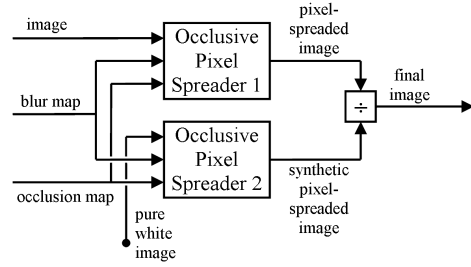


Fig. 5. Block diagram of the top level of the proposed occlusive selective blurring algorithm. For simplicity, the “pixel spreading” is shown as being applied to the whole color image, whereas in reality it is applied separately to each of the three color planes (R,G,B).

depth segmentation, or depth-from-motion techniques if video is available.

The proposed algorithm has an $\mathcal{O}(N \log(N)^2)$ cost and it can handle a depth map which is continuous, up to blur level quantization granularity. We spread the intensities of each pixel uniformly over a square area of variable size, subject to sharp occlusions by any nearer pixel. Then, we apply an adjustment factor to compensate for the fact that this pixel spreading will unnaturally darken or brighten the image in regions where the blur level is not constant. For $n=1,2,3$, each occlusively selectively blurred color plane C'_n is computed (using adjustment factor $1/U'(\mathbf{x})$) as follows:

$$C'_n(\mathbf{x}) = P_n(\mathbf{x})/U'(\mathbf{x}), \quad (2)$$

where P_n is the occlusively pixel-spread version of original color plane C_n under the given blur map and occlusion map, and U' is the equivalent when applied to a pure white image, U . The terms *pixel-spreading* and *blurring* will be used hereonwards to describe the creation of the unadjusted result and the final, adjusted result, respectively. The top level (the “blurring” stage) of the proposed algorithm is illustrated in Fig. 5.

A key aspect of the proposed algorithm is the concept of a *corner* of the spread of a given pixel under the blurring. Given a color plane G , and a blur map B , the intensity $G(\mathbf{x})$ at location \mathbf{x} will, neglecting image boundary issues, be spread over a square area of width $2B(\mathbf{x}) + 1$, with center \mathbf{x} .

Considering the pixel at $\mathbf{x} = (x, y)$, the image, $\mathcal{P}_{\mathbf{x}}$, of the spread of this sole pixel will be given, for every image location $\mathbf{x}' = (x', y')$, by

$$\mathcal{P}_{\mathbf{x}}(x', y') = \begin{cases} v & \text{if } |x' - x| \leq B(\mathbf{x}) \\ & \text{and } |y' - y| \leq B(\mathbf{x}) \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where $v = \frac{G(\mathbf{x})}{(2B(\mathbf{x})+1)^2}$. $\mathcal{P}_{\mathbf{x}}$ can also be expressed as a cumulative sum, as follows:

$$\mathcal{P}_{\mathbf{x}}(x', y') = \sum_{\substack{x'' \leq x' \\ y'' \leq y'}} \mathcal{P}'_{\mathbf{x}}(x'', y''), \quad (4)$$

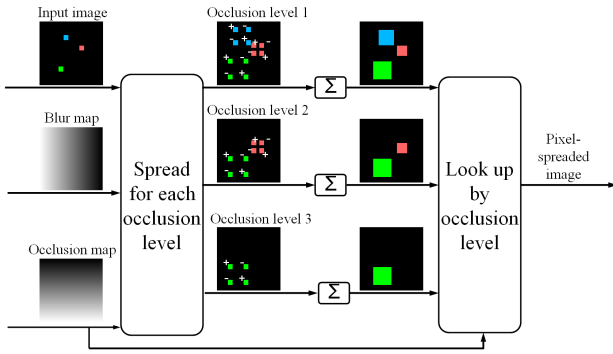


Fig. 6. Block diagram illustrating the concept of corners in the pixel-spreading. For each occlusion level, there is conceptually a differential image (second column from left), whose cumulative sum gives a pixel-spread image (third column from left), and from these, the overall occlusively pixel-spread image can be formed by looking up each output pixel from the appropriate cumulative sum according to its individual occlusion level. However, in practice, separate full-image cumulative summations (denoted above by “ Σ ”) are not performed, as the occlusive sum look-up structure allows the selected output pixels to be computed on their own, without full cumulative sums.

for all $\mathbf{x}' = (x', y')$, where image $\mathcal{P}'_{\mathbf{x}}$ is defined as follows:

$$\mathcal{P}'_{\mathbf{x}}(\mathbf{x}'') = \begin{cases} v & \text{if } \mathbf{x}'' = \mathbf{c}^{11}(\mathbf{x}, B(\mathbf{x})) \\ -v & \text{if } \mathbf{x}'' = \mathbf{c}^{12}(\mathbf{x}, B(\mathbf{x})) \\ -v & \text{if } \mathbf{x}'' = \mathbf{c}^{21}(\mathbf{x}, B(\mathbf{x})) \\ v & \text{if } \mathbf{x}'' = \mathbf{c}^{22}(\mathbf{x}, B(\mathbf{x})) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

for all possible \mathbf{x}'' . Functions \mathbf{c}^{11} , \mathbf{c}^{12} , \mathbf{c}^{21} and \mathbf{c}^{22} can be thought of as giving the four corners of the spread of the pixel at \mathbf{x} . The horizontal components, c_1^{11} , c_1^{12} , c_1^{21} and c_1^{22} , and vertical components, c_2^{11} , c_2^{12} , c_2^{21} and c_2^{22} , of these functions are defined as

$$c_1^{11}(\mathbf{x}, b) = c_1^{12}(\mathbf{x}, b) = \begin{cases} x - b & \text{if } x > b \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

$$c_2^{11}(\mathbf{x}, b) = c_2^{21}(\mathbf{x}, b) = \begin{cases} y - b & \text{if } y > b \\ 1 & \text{otherwise} \end{cases} \quad (7)$$

$$c_1^{22}(\mathbf{x}, b) = c_1^{21}(\mathbf{x}, b) = x + b + 1 \quad (8)$$

$$c_2^{22}(\mathbf{x}, b) = c_2^{12}(\mathbf{x}, b) = y + b + 1 \quad (9)$$

for every possible location $\mathbf{x} = (x, y)$ and blur level b .

The overall pixel-spread color plane, P , as produced by the *Occlusive Pixel Spreader*, may be defined as

$$P(\mathbf{x}') = \sum_{\substack{\mathbf{x} \in D: \\ \Omega(\mathbf{x}) \geq \Omega(\mathbf{x}')}} \mathcal{P}_{\mathbf{x}}(\mathbf{x}') \quad (10)$$

$$= \sum_{\substack{x'' \leq x' \\ y'' \leq y'}} \sum_{\substack{\mathbf{x} \in D: \\ \Omega(\mathbf{x}) \geq \Omega(\mathbf{x}')}} \mathcal{P}'_{\mathbf{x}}(\mathbf{x}''), \quad (11)$$

for all $\mathbf{x}' = (x', y') \in D$, where D is the set of image locations, as before, and $\mathcal{P}'_{\mathbf{x}}(\mathbf{x}'')$ is as defined in Eq. 5. This selective cumulative sum is conceptually represented in Fig. 6, which illustrates the meaning of the corners of a spread and how they are used.

