Proceedings of ALGORITMY 2016
pp. 194–203

EFFICIENT CO-DOMAIN QUANTISATION FOR PDE-BASED IMAGE COMPRESSION

LAURENT HOELTGEN* AND MICHAEL BREUß†

Abstract. Finding optimal data for inpainting is a key problem for image compression with partial differential equations (PDEs). Not only the location of important pixels but also their values should optimise the compression quality. The position of such important data is usually encoded in a binary mask. The corresponding pixel values are real valued and yield prohibitively high storage costs in the context of data compression. Therefore, quantisation strategies for the pixel value domain are mandatory to obtain high compression ratios. While existing methods to quantise the data for PDE-based compression show good quality, unfortunately, they are too slow for many applications. We discuss several strategies, based on data clustering models from machine learning, to speed up the quantisation step.

Key words. Laplace interpolation, inpainting, compression, quantisation, clustering, partial differential equations

AMS subject classifications. 65N99, 90C25, 97N50

1. Introduction. A major challenge in data analysis is the reconstruction of an arbitrary signal from very few data points. In image processing this interpolation problem is called inpainting. Often one has no influence on the given data and thus improvements can only be made by introducing more powerful reconstruction models. In some interesting applications however, one has the freedom to choose the data used for the reconstruction. For instance, in recent approaches related to image compression [3–5, 10, 17, 19, 21, 27] the authors select suitable interpolation data for reconstructions via partial differential equations (PDEs). In recent years it has been shown that PDE-based image compression schemes can compete with other state-of-the-art approaches [24, 25, 27]. Even simple methods that rely on the interpolation capabilities of the Laplacian may yield more accurate reconstructions than JPEG 2000. Unfortunately, the underlying optimisation problem concerned with selecting optimal point locations that occurs during the compression process is computationally intensive and run times in the range of hours are quite common. Another crucial step in the encoding process consists in finding the optimal quantisation of the pixel values. Storing these in floating point precision is prohibitively expensive and often unnecessary. Already thirty different grey values may be sufficient to achieve convincing reconstructions. These thirty colours can be encoded with 5 bit instead of the 64 bit required for a floating point value in double precision. Finding the optimal number and distribution of these grey values is however a complicated task.

In this work we suggest to use data clustering approaches to perform the quantisation step. Clustering methods have been applied in the literature in the context of colour quantisation but unrelated to the intended image compression task. As a popular example for such methods let us mention the median cut algorithm of Heckbert [16]. Data clustering algorithms are also at the heart of many machine learning applications,

*Brandenburg University of Technology, Platz der Deutschen Einheit 1,03046 Cottbus, Germany (hoeltgen@b-tu.de)

†Brandenburg University of Technology, Platz der Deutschen Einheit 1,03046 Cottbus, Germany (breuss@b-tu.de)

where the goal is to sort a certain set of data points into different categories or groups such that the elements in each group exhibit a strong similarity. The exact definition of what means similarity is problem specific. We refer to [1, 11, 13] for an overview on possible approaches and algorithms. The large variety of applications where data classification occurs have lead researchers to propose a variety of quality measures. We refer to [14] for a discussion on their viability.

Our Contribution. In this paper we investigate for the first time in the literature several combinations of algorithms and features for grey value quantisation tasks within the context of PDE-based image compression. We show that well established methods are capable of reducing the number of necessary grey values by as much as 80% without causing any loss in the reconstruction quality. The reduced grey value map exhibits a much lower entropy than the original grey value distribution and is therefore much better suited for compressing the image data. All our experiments have a run time in the range of less than a seconds whereas the best performing methods from the literature for PDE-based image compression tend to require hours to optimise the same data set.

2. PDE-based Image Compression. Even though most diffusion-type PDEs can be used to reconstruct the data, previous works have shown that homogeneous, linear diffusion ranks among the best performing ones [24]. Its simplicity allows a thorough mathematical analysis and an extensive optimisation [17, 18]. Since the quantisation is likely to depend on the underlying reconstruction process we thus focus on homogeneous diffusion in this work.

