Proc. 21st IEEE International Conference on Image Processing
(ICIP 2014, Paris, France, October 2014), 4822-4826, 2014.
The final publication is available on IEEE Xplore.

COLOUR IMAGE COMPRESSION WITH ANISOTROPIC DIFFUSION

Pascal Peter and Joachim Weickert

Mathematical Image Analysis Group, Faculty of Mathematics and Computer Science,
Campus E1.7, Saarland University, 66041 Saarbrücken, Germany.
{peter, weickert}@mia.uni-saarland.de

ABSTRACT

Schmaltz et al. (2009) have shown that for reasonably high compression rates, diffusion-based codecs can exceed the quality of transformation-based methods such as JPEG 2000. They store only data at a few optimised pixel locations and inpaint missing data with edge-enhancing anisotropic diffusion (EED). However, research on compression with diffusion methods has mainly focussed on grey-value images, and colour images have been compressed in a straightforward way using anisotropic diffusion in RGB space. So far, there is no sophisticated diffusion-based counterpart to the colour mode of JPEG 2000. To address this shortcoming we introduce an advanced colour compression codec that exploits properties of the human visual system in YCbCr space. Since details in the luma channel Y are perceptually relevant, we invest a large fraction of our bit budget in its encoding with high fidelity. For the chroma channels Cb and Cr, the stored information can be very sparse, if we guide the EED-based inpainting with the high quality diffusion tensor from the luma reconstruction. Experiments demonstrate that our novel codec outperforms JPEG 2000 and compression with RGB-diffusion, both visually and quantitatively.

***Index Terms*—** colour, edge-enhancing anisotropic diffusion, compression, YCbCr space, luma preference

1. INTRODUCTION

Transformation coders such as JPEG [1] and JPEG 2000 [2] exploit the properties of the human visual system in order to improve the perceived fidelity of compressed colour images. In particular, they use YCbCr or YUV colour spaces and compress the chroma channels in a coarser manner either by subsampling or by omitting fine-scale wavelet coefficients.

A fundamentally different approach to image compression is pursued by diffusion-based compression methods. They started out in 2005 as a proof of concept [3] for an alternative to transformation coders. Meanwhile, they have evolved to a refined stage that enables them to surpass state-of-the-art competitors. In particular, the R-EED codec [4] uses the inpainting qualities of edge-enhancing anisotropic diffusion (EED) [5] to beat JPEG 2000 for low-texured images. However, R-EED has mainly been optimised for the compression of grey-value data. It only supports colour compression in RGB space so far [6], where a weighting of channels according to perceptual importance is not possible. Therefore, on colour data, R-EED and the transformation-based coders do not compete on equal footing. In this paper, we address this disadvantage by introducing a new colour compression mode to R-EED.

Our Contribution. We extend R-EED by a so-called *luma preference (LP) mode* that is based on YCbCr space. This novel compression mode relies on two core ideas: 1. The luma channel is more important for the human visual system than the colour components. Therefore, it should be stored with higher quality. 2. The diffusion tensor of the luma reconstruction steers the inpainting of the chroma channels. This improves the quality of the chroma reconstruction.

Related Work. Our reconstruction of the chroma channels has been inspired by Kaufhold's image colourisation method [7]. She uses linear anisotropic diffusion guided by the luma channel of a YCbCr representation to propagate manually added colour strokes to the remainder of the image. Also some transformation-based coders exploit colourisation ideas for compression. Most of them rely on the method of Levin et al. [8]. In [9], an extended version of Levin's method is used in combination with JPEG to restore colour information from samples in CIELAB space. A machine learning approach to the problem is investigated in [10]. It interprets Levin's method as a learning algorithm and incorporates a modified version into JPEG. Furthermore, variations of a colourisation method based on Markov random fields have been applied in a postprocessing step to JPEG [11] and JPEG

2000 [12].