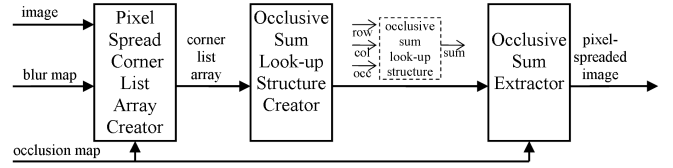


Fig. 7. Block diagram of the Occlusive Pixel Spreader (see Eq. 10). The blocks correspond to the three main blocks of the *spread* operation. The leftmost block creates a two-dimensional array of the corners of the spread of each pixel (see Eqs (6)-(9) and Fig. 6). The workings of the middle and rightmost blocks are given by the *createstruct* and *extractsum* operations of Alg. 1.

Algorithm 1: Handling occlusive sum look-up structures

Operation $S = \text{createstruct}(L)$

Input: 2-D array L of corner lists.

```

1: for each  $m \in \{0, \dots, \lfloor \log_2(H) \rfloor\}$  do
2:   Set  $h = 2^m$ .
3:   for each  $Y \in \{1, \dots, \lfloor H/h \rfloor\}$  do
4:     Set  $T = \text{null}$ .
5:     for each  $x \in \{1, \dots, W\}$  do
6:       for each  $y \in \{Yh - (h-1), \dots, Yh\}$  do
7:         for each pair  $(v, \omega) \in L(x, y)$  do
8:           Set  $T = \text{treeadd}(T, v, \omega)$ .
9:         end for
10:      end for
11:      Set  $S(m, Y, x) = T$ .
12:    end for
13:  end for
14: end for

```

Output: summation structure S .

Operation $L = \text{extractsum}(S, \Omega)$

Input: summation structure S , occlusion map Ω .

```

1: for each pixel location  $(x, y) \in D$  do
2:   Set  $s = 0$ ,  $m = 0$  and  $y' = y$ .
3:   repeat
4:     if  $y'$  is odd then
5:       Set  $s += \text{treeget}(S(m, y', x), \Omega(x, y))$ .
6:     end if
7:     Set  $y' = \lfloor \frac{y'}{2} \rfloor$  and  $m += 1$ .
8:   until  $y' = 0$ .
9:   Set  $P(x, y) = s$ .
10: end for

```

Output: pixel-spread color plane P .

Fig. 8. Operations *createstruct* and *extractsum*. L gives at each image location a list of color intensity values (v) and occlusion (ω) levels for all pixel-spread corners at that location. Operation $T = \text{treeadd}(T, v, \omega)$ adds pair (v, ω) to the tree as illustrated by Fig. 9. Operation *treeget* looks up the sum of all intensity values in the tree from the given node upwards that have a higher occlusion level. D is the set of locations in the $W \times H$ image domain, and $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote integer upward and downward rounding.

The $\mathcal{O}(N^2)$ cost that would be required by the naïve approach of independently computing the spreading at each occlusion level is reduced to the order of $\mathcal{O}(\log(N)^2 N)$ by using the method described in the next subsection.

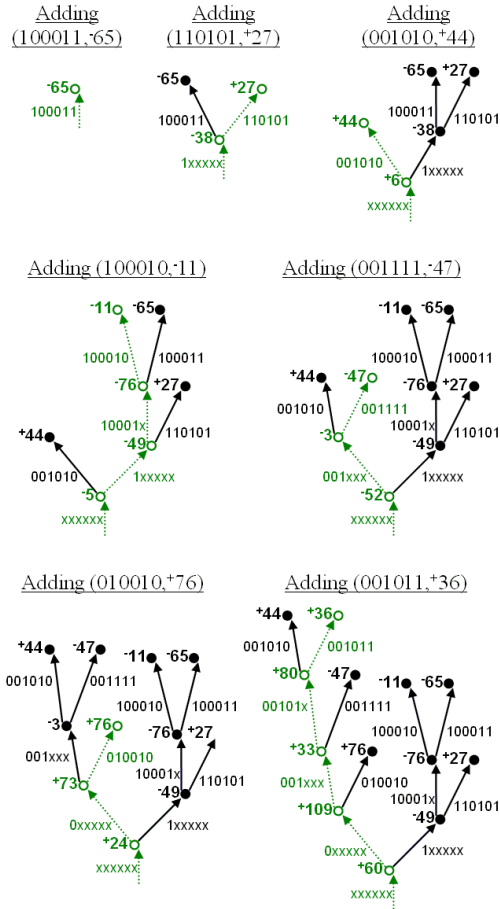


Fig. 9. An illustrative example of using the *treeadd* operation (see Alg. 1). Each stage in the diagram illustrates the addition of a new list item to the tree. Hollow circles and dotted arrows represent newly added nodes and pointers. The occlusion levels are represented here in binary, whereas the values to be stored and summed are represented in decimal. The letter “x” may be digit 0 or 1; for example, “001xxx” represents the range 001000 to 001111.

C. Implementation

The method used for computing the occlusive pixel-spreading of the proposed depth-blurring algorithm is illustrated in Fig. 7.

Firstly, the corner list array is created as follows. For each pixel location \mathbf{x} in the original image, the intensity $g=G(\mathbf{x})$ and blur level $b=B(\mathbf{x})$ are read. The spreaded intensity $v=\frac{g}{(2b+1)^2}$ is paired with occlusion level $\omega=\Omega(\mathbf{x})$. The pair (v, ω) is appended to four lists, associated with the four corner points $c^{11}(\mathbf{x}, b)$, $c^{12}(\mathbf{x}, b)$, $c^{21}(\mathbf{x}, b)$ and $c^{22}(\mathbf{x}, b)$ (see Eqs (6)-(9)), except for corner points which lie outside the image domain D , which are ignored.

The look-up structure is the key part of the algorithm, and is the part which reduces the complexity of the occlusive selective blurring from $\mathcal{O}(N^2)$ to $\mathcal{O}(\log(N)^2N)$. The structure takes as inputs the location of a pixel (by row and column) and an occlusion level, and outputs the sum of all the entries with a higher occlusion level in the corner list array in the rectangle bounded by that pixel and the top-left pixel of the image. This is created firstly by partitioning the image domain into a hierarchy of groups of adjacent rows of pixel locations,