2.1. Inpainting with Homogeneous Diffusion. Inpainting with homogeneous diffusion (sometimes also called Laplace interpolation) is a rather simple reconstruction method that is well suited for highly scattered data in arbitrary dimensional settings. It can be modelled as follows. Let $f : \Omega \to \mathbb{R}$ be a smooth function on some bounded domain $\Omega \subset \mathbb{R}^n$ with a sufficiently regular boundary $\partial\Omega$. Throughout this work, we will restrict ourselves to the case $n = 2$ (grey value images) even though many of the results hold for arbitrary $n \geqslant 1$. Moreover, let us assume that there exists a closed nonempty set of known data $\Omega_K \subsetneq \Omega$ that will be interpolated by the underlying diffusion process. Homogeneous diffusion inpainting considers the following partial differential equation with mixed boundary conditions.

$$-\Delta u = 0 \text{ on } \Omega \setminus \Omega_K, \quad \text{with} \quad u = f \text{ on } \partial\Omega_K, \text{ and } \partial_n u = 0 \text{ on } \partial\Omega \setminus \partial\Omega_K, \tag{2.1}$$

where $\partial_n u$ denotes the derivative of u in the outer normal direction. We assume that both boundary sets $\partial\Omega_K$ and $\partial\Omega \setminus \partial\Omega_K$ are non-empty. Equations of this type are commonly referred to as mixed boundary value problems and sometimes also as Zaremba's problem. The existence and uniqueness of solutions has been extensively studied during the last century. Showing that (2.1) is indeed solvable is by no means a trivial feat. We refer to [12] for an extensive study of linear elliptic partial differential equations. A particularly easy case is the 1-D setting, where the solution can obviously be expressed using piecewise linear splines interpolating data on $\partial\Omega_K$.

2.2. Optimal Inpainting Positions. Finding good inpainting positions is a task related to the free knot problem from interpolation theory [6, 15]. In the special context of image inpainting and image compression the authors of [5,8,17,21] suggested several strategies to find good locations. Mainberger et al. [21] propose stochastic optimisations that bear similarities to simulated annealing approaches. Chen et al. [5] use fast bi-level optimisation schemes and in [17] Hoeltgen et al. present an optimal

control model. Finally the works of Demaret et al. [7–9] use adaptive triangulations and mesh optimisation strategies, whereas Ochs et al. [23] suggest fast numerics for the task at hand.

2.3. Continuous Grey Value Optimisation. In [18, 21] the authors discuss approaches to find the best pixel values. These algorithms have in common, that they all yield real valued floating point results. If we denote the reconstruction operator that solves (2.1) for a given mask Ω_K by $M(\Omega_K)$, then this problem can be formulated as

$$\arg\min_u \left\{ \|M(\Omega_K)\, u - f\|^2 \right\} \tag{2.2}$$

Since our initial PDE is linear, the task stated in (2.2) corresponds to a linear least squares problem and can be solved efficiently. An alternative discrete optimisation has been suggested by Schmaltz et al. [27]. The latter approach iterates over the inpainting data and selects in a greedy way the currently best quantised grey value. The iterations continue until a convergent state is reached. While this approach comes conceptually close to clustering strategies, it has two notable drawbacks: *(i)* The approach requires that the distribution of the quantised values is fixed at the beginning, i.e. it does not change during the iterations; *(ii)* to test a certain grey value for its quality requires the solution of a large and sparse linear system, i.e. the computational costs are considerable as the algorithm requires the solution of thousands of linear systems. Nevertheless, we remark that it yields excellent qualitative results.

3. Clustering Inpainting Data. The following paragraphs discuss potential choices for the feature selection as well as for the algorithmic execution. Our motivation is to use models that come conceptually close to the continuous grey value optimisation and the quantisation aware approach of Schmaltz et al. [27]. Therefore, we focus on the squared Euclidean distance as cost function and seek the best set of grey values for the mask pixel locations. However, we will refrain from using the reconstruction inside our algorithms in order to obtain fast methods.