2. THE R-EED CODEC FOR RGB IMAGES

The foundation of our diffusion-based colour image compression method is the R-EED codec of Schmaltz et al. [4, 6]. It stores only a small amount of image data at well-chosen positions and reconstructs the remainder of the image via diffusion-based inpainting. The evaluation in [6] has shown that edge-enhancing anisotropic diffusion (EED) [5] has better reconstruction qualities than other partial differential equations. Let us consider some rectangular image domain $\Omega \subset \mathbb{R}^2$, and let $\boldsymbol{f} = (f_1, f_2, f_3)^\top$ be the original image with RGB channels $f_i : \Omega \to \mathbb{R}$, $i \in \{1,2,3\}$. Furthermore, we assume that we have stored $\boldsymbol{f}$ only on a subset $K \subset \Omega$, the so-called *inpainting mask*. Then the reconstruction in the *inpainting domain* $\Omega \setminus K$ is computed as the steady state $(t \to \infty)$ of the anisotropic diffusion equation

$$\partial_t u_i = \operatorname{div}(\boldsymbol{D}(\boldsymbol{J})\boldsymbol{\nabla} u_i), \quad i \in \{1,2,3\} \tag{1}$$

with reflecting boundary conditions. The known data on K is unaffected during the evolution. The initialisation in the inpainting domain $\Omega \setminus K$ at time $t = 0$ can be chosen arbitrarily, as it has no influence on the steady-state.

The diffusion equation (1) is guided by the joint diffusion tensor $\boldsymbol{D}$, a positive definite 2×2 matrix that adapts itself to the local image structure (see also [13]). This adaptation relies on Di Zenzo's structure tensor for colour images [14]:

$$\boldsymbol{J} := \sum_{k=1}^{3} \boldsymbol{\nabla} u_{k,\sigma} \boldsymbol{\nabla} u_{k,\sigma}^\top, \tag{2}$$

where $u_{k,\sigma}$ denotes a convolution of the channel u_k with a Gaussian of standard deviation σ. The nonnegative eigenvalues $\mu_1 \geq \mu_2$ of $\boldsymbol{J}$ measure the contrast in the directions of the eigenvectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$. To reflect the image structure, the diffusion tensor $\boldsymbol{D}$ uses the same eigenvectors. In order to reduce diffusion *across* edges, its eigenvalue λ_1 is given by the Charbonnier diffusivity [15]

$$\lambda_1 = g(\mu_1) := \frac{1}{\sqrt{1 + \mu_1/\lambda^2}} \tag{3}$$

with some contrast parameter $\lambda > 0$. Full diffusion *along* edges is achieved with a constant second eigenvalue $\lambda_2 := 1$.

The Gaussian convolution within the structure tensor $\boldsymbol{J}$ guarantees that structural information is propagated to the neighbourhood of each pixel. This way, EED can inpaint edges also in situations where only very few pixels of an edge contour are specified; see [6] for experiments. Optimising the inpainting mask K and the contrast parameter λ can increase the quality of the inpainting result.

In practice, R-EED limits the admissible pixel set K to a regular adaptive grid that can be represented by a binary decision tree. This allows to store the pixel locations efficiently and reduce the search space for an optimal choice of K. In order to find a good set K, we use a subdivision strategy: If the reconstruction error in a rectangular subarea exceeds a threshold, the area is split across its largest dimension. This subdivision inserts pixels to K and is represented by a node in a binary tree. For more details, we refer to [6].

The known colour data is quantised and stored with a suitable entropy encoder. In addition to an optimisation of the mask K, the contrast parameter λ and the number of quantised grey-values q, the colour data can also be optimised in a so-called *brightness optimisation step*: Allowing errors at the small set K of specified pixels may reduce errors in the large inpainting domain $\Omega \setminus K$. More details can be found in [6].

3. LUMA PREFERENCE MODE

The choice of RGB space for the original colour compression mode of R-EED enforces equal treatment of all three colour channels in vector-valued diffusion inpainting. In our novel extension to R-EED, we propose to use YCbCr colour space instead. YCbCr separates the image data into intensity information in the luma channel Y and colour information in the chroma channels Cb and Cr; see e.g. [16].

For the human visual system, errors in the chroma channels have a smaller impact on the perceived quality than errors in the luma component (see e.g. [17]). This effect can be exploited by compressing the luma channel in a higher quality than the chroma data. Even though the mean square error (MSE) in RGB space might increase due to an unequal treatment of the channels, the perceived overall quality can be improved.