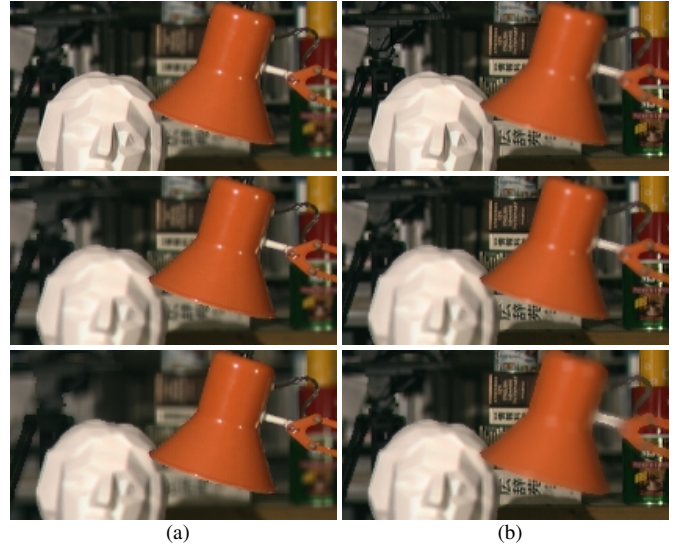


Fig. 10. Examples of occlusive selective blurring applied to Tsukuba head and lamp image and disparity map [10]. The depth map was simply taken as the reciprocal of the disparity map; then, each blur level is computed according to Eq. (1), with k chosen each time to attain a desired maximum blur level. (a) Lamp in focus. (b) Cans in focus. Max blur levels (i.e., max spread, in pixels, in any direction): top: 3; middle: 5; bottom: 10.

with sets of 1 row at the bottom level of the hierarchy, then sets of 2 adjacent rows at the next level, then sets of 4 adjacent rows, then sets of 8 adjacent rows, etc. For each level of the hierarchy, and each row group, a one-dimensional array (one location for each horizontal position) of trees is constructed (by *createstruct* in Alg. 1), each of which can be used to efficiently look up the sum of all values in that row group to the left of the given column. These trees are referred to herein as *occlusive sum look-up trees*. The look-up tree for each location in each of these one-dimensional arrays can be considered to hold an array of partial sums, one for each occlusion level.

The final block (*Occlusive Sum Extractor*) reads the occlusive sum look-up structure for every pixel in the image. To calculate the cumulative occlusive sum for a given row, column and occlusion level, the appropriate partial sums from the appropriate row groups are separately extracted then added together (as in *extractsum* in Alg. 1).

The workings of the occlusive sum look-up tree are illustrated in Fig. 9. Each tree node T , at the root of its own subtree, may be regarded as a pointer to a tuple, (V_T, R_T, L_T, U_T) , where $V_T \in \mathbb{R}$ is the graylevel (or color component) total for the subtree, $R_T \subset \mathbb{R}$ is the smallest contiguous subset of \mathbb{R} which spans all ω (occlusion) values covered by the subtree, and L_T and U_T are pointers to the lower and upper branch nodes which (optionally) sprout from node T . The occlusive sum look-up tree has the following properties: (1) When an addition or removal is made to the tree, an unaltered copy of how the tree was before may be retained at no extra cost, and at each stage of the algorithm, all non-new nodes are shared with the previously constructed trees. (2) The number of operations required to add a new occlusion level-value pair to a tree is of order $\mathcal{O}(\log(M))$, where M is the maximum absolute value of any integer occlusion level. (3) The amount of additional storage space required each time a new occlusion level-value

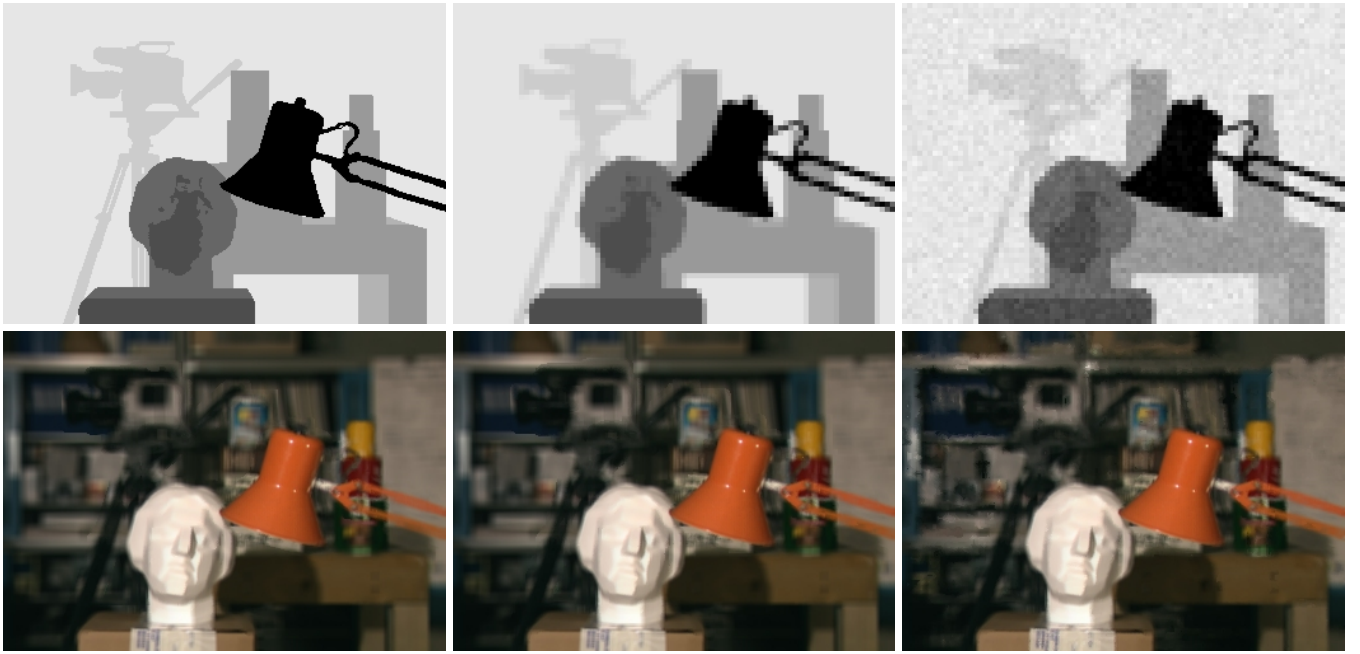


Fig. 11. Visual example of the effects of employing low-quality input blur maps. Top row, left to right: (1) unadulterated blur map (black = unblurred; max blur = 2 pixel widths); (2) resolution-reduced blur map (by a factor of 4); (3) resolution-reduced and noise-degraded blur map (resolution reduced a factor of 4 and Gaussian noise added of s.d. 0.1). Bottom row: synthetically blurred images using low-pass filtered versions of the given blur maps (using 15×15 uniform weighted average). The occlusion map applied in each case was the inverse of the filtered blur map.

pair is added is $\mathcal{O}(\log(M))$. (4) The number of operations required to look up a sum value for a given occlusion level is $\mathcal{O}(\log(M))$.

The asymptotic complexity of the cost of applying the algorithm to an N -pixel image ($N=HW$) with N occlusion levels is dominated by the *createstruct* and *extractsum* operations (see Alg. 1). These operations involve, for each pixel location, $\mathcal{O}(\log(H))$ calls to *treeadd* or *treeget*, each of which is of order $\mathcal{O}(\log(HW))$. Each call to either *createstruct* or *extractsum* will therefore be $\mathcal{O}(HW \log(H) \log(HW))$. Thus the Occlusive Pixel Spreader (see Figs 5 and 7) and the top-level Occlusive Selective Blurrer are $\mathcal{O}(HW \log(H) \log(HW))$ operations. Hence, assuming a fixed aspect ratio as N gets larger, the total cost is of order $\mathcal{O}(\log(N)^2 N)$. This compares with the $\mathcal{O}(N^2)$ cost that would be required by the naïve approach of independently computing the spreading at each occlusion level as illustrated in Fig. 6, assuming the worst case scenario of a different occlusion level for every pixel.