3.1. Feature Selection. Our setup provides us the full image data f as well as an optimised inpainting mask Ω_K. There exist many possible choices to extract interesting features from this data. The following *six feature choices* seem reasonable:

1. The grey value information at the mask positions.
2. The grey value information at the mask positions with their frequency count.
3. The grey value information of all pixels.
4. The grey value information of all pixels with their frequency count.
5. The full image pixel information (grey value and pixel position) at the mask positions only.
6. The full image data (grey value and pixel position).

We remark that including the pixel position into the feature vector renders each sample unique. In that case it makes no sense to track their frequency as it will always be 1.

Approaches that exploit the full image information have better chances at adapting to features in the image that are not covered by the mask pixels alone. On the other hand, optimised positions for the inpainting with homogeneous diffusion are usually in the vicinity of edges and other important structures. Therefore, we expect that the outcomes of these two setups will be similar.

Strategies that focus exclusively on the grey values of an image are likely to perform fastest since they can be carried out on the histogram. As soon as we include

the spatial position into the feature set we obtain as much unique feature values as pixels. For large images we may encounter memory or run time restrictions.

Finally let us remark that from a clustering perspective it is irrelevant whether we consider grey values or colour images. Only the mask optimisation and inpainting steps are more difficult to carry out for colour images.

3.2. Clustering Approaches for Image Quantisation. In this work we opt for some of the most popular and best understood approaches to classify our data. Since clustering approaches are commonly subdivided into partitional and hierarchical methods we select k-means clustering as a representative of the former class and consider hierarchical approaches based on Euclidean distances for the latter class. Furthermore, we consider a second partitional approach based on probabilistic methods. The k-means algorithm performs a hard thresholding when it assigns labels whereas the probabilistic approach tries to fit a Gaussian mixture model over the data and yields a probability that a certain feature belongs to one of the given probability density functions. These softer transitions from one cluster to the next one could be beneficial when the labelling is ambiguous.

Let us now briefly present the adopted strategies in some detail. In the following paragraphs we assume that we are given a set $X \subset \mathbb{R}^p$ of observed feature values $x_i \in X$ with $i = 1, \ldots, m$. The overall task consists in dividing X into k pairwise disjoint subsets C_j that still cover the whole set X.

3.2.1. K-means. The k-means algorithm of Lloyd [20] is one of the oldest and most popular clustering approaches. Its roots date back to 1957 [28], resp. 1967 [22]. It aims to partition the set X into k classes C_j such that

$$\arg\min_{C_1,\ldots,C_k} \left\{ \sum_{i=1}^{k} \left(\sum_{x_j \in C_i} \|x_j - m_i\|^2 \right) \right\} \tag{3.1}$$

becomes minimal. Here, the m_i represent the centroids of the clusters C_i. A detailed description of the k-means algorithm of Lloyd is given in Algorithm 1. We note that extensions, such as the kmeans++ approach [2] can also be applied to our setup. They merely improve the initialisation but have no influence on the algorithm otherwise.

Algorithm 1: K-Means Algorithm of Lloyd [20]

Input : Feature set X, number of clusters k
Output : Clusters $C_i \subseteq X$, centroids m_i, $i = 1, \ldots, k$
Initialise : Choose k initial centroids m_i arbitrarily

repeat
  Assign each feature x_j to its closest centroid m_i.
  Update position of the centroids m_i.
until *centroids m_i reach fixed point*

3.2.2. Hierarchical Clustering. In these methods, the clusters are represented hierarchically in a tree-like structure called dendrogram. Two approaches exist. Either the tree is set up top-down where one starts with a single cluster containing all features and which is iteratively split into smaller subclusters. Alternatively, a bottom-up strategy can be used which assigns in the beginning each feature to a single cluster and merges clusters in subsequent steps to larger supersets.