On the basis of these observations, we introduce a *luma preference (LP) mode* for colour compression with R-EED. In LP mode, the luma and chroma components are compressed in separate steps and with different compression ratios. For a given compression ratio, more of the total bit budget is dedicated to Y than to Cb or Cr. A new parameter, the *LP ratio* r, determines the weighting between the available storage s_Y, s_{Cb} and s_{Cr} for the respective channels in the form

$$s_Y = r \cdot s_{Cb} = r \cdot s_{Cr}. \tag{4}$$

LP mode chooses s_Y such that an overall compression ratio $R : 1$ with respect to the original file size s_O is achieved:

$$s_Y := \frac{s_O}{(1 + 2r)R}. \tag{5}$$

This *channel weighting* in YCbCr mode constitutes the first significant difference to the RGB mode of R-EED. However, we aim to go one step beyond a pure increase in perceived visual fidelity: Instead of just redistributing the error from intensity to chroma components, we want to decrease the overall reconstruction error. LP mode achieves this goal by exploiting the correspondences of important image structures between luma and chroma channels. In the following, we describe an inpainting approach that uses the high quality luma reconstruction to improve the inpainting results in the chroma components. Let an image in YCbCr space be given by the vector $(f_Y, f_{Cb}, f_{Cr})^\top$. Each channel $c \in \{Y, Cb, Cr\}$ has an individual set of known data $K_c \subset \Omega$ in this formulation. The reconstruction on the inpainting domain $\Omega \setminus K_c$ is obtained as the steady state of

$$\partial_t u_c = \operatorname{div}(\boldsymbol{D}(\boldsymbol{\nabla} u_{Y,\sigma} \boldsymbol{\nabla} u_{Y,\sigma}^\top) \boldsymbol{\nabla} u_c). \tag{6}$$

In contrast to Equation (1), the tensor $\boldsymbol{D}$ in Equation (6) depends solely on the luma channel: Its eigenvectors $\boldsymbol{v}_1 = \boldsymbol{\nabla} u_{Y,\sigma}$ and $\boldsymbol{v}_2 = \boldsymbol{\nabla} u_{Y,\sigma}^\perp$ represent the local edge directions in the luma channel. In the direction $\boldsymbol{v}_2$ along luma edges, the eigenvalue $\lambda_2 := 1$ allows full diffusion. Across luma edges, the diffusion is inhibited according to the local contrast in the luma channel. To this end, we set the eigenvalue λ_1 to $g(|\boldsymbol{\nabla} u_{Y,\sigma}|^2)$ with the Charbonnier diffusivity g from Equation (3).

Thus, for the luma component, the diffusion-based reconstruction is identical to EED inpainting for grey-value images. However, the chroma channels are reconstructed with linear anisotropic diffusion that is guided by the diffusion tensor of the luma reconstruction. Since all edge information from the luma reconstruction is contained in its diffusion tensor, the *luma-guided inpainting* can also reconstruct these in the chroma channels. Therefore, LP mode needs to store less known data for the chroma components. The gained additional bit budget can be invested into the luma channel instead, which increases the overall reconstruction quality (see experiments in Section 4). With channel weighting and luma-guided diffusion, the core ideas of LP mode have been established. In the following, we present a detailed description of the compression and decompression steps of LP mode and the corresponding file formats.

Compression in LP mode proceeds in two steps.

Step 1: Luma Compression. We compress the luma channel like a grey-value image in R-EED. The contrast parameter λ, the number of quantised grey-values q, and the mask positions are optimised w.r.t. the MSE of the luma channel. Then we store these positions in a binary decision tree, the so-called *luma tree*. At these positions, we optimise the quantised intensity values. Afterwards a specific entropy coder such as PAQ [18] helps to remove redundancies.

Step 2: Chroma Compression. This step benefits strongly from the luma diffusion tensor from Step 1, since luma-guided chroma diffusion requires less interpolation points to achieve good results. Moreover, the positions of these points are less critical, such that we can use the same positions for both chroma channels. Thus, it suffices to built a single, small joint splitting tree for both chroma channels. In particular for small values of r, this avoids a lot of overhead in the coding of the positions. We invest this free bit budget in a better coding of colour values. Also for the chroma

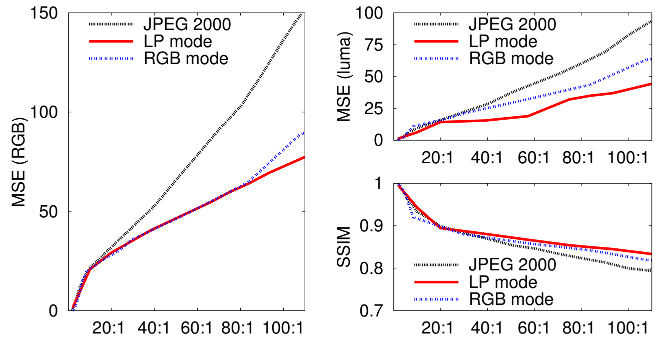

Fig. 1. Error comparison at different compression rates for the 256×256 image *peppers* shown in Fig. 2. **(a) Left:** Joint MSE in RGB space. Lower is better. **(b) Top right:** MSE in the luma channel. **(c) Bottom right:** SSIM. Higher is better.