Fig. 10 shows the results of applying the proposed occlusive selective blurring technique to a raw image and disparity map. Fig. 11 compares blurring using low-resolution and noisy blur maps. It can be observed that the visible degradation to the blurred image is more affected by the blur map’s noise degradation than resolution reduction. The level of visual quality of the output depends on how well the depth map matches the perceived depths of each part of the scene. The method will in fact fail when either the resulting depth-blurred image appears to be degraded (i.e., when the image has noticeable distortion other than realistic depth-of-field) or if the wrong objects are in sharp focus, which is equally a problem for foveation filtering.

Fig. 12 compares the output of the proposed technique

to real-world depth of field effects as caused by the lens of a camera. Note that if the level of blur already present is sufficient for sourcing the depth map from a depth-from-defocus technique [24], the in-focus depth should be chosen as the existing focal plane, so that the synthetic blurring will enhance the photographer’s original choice of depth of interest.

III. EVALUATION METHOD

This section describes the evaluation method to assess the relative merits of two styles of space-variant blurring, namely depth and foveation blurring. These two styles of blurring are evaluated in an image coding context as a preprocessing stage prior to a JPEG codec.

A. Subjective testing

Two types of test are performed, namely a single stimulus test and a *method-of-adjustment* [17] test. The first type of test, which we refer to as *SSCQS*, is a single-stimulus modified version of Variant I of the *Double-Stimulus Continuous Quality Scale (DSCQS)* method [25], producing a score for each image in the range 0 to 100. These tests deliberately avoid showing any image under test alongside its reference image.

The method-of-adjustment test is designed to address the problem of the DSCQS adjectives (*excellent/good/fair/poor/bad*). This test performs JPEG coding of the unblurred test image in real time to produce, for each image, an *equivalent distortion* as judged by the subject. Each subject is instructed as follows: “The right-hand picture is associated with a vertically-sliding scale that will change the picture’s quality when you move it. You are asked to move this sliding scale up or down until it is your opinion that both pictures have the

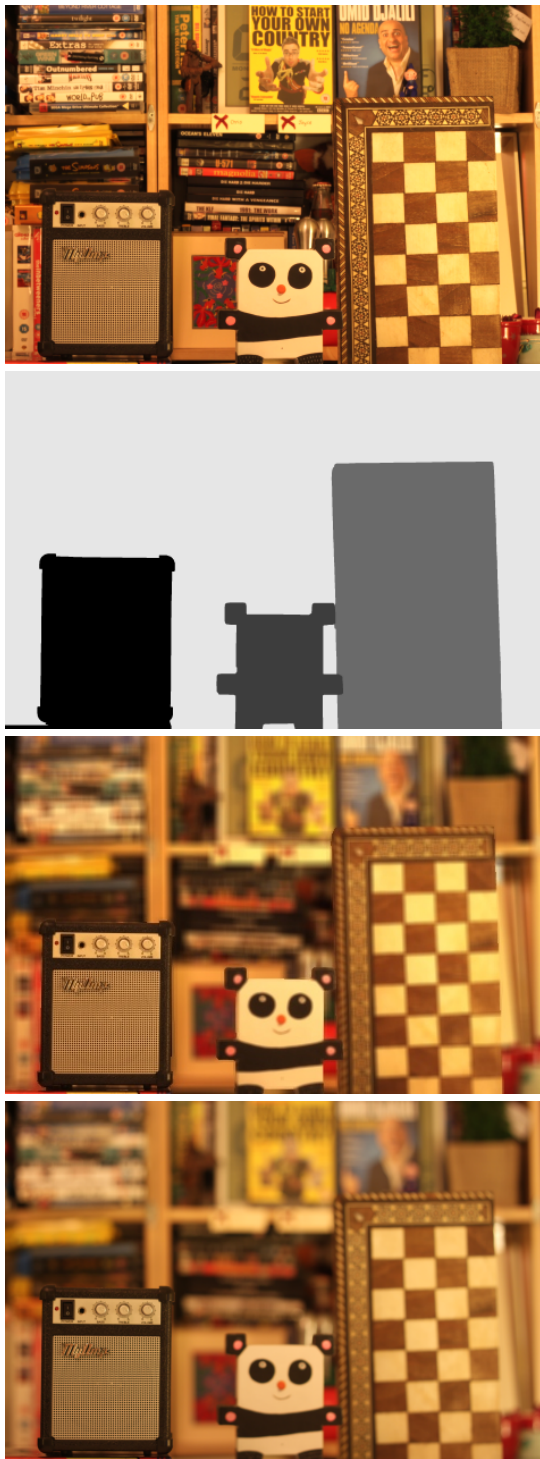


Fig. 12. Visual comparison between the output of the proposed approach and true depth of field as cause by a camera lens. Top: input image. Upper middle: manually-created input blur map (black: unblurred), the inverse of which was used as the occlusion map. Lower middle: emulated depth of field using the proposed approach, with average blur set to 3 (i.e., spread of 3 pixel widths in each direction). Bottom: real depth of field created by the lens of a camera (a Canon EOS 500D with 50mm lens, set at F2.8).

same overall quality.” In order to represent this quality on a meaningful scale, we record the compression ratio of each image, in logarithmic form to reduce sensitivity to outliers when processing results, as opposed to using the compression ratio

directly. We therefore store $\log(b_2/b_1)$ where b_1 represents the bitrate of the (preblurred) image under test and b_2 is the bitrate of the equivalent-quality unblurred JPEG-encoded image.

As required for DSCQS Variant I, each test image (or pair of images in the case of method-of-adjustment tests) is displayed to the subject until he chooses to proceed to the next test. The ordering of images is randomized, with the constraint that no two images from the same raw image are shown in succession. No subject spent more than 30 minutes in a complete test session.

B. Blurring

Our foveation method assumes an established *contrast threshold formula* [26], which provides a model of the directional variation of eye sensitivity. Specifically, we used the cutoff frequency interpretation of Wang, Lu & Bovik [4], whereby, for all e ,

$$f_c(e) = \frac{e_2 \ln(1/CT_0)}{(\|e\| + e_2)\alpha}, \quad (12)$$

where $f_c(e)$ is the spatial cutoff frequency (in cycles per unit angle) for a given retinal eccentricity, e (that is, the angle, relative to the observer’s eye, between a given point and the point of focus), and e_2 , α and CT_0 were constants defined as follows: $e_2 = 2.3^\circ$, $\alpha = 0.106^\circ/\text{cycle}$ and $CT_0 = 1/64$. The viewing direction is assumed to be head-on and angles are taken directly from pixel co-ordinates using a fixed conversion factor based on a viewing angle of one pixel width at the nearest point of the image to the viewer. The conversion from $f_c(e)$ values into a blur map is done by calculating the corresponding $1/f_c(e)$ values and scaling them proportionally, such that the desired mean blur level is attained. This is so that the imposed blurring will be the same at each location relative to the eye’s local resolution.