In this work we opt for the latter approach. As a means to merge two clusters we choose to use Ward's criteria [31]. It aims at minimising the increase of the within-cluster variance after the merging of two clusters. For any two clusters C_i and C_j, this increase is given by

$$\Delta_{i,j} \coloneqq \frac{2|C_i||C_j|}{|C_i| + |C_j|} \|m_i - m_j\|_2^2 \tag{3.2}$$

where m_i and m_j are the centroids of the clusters C_i and C_j respectively and where $|\cdot|$ denotes the cardinality of the cluster. The multiplicative factor 2 is included such that the formula coincides with the squared Euclidean distance for singleton clusters. The full algorithm is given in Algorithm 2.

Algorithm 2: Hierarchical Clustering with Ward's Method

Input : Feature set X
Output : Hierarchical cluster structure
Initialise : Assign each feature x_k to its own cluster C_k
repeat
  Find clusters C_i and C_j with smallest $\Delta_{i,j}$.
  Merge C_i and C_j into single cluster $C_{i \cup j}$.
until *single cluster remains*

We remark that hierarchical approaches do not yield a single clustering but rather a complete evolution similar to the well known scale spaces from image processing [32]. We will address the task of extracting a good clustering in a later paragraph.

3.2.3. Fuzzy Clustering with Gaussian Mixture Models. As a third alternative we consider a parametric Gaussian mixture model where all features are assumed to follow the distribution

$$\sum_{j=1}^{k} \omega_j \, \mathcal{N}(\,\cdot\,; \mu_j, \sigma_j) \tag{3.3}$$

and where $\mathcal{N}(\,\cdot\,; \mu_j, \sigma_j)$ is the probability density function of a Gaussian with mean μ_j and standard deviation σ_j. The difficulty lies in estimating the parameters ω_j, μ_j, and σ_j for all $j = 1, \ldots, k$, such that the resulting density function in (3.3) fits best to the observed features x_i. Additionally, the weights ω_j must be non-negative and sum up to 1. Once the optimal parameters are found, it is possible to assign each feature to one of the Gaussians to obtain a partitioning of the data. This assignment can be deduced with Bayes' Theorem. In order to derive the best parameter settings we use the popular expectation maximisation algorithm of Sundberg [29, 30]. The underlying idea of this approach is to maximise iteratively the log-likelihood function. The complete algorithm (referred to in the following as EM-GMM) is presented in Algorithm 3. The variable $T_{j,i}$ in this listing states the probability that an observation x_i stems from the Gaussian j. These probabilities $T_{j,i}$ mark the basis of our clustering strategy. We assign to each feature vector the distribution $\mathcal{N}(\,\cdot\,; \mu, \sigma)$ with highest probability. We emphasise that the classification is not strict as for the two other methods and may yield ambiguous results in certain cases. If an observation belongs with a probability 0.5 to two different Gaussians, there is no optimal assignment anymore.

Algorithm 3: Expectation Maximisation with Gaussian Mixture Model

Input : Feature set $X \coloneqq \{x_i \,|\, i = 1, \dots, n\}$, number of clusters k
Output : Fuzzy clustering of X into k sets
Initialise : Choose initial values for each parameter ω_k, μ_k, σ_k

repeat

Compute membership probabilities for all j, i

$$T_{j,i} = \frac{\omega_j \mathcal{N}(x_i; \mu_j, \sigma_j)}{\sum_\ell \omega_\ell \mathcal{N}(x_i; \mu_\ell, \sigma_\ell)} \tag{3.4}$$

Update parameters of the mixture model for all $j = 1, \dots, k$:

$$\omega_j = \frac{1}{n} \sum_{i=1}^{n} T_{j,i}, \quad \mu_j = \frac{\sum_i T_{j,i} x_i}{\sum_i T_{j,i}}, \quad \sigma_j = \frac{\sum_i T_{j,i} (x_i - \mu_j)}{\sum_i T_{j,i}} \quad \forall j \tag{3.5}$$

until *until fixed-point is reached*

Assign to each x_i the Gaussian $\mathcal{N}(\,\cdot\,; \mu_j, \sigma_j)$ with highest probability $T_{j,i}$

3.3. Quality Measures and Optimal Number of Clusters. When evaluating a partitioning of the data we need to consider two criteria. First, the quality of the clustering itself: Features in the same group should indeed be similar and also distinct from observations from other clusters. Secondly, we need to keep in mind that our ultimate goal is to reconstruct images from sparse quantised data by means of PDE-based inpainting. Since we are focusing on data compression tasks we can assume that the full original data is available and may be used as a ground truth. Thus, we consider two quantities to evaluate the quality of our partitioning. On the one hand we use the silhouette value [26] and on the other hand we consider the Mean Squared Error (MSE).