compression, we perform optimisation of the chroma values. The number q_{CbCr} of quantised chroma levels and the type of entropy codec can be chosen independently from the corresponding decisions for the luma channel.

Container Generation and File Format. The data from both compression steps has to be stored in a container file. First we generate a common header that contains the image dimensions, all parameters of the method, and the file size s_C of the joint positional and colour data for the chroma channels. They are saved as binary numbers of custom length for each stored value. Afterwards the luma and chroma trees as well as the entropy coded pixel data is appended. We store the trees in form of a minimal and maximal tree depth and a sequence of bits that explicitly encodes the splitting decision in between the depth limits.

Decompression. For decompression, the header of the container file is read to obtain all parameters, including the chroma length s_C. With this length, we split the file into luma and chroma data and recover both trees. After reconstructing the mask from the decision tree and decoding the known image data, we reconstruct the luma channel with EED inpainting. Using the diffusion tensor from this inpainting step and the decoded colour data, the chroma channels are reconstructed as well. A backtransformation to RGB recovers the final decompressed image.

4. EXPERIMENTS

Using two well-known test images, we compare our new LP mode to the RGB mode of R-EED and to JPEG 2000. The test image *peppers* represents images with homogeneous areas, whereas *lena* serves as an example for images with fine-scale detail. We have implemented the anisotropic diffusion algorithms as in [6] and have chosen PAQ [18] as entropy coder. Since the overall mean square error (MSE) does not always adequately reflect the perceived quality, we also employ