In order to provide a fair comparison, both blurring schemes are implemented using the proposed algorithm, with the only differences being the blur maps and occlusion maps used. The blur maps used for the depth blurring are created in a manner such that, firstly, the chosen point of interest is in focus and, secondly, the histogram of the number of pixels incurring each blur level is exactly the same as for the foveation blurring. Therefore it can only be the spatial distribution and nature of the blurring, rather than the amount of blurring, that determined the outcome of these experiments. The histogram of the depth blurring is created by taking the depth map and ranking image locations in order of how close their depth values are to that of the chosen point of interest. The blur levels are then assigned in order of blurring, such that the point of interest is the sharpest in focus. When the depth values alone are not sufficient to define the ordering of the pixels (that is, when the same depth value is shared by more than one pixel), the order is resolved according to their distances from the point of interest.

C. Experimental setup

Six publicly available images were chosen for the experiments. Three of them, *Cones* (450×375) *Dolls* (463×370) and

TABLE I
SSCQS AND METHOD-OF-ADJUSTMENT TEST SCORES

Raw image	Blur level	Rate (bpp)	Normalized SSCQS score (0-100)				Method-of-adjustment score (cmpr ratio)			
			Foveated (mean)	Depth-blurred (mean)	Comparison		Foveated (mean)	Depth-blurred (mean)	Comparison	
					Mean	C.I.			Mean	C.I.
Cones	3	0.4	48.42	42.82	-5.61	5.10	1.121	1.141	1.019	1.113
		0.3	25.13	22.91	-2.22	4.73	1.254	1.253	1.000	1.089
	5	0.4	43.11	46.07	2.96	5.45	0.982	1.120	1.141	1.081
		0.5	51.53	54.63	3.10	6.17	0.880	0.859	0.976	1.117
		0.4	29.00	43.85	14.85	4.14	0.873	1.014	1.162	1.156
Dolls	3	0.4	43.00	42.27	-0.73	3.91	1.159	1.207	1.042	1.107
		0.3	22.32	22.26	-0.06	4.11	1.153	1.191	1.033	1.065
	5	0.4	39.56	44.47	4.91	5.06	1.015	1.062	1.046	1.062
		0.5	50.48	57.86	7.38	4.96	0.899	1.018	1.133	1.138
		0.4	24.83	36.87	12.03	5.16	0.862	0.963	1.118	1.083
Art	3	0.4	53.03	56.71	3.68	5.28	1.036	1.074	1.036	1.100
		0.3	33.53	33.63	0.09	4.51	1.087	1.211	1.114	1.104
	5	0.4	50.16	55.00	4.84	3.97	1.009	1.032	1.023	1.075
		0.5	56.07	66.71	10.64	5.05	0.902	0.940	1.043	1.099
		0.4	32.50	46.44	13.94	5.11	0.857	0.956	1.116	1.110
Akiyo	3	0.4	62.97	62.41	-0.56	4.28	0.999	1.037	1.038	1.081
		0.3	34.72	36.38	1.66	5.29	1.094	1.122	1.026	1.041
	5	0.4	61.52	62.44	0.93	3.38	1.019	0.936	0.918	1.078
		0.5	68.26	69.32	1.06	3.80	0.824	0.824	1.000	1.090
		0.4	24.90	32.29	7.39	4.88	0.750	0.747	0.995	1.040
Foreman	3	0.4	40.03	38.60	-1.43	3.85	1.087	1.131	1.040	1.051
		0.3	19.18	21.21	2.03	2.85	1.235	1.247	1.010	1.056
	5	0.4	39.31	41.69	2.38	4.13	1.082	1.068	0.987	1.049
		0.5	45.66	50.55	4.90	5.58	0.911	0.912	1.001	1.067
		0.4	22.48	25.39	2.91	3.88	0.946	0.958	1.013	1.064
Silent	3	0.4	32.86	38.66	5.79	4.84	1.087	1.139	1.048	1.073
		0.3	17.22	17.70	0.49	4.05	1.242	1.277	1.028	1.043
	5	0.4	33.14	40.59	7.45	4.45	1.117	1.199	1.073	1.091
		0.5	52.83	57.67	4.83	4.94	1.068	1.156	1.082	1.084
		0.4	21.68	28.97	7.29	4.74	0.926	1.006	1.086	1.085

Art (463×370), were from the Middlebury Stereovision test set [22], [23] (with continuously-varying disparity maps). Each of the remaining three was the first frame of the well-known *Akiyo*, *Foreman* and *Silent* video sequences (all 352×288) (with a two-level background-foreground depth map, with the human as foreground). For each test image, we selected a fixation point on a face, as commonly done in the assessment of foveation techniques (e.g., [5], [27], [28]). The use of a single fixation point reflects a number of foveated coding publications [5], [7], [29], [30].

We used the Sun Microsystems Standard JPEG Image Writer (v. 0.5) and focussed on a range of bitrates in the region of 0.5 bits per pixel (bpp) or less. We investigated three blur levels: 3, 5 and 10 pixel widths. These levels are the average of the b values applied when spreading a pixel over an area of width $2b + 1$. Each b value was rounded to the nearest integer value before being used, so that inter-pixel interpolation will never be necessary. For the middle blur level, three fixed bitrates were applied: 0.3, 0.4 and 0.5 bpp; for the others, only 0.4 bpp were applied. All images were shown to the subject at native screen resolution. We used an LCD monitor with a resolution of 0.264 mm per pixel and assumed a viewing distance of 40 cm (± 10 cm). A specially-devised graphical user interface displayed the images under test, located centrally on the screen with mid-gray background. The slider bar for user input was located vertically on the rightmost part of the screen. User input was provided by an optical mouse located

TABLE II
MEAN PER-SUBJECT SCORE COMPARISONS ACROSS IMAGES

Blur Level	Rate (bpp)	Normalized SSCQS results		Method-of-adjustment results	
		Cones, Dolls & Art	Akiyo, Foreman & Silent	Cones, Dolls & Art	Akiyo, Foreman & Silent
3	0.4	-0.89	1.27	1.032	1.042
	0.3	-0.73	1.39	1.049	1.021
5	0.4	4.24	3.58	1.07	0.993
	0.5	7.04	3.6	1.051	1.028
	0.4	13.61	5.86	1.132	1.031
Overall mean:		4.65	3.14	1.067	1.023

on a desk in front of the subject, above which the monitor was held at eye level by an adjustable stand.

Forty-three non-expert subjects performed both types of tests for a selection of test images, blur levels and bitrates. Precautions were taken to ensure subjects were ignorant of any processing that had been applied to the images. All subjects had normal or corrected to normal vision. In both types of test, each subject was asked to assess overall quality; the subject was asked to think in terms of his preference in choosing an image as the backdrop of his computer desktop.

All the results of both tests types were passed through the recommended screening for DSCQS tests [25]. For the results we will discuss in Section IV, no observers needed to be rejected.