The silhouette value is a method to interpret and validate the consistency within clustered data. It is applicable to any clustering method and appealing due to its simplicity. If the feature x belongs to the cluster C, then we define the silhouette $S(x)$ of x to the cluster C as

$$S(x) \coloneqq \begin{cases} 0, & \operatorname{dist}(x, C) = 0, \\ \frac{\operatorname{dist}(x,D) - \operatorname{dist}(x,C)}{\max\{\operatorname{dist}(x,C), \operatorname{dist}(x,D)\}}, & \text{else}, \end{cases} \tag{3.6}$$

where $\operatorname{dist}(x, C)$ corresponds to the average distance of x to all other elements from the cluster C. The silhouette values range from -1 to 1. Since $\operatorname{dist}(x, C)$ is a measure of how dissimilar x is to its own cluster, a small value means it is well matched. Furthermore, a large $\operatorname{dist}(x, D)$ suggests that x is badly matched with respect to D. It follows that an $S(x)$ close to one means that x is appropriately labelled. If $S(x)$ is close to negative one, then by the same logic we see that x would be better placed in D. An $S(x)$ near zero means that x is on the border of two clusters. Finally, the average $S(x)$ over all data of a cluster is a measure of how tightly grouped all the data in the cluster is. Thus the average $S(x)$ over all data can be understood as a measure of how well the data has been grouped. If there are too many or too few clusters some of the clusters will typically display much narrower silhouettes than the rest. Thus silhouette plots and averages may be used to determine the optimal number of clusters within a data set.

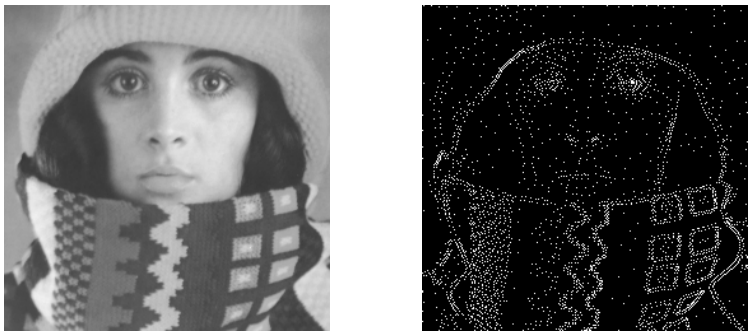

FIGURE 4.1. *Left: trui input image (*256×256 *pixel), Right: inpainting mask with 5% density, pixel locations used for the reconstruction are marked in white.*

The MSE simply measures the average squared distance between the original ground truth image f and a reconstruction g from the quantised data. It can be computed as

$$\mathrm{MSE}\,(f,g) \coloneqq \frac{1}{|f|}\sum_{i,j}\left(f_{i,j}-g_{i,j}\right)^2 \quad , \tag{3.7}$$

where $|f|$ denotes the number of pixels in the image f.

Let us shortly summarise the criteria that we will consider. An ideal clustering should have a relatively high mean silhouette value. Averages above 0.6 are considered to be good. Furthermore, there should be no negative silhouette values, since those suggest erroneous labels. Secondly, uncompressed encodings require 256 grey values in their colour map. Good clusterings should have significantly less. A rule of thumb says that a reasonable quantisation should have around 35 distinct grey values. Finally, the reconstruction quality should not suffer from the smaller spectrum of possible grey levels. Therefore, the MSE of the quantised reconstruction should be roughly equal to the MSE of the unquantised reconstruction. Let us emphasise that PDE-based image compression is usually lossy, i.e. even the reconstruction with the unquantised data is not perfect and differs from the original input image.