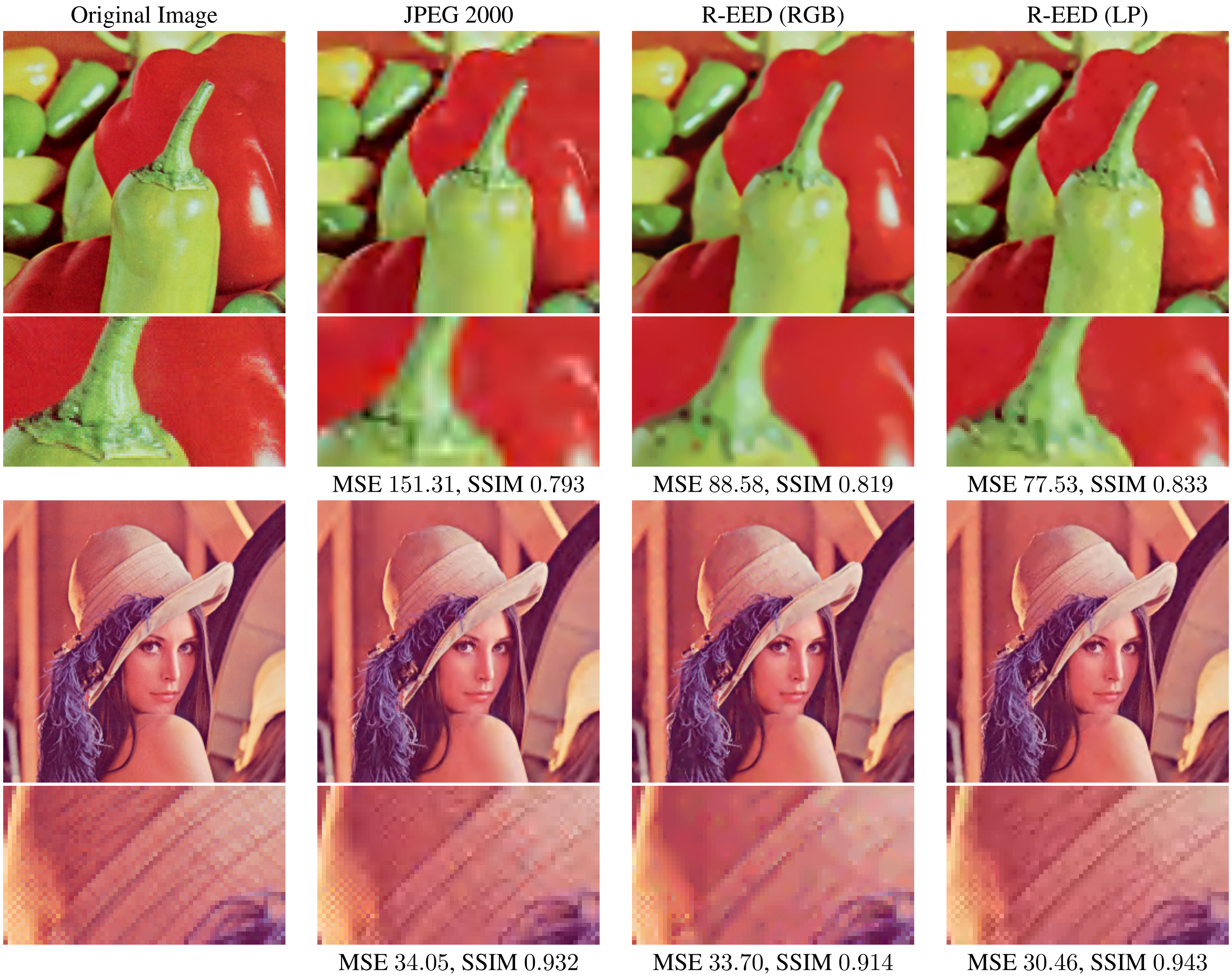

Fig. 2. Compression results (overall MSE and SSIM) for *lena* and *peppers*. **(a) Top:** Results for *peppers* (256×256 pixels) with compression ratio 110:1. **(b) Bottom:** Results for *lena* (256×256 pixels) with compression ratio 20:1.

the luma channel MSE and a dedicated perceptual measure, the structural similarity index (SSIM) for greyscale-converted colour images [19].

Fig. 1(a) shows that for *peppers*, both R-EED modes consistently outperform JPEG 2000. For compression rates below 80:1, the overall MSE of the LP mode and the RGB mode of R-EED is almost identical. However, for higher ratios, the LP mode performs up to 10% better than the RGB mode. Fig. 1(b) displays the MSE in the luma channel. Here one immediately sees that the channel weighting of our LP mode is rewarded by a very low luma MSE. Also the perceptually relevant SSIM curves in Fig. 1(c) confirm the favourable performance of R-EED in LP mode. Fig. 2(a) allows visual comparisons for high compression rates. We see that R-EED in LP mode gives good results in scenarios where the other codecs reach their limits: The overall MSE of the JPEG 2000 standard, for example, is almost twice as high.

On images that contain more texture, LP mode can improve the overall MSE of R-EED also for lower compression rates, as we can see in Fig. 2(b). For the *lena* image with ratio 20:1, R-EED in RGB mode is inferior to JPEG 2000: It lacks a sufficient bit budget to reproduce texture details. The LP mode, however, reconstructs more fine-scale detail due to the channel weighting. It surpasses JPEG 2000 w.r.t. the overall MSE and SSIM.

5. CONCLUSION AND OUTLOOK

We have presented the first diffusion-based codec in YCbCr colour space. It benefits from the perceptual properties of the human visual system. Experiments show that our luma preference mode gives a higher quality than the traditional RGB mode of diffusion coding and also surpasses JPEG 2000. Extensions to video compression and more efficient storage of positional data are promising areas for further research.

Our contribution is one more step towards supplementing diffusion codecs with a similar level of sophisticated engineering as established transformation-based methods. Other recent steps along these lines include progressive mode coding [20], dedicated diffusion codecs for depth maps [21, 22, 23], and 3-D data compression [24]. We are convinced that a lot of potential in this area still awaits being explored.

6. REFERENCES

[1] W. B. Pennebaker and J. L. Mitchell, *JPEG: Still Image Data Compression Standard*, Springer, New York, 1992.

[2] D. S. Taubman and M. W. Marcellin, Eds., *JPEG 2000: Image Compression Fundamentals, Standards and Practice*, Kluwer, Boston, 2002.

[3] I. Galić, J. Weickert, M. Welk, A. Bruhn, A. Belyaev, and H.-P. Seidel, "Towards PDE-based image compression," in *Variational, Geometric and Level-Set Methods in Computer Vision*, N. Paragios, O. Faugeras, T. Chan, and C. Schnörr, Eds., vol. 3752 of *Lecture Notes in Computer Science*, pp. 37–48. Springer, Berlin, 2005.