IV. DISCUSSION

Table I shows the mean scores of both test types for both styles of preblurring. In each case, a higher score is better than a lower score. The SSCQS scores were normalized so that each subject had the same mean and standard deviation as the overall mean and standard deviation across subjects; following this, all statistics were computed using the recommended DSCQS formulae [25]. For SSCQS, the statistics were computed on the scores directly. For the method-of-adjustment tests, the statistics were computed on the log compression ratios; however, to add meaning to these figures, their antilogarithms are displayed (i.e., compression ratios instead of log compression ratios); therefore *mean* here is the *geometric mean* for the method-of-adjustment results (but the normal, *arithmetic mean* for the SSCQS results). Comparison figures between the equivalent results for foveated and depth-based blurred images are also given; for the SSCQS results these are the mean score differences (positive = “depth-based is better”) whereas for the method-of-adjustment results they are relative ratios (greater than one = “depth-based is better”). For the SSCQS results, the half-widths of the 95% confidence intervals are given (i.e., true = estimate \pm CI), while the multiplicative equivalents of these are given for the method-of-adjustment results (i.e., true = estimate \times CI). Statistically significant results (where the *comparison* values lie outside the confidence intervals) are highlighted in bold font.

The average of the compression ratios across the test images was 1.016 for the foveated images, and 1.060 for the depth-blurred images. That is, the average foveated JPEG image was as good as as the equivalent unblurred JPEG image with 1.6%

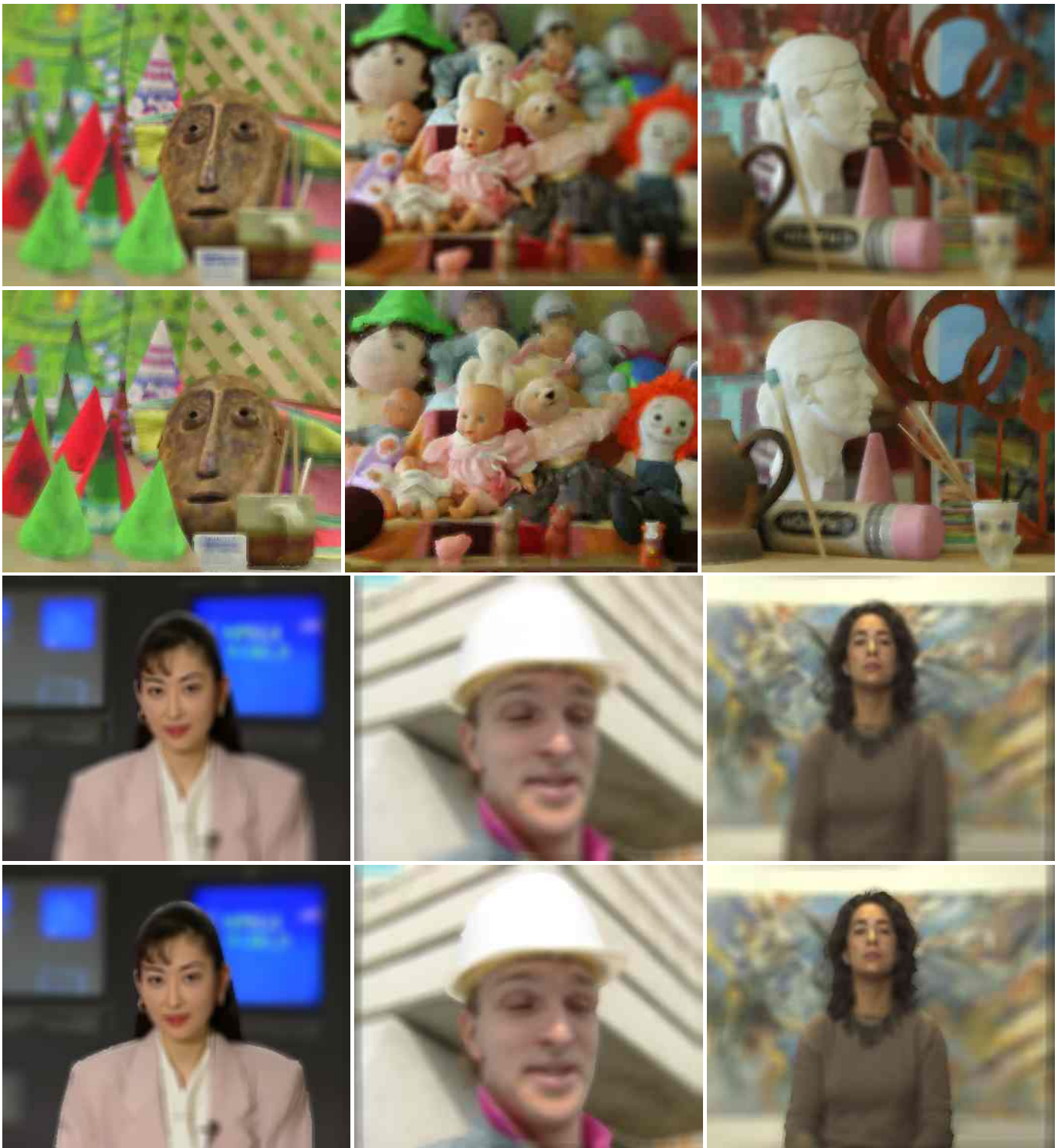


Fig. 13. Test images in the case of mean blur level 10 and 0.4 bpp encoding. Rows 1 and 2, left to right: *Cones*, *Dolls* and *Art*. Rows 3 and 4, left to right: *Akiyo*, *Foreman* and *Silent*. Rows 1 and 3 employ foveation-preblurring, and rows 2 and 4 give the equivalent depth-preblurred images. The point of interest is the right eye of the face or face-like object in each image (the central doll in the case of *Dolls*). The depth-preblurred images in row 2 were generated using multi-level disparity maps from the Middlebury test set (see Fig. 1, left, and Fig. 17). The depth-preblurred images in row 4 were generated using two-level depth maps which separate the foreground (person) from the background (see Fig. 1, right).

more bits, and the average depth-preblurred JPEG image was as good as the equivalent unblurred JPEG image with 6.0% more bits.

Table II shows the averages of comparison values across subjects and images. All test images for the maximum blur

level (10) are shown in Fig. 13.