4. Numerical Experiments. Our experiments cover the following choices:

1. Hierarchical clustering of the grey values at the mask positions.
2. Hierarchical clustering of the grey values and positions for all pixels.
3. K-means clustering of the grey values and positions for all pixels.
4. K-means clustering of the grey values at the mask positions.
5. Hierarchical clustering of the histogram of the grey values from the mask positions.
6. Hierarchical clustering of the histogram of the grey values from all image pixels.
7. EM-GMM clustering of the grey values at the mask positions.

Each test case uses the popular test image *trui* as well as an optimised inpainting mask with a density of 5% obtained with the optimal control approach of Hoeltgen et al. [17]. The image and the mask are depicted in Figure 4.1. In Figure 4.2 we have plotted the number of clusters against the mean squared error for the test cases 1, 4, and 7. The remaining setups performed significantly worse and will not be considered in the following anymore. The solid horizontal line in Figure 4.2 corresponds to the

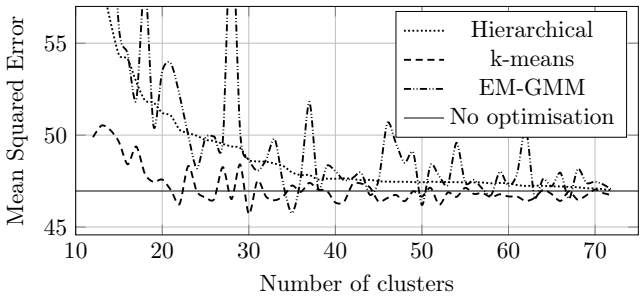

FIGURE 4.2. *MSE obtained from clustering the grey values from the mask positions by hierarchical clustering, k-means, and EM-GMM. The solid line indicates the error obtained by inpainting with the original unaltered data (186 grey values). The hierarchical clustering yields a very stable behaviour, whereas k-means exhibits sudden fluctuations. Similar fluctuations occur with EM-GMM, too. In this case the k-means approach has been able to improve the error from the unaltered data with only 30 colours. EM-GMM achieves a similar feat with 35 clusters.*

reconstruction error with the unquantised image data. Interestingly, the k-means and the EM-GMM approach have rather strong fluctuations in the quality of their clusterings. An issue that stems from the fact that the initialisation is random. It may be overcome by restarting the methods several times and choosing the best result. However such a strategy will also have an negative impact on the run time. Nevertheless, the k-means approach is capable of improving on the original reconstruction with just 30 clusters and the EM-GMM approach does the same for 35 clusters. The original image data had 186 distinct grey values.

The hierarchical approach is very stable and yields an almost monotonic decrease in the error when increasing the number of clusters. Figures 4.3 also presents the silhouette plots of the hierarchical and k-means approach for the experiment with 30 clusters. The findings correlate with the results depicted in Figure 4.2. The clustering of the k-means approach is superior to the the hierarchical strategy. The hierarchical clustering exhibits a high median silhouette value of 0.8 but its plot indicates that a few pixels may have bad labels. On the other hand, the k-means approach yields a clean graph that contains no negative coefficients and many values exceeding 0.8. Its mean value is 0.63 with a standard deviation of 0.28. Furthermore, all clusters have similar widths suggesting an overall good classification. The silhouette plot of the EM-GMM approach for 35 clusters has a similar appearance. A pure Matlab implementation of the k-means algorithm also has a run time of about 0.35 seconds for a single clustering. The EM algorithm is similarly fast. The hierarchical method yields a run time of 0.15 seconds for a single clustering. Overall, our whole evaluation is significantly faster than the method of Schmaltz et al. [27], which has a run time in the range of hours. Finally, the fact that we were able to reduce the number of grey values down to roughly a sixth of the original amount without any loss in the reconstruction quality is a significant advantage for the subsequent use of entropy coders, which perform much better in presence of smaller number of colours.