[4] C. Schmaltz, J. Weickert, and A. Bruhn, "Beating the quality of JPEG 2000 with anisotropic diffusion," in *Pattern Recognition*, J. Denzler, G. Notni, and H. Süße, Eds., vol. 5748 of *Lecture Notes in Computer Science*, pp. 452–461. Springer, Berlin, 2009.

[5] J. Weickert, "Theoretical foundations of anisotropic diffusion in image processing," *Computing Supplement*, vol. 11, pp. 221–236, 1996.

[6] C. Schmaltz, P. Peter, M. Mainberger, F. Ebel, J. Weickert, and A. Bruhn, "Understanding, optimising, and extending data compression with anisotropic diffusion," *International Journal of Computer Vision*, in press.

[7] L. Kaufhold, "Variational methods for the colourisation of greyscale images," Bachelor Thesis, Dept. of Computer Science, Saarland University, Saarbrücken, Germany, 2013.

[8] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," in *Proc. SIGGRAPH 2004*, New York, NY, Aug. 2004, pp. 689–694.

[9] T. Horiuchi and S. Tominaga, "Color image coding by colorization approach," *EURASIP Journal on Image and Video Processing*, vol. 2008, no. 158273, 2008.

[10] L. Cheng and S. V. N. Vishwanathan, "Learning to compress images and videos," in *Proc. 24th International Conference on Machine Learning*, Corvallis, OR, June 2007, pp. 161–168.

[11] H. Noda, N. Takao, and M. Niimi, "Colorization in YCbCr space and its application to improve quality of JPEG color images," in *Proc. 14th IEEE International Conference on Image Processing*, San Antonio, TX, Sept. 2007, vol. 4, pp. 385–388.

[12] H. Noda and M. Niimi, "Local MAP estimation for quality improvement of compressed color images," *Pattern Recognition*, vol. 44, no. 4, pp. 788–793, 2011.

[13] J. Weickert, "Coherence-enhancing diffusion of colour images," *Image and Vision Computing*, vol. 17, no. 3–4, pp. 199–210, Mar. 1999.

[14] S. Di Zenzo, "A note on the gradient of a multi-image," *Computer Vision, Graphics and Image Processing*, vol. 33, pp. 116–125, 1986.

[15] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Transactions on Image Processing*, vol. 6, no. 2, pp. 298–311, 1997.

[16] W. K. Pratt, *Digital Image Processing: PIKS Inside*, Wiley, New York, third edition, 2001.

[17] C. J. van den Branden Lambrecht, Ed., *Vision Models and Applications to Image and Video Processing*, Kluwer, Boston, 2001.

[18] M. Mahoney, "Adaptive weighing of context models for lossless data compression," Tech. Rep. CS-2005-16, Florida Institute of Technology, Melbourne, Florida, Dec. 2005.

[19] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[20] C. Schmaltz, N. Mach, M. Mainberger, and J. Weickert, "Progressive modes in PDE-based image compression.," in *Proc. 30th Picture Coding Symposium*, San Jose, CA, Dec. 2013, pp. 233–236.

[21] J. Gautier, O. L. Meur, and C. Guillemot, "Efficient depth map compression based on lossless edge coding and diffusion," in *Proc. 29th Picture Coding Symposium*, Krakow, Poland, May 2012, pp. 81–84.

[22] Y. Li, M. Sjostrom, U. Jennehag, and R. Olsson, "A scalable coding approach for high quality depth image compression.," in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, Zurich, Switzerland, Oct. 2012, pp. 1–4.

[23] S. Hoffmann, M. Mainberger, J. Weickert, and M. Puhl, "Compression of depth maps with segment-based homogeneous diffusion," in *Scale-Space and Variational Methods in Computer Vision*, A. Kuijper, K. Bredies, T. Pock, and H. Bischof, Eds., vol. 7893 of *Lecture Notes in Computer Science*, pp. 319–330. Springer, Berlin, 2013.

[24] P. Peter, "Three-dimensional data compression with anisotropic diffusion," in *Pattern Recognition*, J. Weickert, M. Hein, and B. Schiele, Eds., vol. 8142 of *Lecture Notes in Computer Science*, pp. 231–236. Springer, Berlin, 2013.