Overall, 14 out of 30 images gave statistically significant results in at least one of the two types of test. Of these, 12 results indicated an average preference for the depth blurring over the foveation blurring. The *Foreman* image yielded no sta-

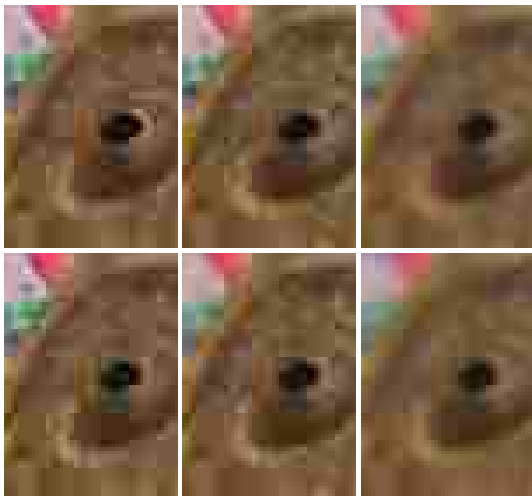


Fig. 14. Sample zoomed-in extracts from different variants of the *Cones* image. Preblurring: top: foveated; bottom: depth-based. Blur levels, left to right: 3, 5 and 10. All encoded at 0.4 bpp. The pseudorandom JPEG artifacts around the eye can be seen to be worse for the depth-based blurring in the case of blur level 3 but worse for the foveation in the case of blur level 10, explaining the statistically significant negative and positive score differences for these images.

tistically significant results whatsoever for any of its variants. This is understandable as, for this image, the visual differences between the two types of blurring are not easy to distinguish even when side by side (see Fig. 13). For this image, the area classed as foreground (the face) occupied a large portion (roughly 30%) of the image, and within this area, both types of blurring were identical, due to the preservation of blur-level histograms (see section III-B). Two results only (the SSCQS test for the minimum-blur *Cones* image and the method-of-adjustment test for the 0.4 bpp mid-blur *Akiyo* image) gave statistically significant results favouring the foveation blurring over the depth blurring. These can be explained by fact that at low blur levels and bitrates, the JPEG compression artifacts can have visual dominance over the blurring effects, and the question of which type of blurring has better perceived quality can become obscured by chance differences in the appearances of compression artifacts (see Fig. 14).

Of the statistically significant results, 9 were from the Middlebury test set images and 5 were from the images with two-level depth maps. This is reflected in higher cross-image mean score differences of 4.65 compared with 3.14 for the SSCQS tests and 1.067 compared with 1.023 for the method-of-adjustment tests (see Table II). In terms of bitrate, this means that, on average, the depth-preblurred images were as good as the 6.7%-higher bitrate foveated equivalents in the case of the Middlebury test set images, and as good as the 2.3%-higher bitrate foveated equivalents in the case of the other test images.

The results show a clear preference for the higher blurring level of ten pixel-widths, for which the method-of-adjustment tests gave statistically significant results for all three Middlebury test set images, and for which the SSCQS tests gave statistically significant results for all images other than *Foreman*.



Fig. 15. Zoomed-in portions of sample test images of (left to right) *Cones* (blur 5, 0.4 bpp), *Akiyo* (blur 10, 0.4 bpp) and *Silent* (blur 5, 0.4 bpp). Top row: foveation preblurring; bottom row: depth-based preblurring. All these images yielded statistically significant results in at least one of the two test types, all in favour of depth-based blurring.



Fig. 16. Zoomed-in portions of *Dolls*, with foveation preblurring (top) and depth-based preblurring (bottom), with (left to right) blur levels 3, 5 and 10 and JPEG bitrates 0.4 bpp, 0.5 bpp and 0.4 bpp.

Fig. 15 shows sample extracts from test images that exhibited statistically significant positive results. In each case, the distribution of blurring as provided by the depth blurring achieves a more satisfying image than the foveation. For example, for *Silent*, the foveation causes an undesirable level of blurring of the necklace, whereas the depth blurring causes slightly greater blurring of the background, which is more acceptable to the viewer. Fig. 16 shows sample extracts from different variants of the *Dolls* image. Statistically significant results were obtained for the images with blur levels 5 and 10, but not for blur level 3, for which the JPEG artifacts dominated over the visible differences in blurring.

A further point of note is the robustness of the proposed algorithm in the case of small patches of missing information from the disparity map. The occlusive effects of the algorithm causes all blurring in these regions to be completely contained within the regions, thus making them barely noticeable due to their small size. Examples of such missing information can be seen in Fig. 17, in the form of small visible patches of black (representing zero disparity, which is interpreted as maximum

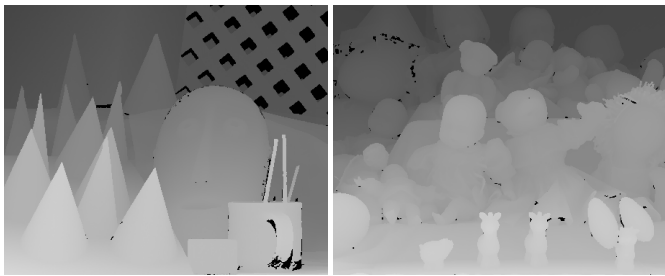


Fig. 17. Examples of disparity maps associated with the test images. It is possible to notice the errors in the form of black patches.

distance from the camera). For these patches, no artifacts are apparent in the corresponding depth-blurred images.

To summarize, the proposed depth blurring approach was found to be significantly preferable to the foveation filtering for 12 out of 30 test images for at least one of two types of subjective tests; a converse preference was found for only 2 out of 30 test images. On an *equivalent JPEG quality* scale, the depth-preblurred images were as good as the 6.7%-higher bitrate foveated equivalents in the case of test images for which we used a high-detail disparity map, and as good as the 2.3%-higher bitrate foveated equivalents in the case of images with background-foreground depth maps.

V. CONCLUSIONS

We have presented a novel selective blurring algorithm that provides a simulation of limited depth of field, with the desirable properties of aiming to mimic naturally-occurring occlusion effects and of being able to handle a continuous range of blurring and occlusion levels. The algorithms has only an $\mathcal{O}((\log N)^2 N)$ cost for an N -pixel image. We have demonstrated the algorithm in the context of space-variant prefiltering for bitrate reduction, with the argument that when human fixation has to be estimated because eye tracking is unavailable, blurring of this style will generally be visually more acceptable than the equivalent level of foveation blurring. The approach is particular appealing now given the increasing availability of cameras providing depth information (e.g., stereo or time-of-flight cameras or set-top boxes with software for 2D to 3D video conversion).

As future work, we will develop our approach to work with circular point spread functions and to cater for partial occlusion, by applying a gradual occlusion of the blur of far objects when the boundaries of nearer objects are themselves blurred.