Let us remark, that in all our experiments, the models that relied on the grey value information alone performed better than those strategies that included the positional information of the pixels as well. Often, the difference between two pixel positions correlates with the difference in the grey values. Neighbouring pixels have similar colours and pixels that are far apart tend to have distinct values. Now, if we have two pixels $(2, 3, 127)$ and $(3, 4, 128)$, then the distance between the grey values is 1 whereas

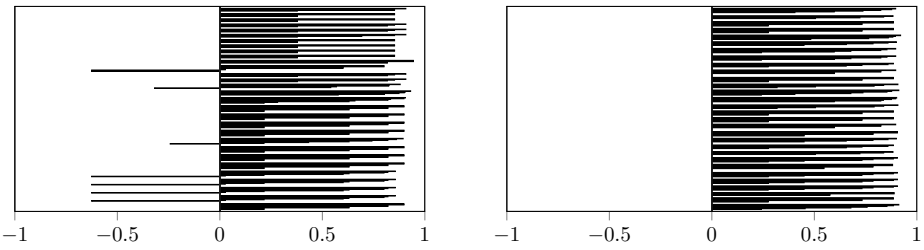

FIGURE 4.3. *Left: The silhouette plot corresponding to the hierarchical clustering approach with 30 clusters. We see that most clusters exhibit large values and very few pixels have been labelled wrongly. The average silhouette value being 0.59 and the median value being 0.8. Right: The silhouette plot corresponding to the k-means approach with 30 clusters. We see that all clusters exhibit large values and that no wrong labels have been assigned. The average silhouette value being 0.63 and a median of 0.73.*

the distance between the full pixel data is $\sqrt{3} > 1$. We conjecture that models that include the positional information are likelier to separate this data than models which only consider the colour information and that these unnecessary splittings cause a degradation of quality.

5. Summary and Conclusions. Our experiments show that we can reduce the number of colours used for PDE-based inpainting by as much as 80% without encountering any loss in the reconstruction quality. Our run times usually lie below one second and are significantly lower than for competing approaches. Therefore, we think that our approach is an important step in the development of a practicable image compression codec.

In the future we would like to perform a more thorough analysis of different clustering strategies to quantise the data in an optimal way for the reconstruction process. The integration of the reconstruction function into the clustering is certainly beneficial for the quality of the classifications.

REFERENCES

[1] C.C. Aggarwal and C. K. Reddy, editors. *Data Clustering, Algorithms and Applications*. CRC Press, 2014.

[2] D. Arthur and S. Vassilvitskii. K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1027–1035. SIAM, 2007.

[3] Z. Belhachmi, D. Bucur, B. Burgeth, and J. Weickert. How to choose interpolation data in images. *SIAM Journal on Applied Mathematics*, 70(1):333–352, 2009.

[4] A. Bourquard and M. Unser. Anisotropic interpolation of sparse generalized image samples. *IEEE Transactions on Image Processing*, 22(2):459–472, 2013.

[5] Y. Chen, R. Ranftl, and T. Pock. A bi-level view of inpainting - based image compression. *Computing Research Repository*, 2014. Available from `http://arxiv.org/abs/1401.4112v2`.

[6] Carl de Boor. Good approximation by splines with variable knots II. In G. Watson, editor, *Conference on the Numerical Solution of Differential Equations*, volume 363 of *Lecture Notes in Mathematics*, pages 12–20. Springer, 1974.

[7] L. Demaret and A. Iske. Scattered data coding in digital image compression. In A. Cohen, J.-L. Merrien, and L. L. Schumaker, editors, *Curve and Surface Fitting: Saint-Malo 2002*, pages 107–117. Nashboro Press, 2003.

[8] L. Demaret and A. Iske. Advances in digital image compression by adaptive thinning. *Annals of the MCFA*, 3:105–109, 2004.

[9] L. Demaret, A. Iske, and W. Khachabi. Contextual image compression from adaptive sparse data representations. In Rémi Gribonval, editor, *Proceedings of SPARS'09 (Signal Processing*

with Adaptive Sparse Structured Representations Workshop), 2009. Available online: `https://hal.inria.fr/inria-00369491`.

[10] 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*, volume 3752 of *LNCS*, pages 37–48. Springer, 2005.