REFERENCES

- [1] Z. Wang and A. C. Bovik, "Foveated image and video coding," in *Digital Video, Image Quality and Perceptual Coding*, H. R. Wu and K. R. Rao, Eds. CRC Press, 2006, ch. 14, pp. 431–457, ISBN 0-8247-2777-0.
- [2] A. P. Bradley and F. W. M. Stentiford, "Visual attention for region of interest coding in JPEG 2000," *J. Visual Communication and Image Representation*, vol. 14, no. 3, pp. 232–250, Aug. 2003.
- [3] T. Popkin, A. Cavallaro and D. Hands, "Accurate and Efficient Method for Smoothly Space-Variant Gaussian Blurring," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1362 – 1370, 2010.
- [4] Z. Wang, L. Lu and A. C. Bovik, "Foveation scalable video coding with automatic fixation selection," *IEEE Trans. Image Process.*, vol. 12, no. 2, pp. 243–254, Feb. 2003.
- [5] S. Liu and A. C. Bovik, "Foveation embedded DCT domain video transcoding," *J. Visual Communication and Image Representation*, vol. 16, no. 6, pp. 643–667, Dec. 2005.
- [6] B. A. Wandell, "Foundations of vision," p. 236, Sinauer Associates, 1995, ISBN 0-87893-853-2.
- [7] P. Kortum and W. Geisler, "Implementation of a foveated image coding system for image bandwidth reduction," *Proc. SPIE, Vol. 2657*, pp. 350–360, Apr. 1996.
- [8] L. Itti, "Automatic foveation for video compression using a neurobiological model of visual attention," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1304–1318, Oct. 2004.
- [9] T. Popkin, A. Cavallaro and D. Hands, "Distance Blurring for Space-Variant Image Coding," in *Proc. IEEE ICASSP*, Taipei, Taiwan, Apr. 2009, pp. 665–668.
- [10] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International J. Computer Vision*, vol. 47, no. 1-3, pp. 7–42, 2002.
- [11] A. Saxena, M. Sun and A. Y. Ng, "Make3D: Learning 3D Scene Structure from a Single Still Image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 824–840, May 2009.
- [12] G. Zhang, J. Jia, T.-T. Wong and H. Bao, "Consistent depth maps recovery from a video sequence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 974–988, Jun. 2009.
- [13] B. Huhle, S. Fleck and A. Schilling, "Integrating 3D Time-of-Flight Camera Data and High Resolution Images for 3DTV Applications," in *3DTV CON - The True Vision*, May 2007, ISBN 978-1-4244-0722-4.
- [14] "Project Natal in detail," <http://www.xbox.com/en-GB/news-features/news/Project-Natal-in-detail-050609.htm> (accessed Sep 16th, 2010).
- [15] A. Cavallaro, O. Steiger and T. Ebrahimi, "Semantic video analysis for adaptive content delivery and automatic description," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 10, pp. 1200–1209, Oct. 2005.
- [16] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.
- [17] J. G. Snodgrass, "Psychophysics," in *Experimental Sensory Psychology*, B. Scharf, Ed. Scott, Foresman and Company, 1975, ch. 2, p. 27, ISBN 0-673-05428-4.
- [18] R. L. Cook, T. Porter and L. Carpenter, "Distributed ray tracing," in *SIGGRAPH '84: Proc. of the 11th Computer Graphics and Interactive Techniques*, vol. 18, no. 3, Jul. 1984, pp. 137–145.
- [19] S. Lee, G. J. Kim and S. Choi, "Real-Time Depth-of-Field Rendering Using Anisotropically Filtered Mipmap Interpolation," *IEEE Trans. Vis. Comput. Graphics*, vol. 15, no. 3, pp. 453–464, 2009.
- [20] M. Kass, A. Lefohn and J. Owens, "Interactive Depth of Field Using Simulated Diffusion on a GPU," 2006, <http://graphics.pixar.com/library/DepthOfField/paper.pdf> (accessed Jan 21st, 2011).
- [21] I. van der Linde, "Multi-resolution image compression using image foveation and simulated depth of field for stereoscopic displays," in *Proc. SPIE*, vol. 5291, 2004, pp. 71–80.
- [22] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in *Proc. IEEE Conf. on Computer Vision and Pattern Rec.*, vol. 1, Madison, WI, USA, Jun. 2003, pp. 195–202.
- [23] D. Scharstein and C. Pal, "Learning conditional random fields for stereo," in *Proc. IEEE Conf. on Computer Vision and Pattern Rec.*, Minneapolis, MN, USA, Jun. 2007, pp. 1–8.
- [24] W. N. Klarquist, W. S. Geisler and A. C. Bovik, "Maximum-likelihood depth-from-defocus for active vision," in *IROS '95: Proc. International Conf. Intelligent Robots and Systems*, vol. 3. Washington, DC, USA: IEEE Computer Society, Aug. 1995, pp. 374–379.
- [25] ITU-R Rec. BT.500-11, *Methodology for the subjective assessment of the quality of television pictures*, International Telecommunication Union Std., 2002.
- [26] W. S. Geisler and J. S. Perry, "A real-time foveated multiresolution system for low-bandwidth video communication," *Proc. SPIE, Vol. 3299*, pp. 294–305, Jul. 1998.
- [27] E.-C. Chang and C. K. Yap, "A wavelet approach to foveating images," in *Proc. of the 13th Symposium on Computational Geometry*. ACM, 1997, pp. 397–399.
- [28] Z. Wang and A. C. Bovik, "Embedded foveation image coding," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1397–1410, Oct. 2001.
- [29] M. M. Farid, F. Kurugollu and F. D. Murtagh, "Adaptive wavelet eye-gaze based video compression," in *Proc. SPIE, Vol. 4877*, Mar. 2003, pp. 255–263.
- [30] Ç. Dikici, H. Isil Bozma and M. R. Civanlar, "Fovea based coding for video streaming," in *Proc. International Conf. Image Analysis and Recognition*, vol. 3211, Sep. 2004, pp. 285–294.



Timothy Popkin received the BA degree in mathematics from Cambridge University, U.K., in 1995, and the M.Sc. degree in signal processing and machine intelligence from the University of Surrey, U.K., in 1997.

He spent nine years as a software engineer at CAE Electronics, U.K., and then at Thales Communications, U.K. In 2006, he joined Queen Mary University of London, under A. Cavallaro, working in space-variant image and video coding and filtering. He received the Ph.D. degree in Electronic

Engineering in 2010.

Dr. Popkin won a student paper award in IEEE ICASSP 2009 for "Distance blurring for space-variant image coding".



Andrea Cavallaro received the Laurea (*summa cum laude*) in Electrical Engineering from the University of Trieste, Italy, in 1996 and the Ph.D. degree in Electrical Engineering from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 2002.

In 1996 and 1998, he served as a research consultant at the Image Processing Laboratory, University of Trieste, Italy, working on compression algorithms for very low bitrate video coding and on digital image sequence de-interlacing. From 1998 to 2003

he was a research assistant at the Signal Processing Laboratory of EPFL. Since 2003, he has been with Queen Mary University of London (QMUL), UK, where he is now Professor.

Professor Cavallaro was awarded a Research Fellowship with British Telecommunications (BT) in 2004/2005; three student paper awards at IEEE ICASSP in 2005, 2007 and 2009; and the best paper award at IEEE AVSS 2009. He is Associate Editor for the IEEE Signal Processing Magazine, the IEEE Trans. on Multimedia, the IEEE Trans. on Signal Processing and for the Journal on Information Security. He served as General Chair for IEEE/ACM ICDSC 2009, BMVC 2009, and IEEE AVSS 2007. He has authored more than 100 papers, including 10 book chapters.



David Hands graduated from the University of Aberdeen, U.K. with an M.A. degree in Psychology, where he remained after graduation to work as a research assistant before studying for his the Ph.D. degree at the University of Essex, U.K.

He is a Research Group Leader with BT Innovate & Design. He joined BT in 1997 to work in the Human Factors research unit. He has research interests in the development of objective quality measurement algorithms and subjective quality testing. He has published more than 50 papers on a range of topics

including knowledge elicitation techniques, subjective quality assessment methods, and objective perceptual quality models. He is an active member of ATIS IIF QoS, ITU-T SG9, ETSI STQ and VQEG standards groups.