[11] G. Gan, C. Ma, and J. Wu. *Clustering: Theory, Algorithms and Applications*. SIAM, 2007.

[12] D. Gilbarg and N. Trudinger. *Elliptic Partial Differential Equations of Second Order*. Springer, 2001.

[13] A. Gordon. Hierarchical classification. In P. Arabie, L. Hubert, and G. Soete, editors, *Clustering and Classification*, pages 65–121. World Scientific, 1996.

[14] L. Guerra, V. Robles, C. Bielza, and P. Larranaga. A comparison of clustering quality indices using outliers and noise. *Intelligent Data Analysis*, 16:703–715, 2012.

[15] H. Hamideh. On the optimal knots of first degree splines. *Kuwait Journal of Science and Engineering*, 29(1):1–13, 2002.

[16] P. S. Heckbert. Color image quantization for frame buffer display. In *ACM SIGGRAPH '82 Proceedings*, pages 297–307, 1982.

[17] L. Hoeltgen, S. Setzer, and J. Weickert. An optimal control approach to find sparse data for Laplace interpolation. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 8081 of *LNCS*, pages 151–164. Springer, 2013.

[18] L. Hoeltgen and J. Weickert. Why does non-binary mask optimisation work for diffusion-based image compression? In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 8932 of *LNCS*, pages 85–98. Springer, 2015.

[19] D. Liu, X. Sun, F. Wu, S. Li, and Y.-Q. Zhang. Image compression with edge-based inpainting. *IEEE Transactions on Circuits, Systems and Video Technology*, 7(10):1273–1286, October 2007.

[20] S. P. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 2(28):129–137, 1982.

[21] M. Mainberger, S. Hoffmann, J. Weickert, C. H. Tang, D. Johannsen, F. Neumann, and B. Doerr. Optimising spatial and tonal data for homogeneous diffusion inpainting. In A. M. Bruckstein, B. M. Haar ter Romeny, A. M. Bronstein, and M. M. Bronstein, editors, *Scale Space and Variational Methods in Computer Vision*, volume 6667 of *LNCS*, pages 26–37. Springer, 2012.

[22] J. B. McQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297. University of California Press, 1967.

[23] P. Ochs, Y. Chen, T. Brox, and T. Pock. iPiano: Inertial proximal algorithm for non-convex optimization. *SIAM Journal on Imaging Sciences*, 7(2):1388–1419, 2014.

[24] P. Peter, S. Hoffmann, F. Nedwed, L. Hoeltgen, and J. Weickert. From optimised inpainting with linear PDEs towards competitive image compression codecs. In T. Bräunl, B. McCane, M. Rivers, and X. Yu, editors, *Advances in Image and Video Technology*, LNCS. Springer, 2015. To appear.

[25] P. Peter, C. Schmaltz, N. Mach, M. Mainberger, and J. Weickert. Beyond pure quality: Progressive mode, region of interest coding and real time video decoding in PDE-based image compression. *Journal of Visual Communication and Image Representation*, 31:253–265, 2015.

[26] P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Computational and Applied Mathematics*, 20:53–65, 1987.

[27] C. Schmaltz, J. Weickert, and A. Bruhn. Beating the quality of JPEG 2000 with anisotropic diffusion. In *Pattern Recognition*, volume 5748 of *LNCS*, pages 452–461. Springer, 2009.

[28] H. Steinhaus. Sur la division des corps matériels en parties. *Bull. Acad. Polon. Sci.*, 12(4):801–804, 1957.

[29] R. Sundberg. Maximum likelihood theory for incomplete data from an exponential family. *Scandinavian Journal of Statistics*, 1(2):49–58, 1974.

[30] R. Sundberg. An iterative method for solution of the likelihood equations for incomplete data from exponential families. *Communications in Statistics – Simulation and Computation*, 5(1):55–64, 1976.

[31] J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244, 1963.

[32] A. P. Witkin. Scale-space filtering. In *Proc. Eighth International Joint Conference on Artificial Intelligence*, volume 2, pages 945–951, 1983.