

Segmenting memory colours

Clément Fredembach, Francisco Estrada, and Sabine Sússtrunk

School of Computer and Communication Sciences, Ecole Polytechnique Federale de Lausanne (EPFL), Lausanne, Switzerland

{clement.fredembach, francisco.estrada, sabine.susstrunk}@epfl.ch

Abstract

Memory colours refer to the colour of specific image classes that have the essential attribute of being perceived in a consistent manner by human observers. In colour correction -or rendering- tasks, this consistency implies that they have to be faithfully reproduced; their importance, in that respect, is greater than other regions in an image.

Before these regions can be properly addressed, one must in general detect them. There are various schemes and attributes to do so, but the preferred method remains to segment the images into meaningful regions, a task for which many algorithms exist. Memory colours' regions are not, however, similar in their attributes. Significant variations in shape, size, and texture do exist. As such, it is unclear whether a single algorithm is the most adapted for all of these classes.

In this work, we concern ourselves with three memory colours: blue sky, green vegetation, and skin tones. Using a large database of real-world images, we (randomly) select and manually segment 900 images that contain one of the three memory colours. The same images are then automatically segmented with four classical algorithms. Using class-specific Eigenregions, we are able to provide insights into the underlying structures of the considered classes and class-specific features that can be used to improve classification's accuracy. Finally, we propose a distance measure that effectively results in determining how well is an algorithm is adapted to segment a given class.

Introduction

All image regions are not created equal; indeed, for digital photography some classes have a much greater importance than others. Some of the most important classes are the so-called memory colours: blue sky, green vegetation and skin tones [10]. It has been shown that human observers locate these classes in very specific areas of the colour gamut [2, 20]. Thus, many colour rendering and correction algorithms specifically try to map these colours to the correct values, such that the resulting images can be rendered. As a result, detecting these regions has been (and still is) a very active area of research.

Detection algorithms generally rely on many different features to classify memory colours: approaches include the use of shape, size, position, colour, and texture [4, 21, 13, 3]. Prior to being detected, however, images have to be segmented into meaningful regions. How meaningful a region is depends on the intended application of the segmentation, but most segmentation evaluation methods are predicated on the idea that all regions are of equal importance. As such, it is in their entirety that the resulting segmentations are compared to manually segmented images [19].

This work addresses the problem of class-specific segmenta-

tion evaluation, as well as the localisation of memory colour regions within natural images. Our framework builds on the eigenregions proposed by Fredembach et al. [11], which are principal component analysis (PCA) based geometrical features that encompass shape, size, and position information of regions. The central idea is to calculate class-specific eigenregions, i.e., obtaining different geometrical descriptors for each class. To be used in such a manner, the considered classes have to be reasonably localised across images, i.e., they should usually be found in similar position within images. The classes we consider here: blue sky, green vegetation, and skin tones generally fulfil, due to physics or photographic composition, this localisation criterion.

An objective ground truth for our experiments is obtained by manually segmenting 900 images, 300 per class. These accurate binary segmentation maps are used to calculate class-specific eigenregions that are subsequently compared to the ones resulting from automatic segmentation of the same images. Four segmentation algorithms that employ very different information are compared: Meanshift (density estimation process) [5], Felzenszwalb and Huttenlocher (minimum spanning trees) [8], k-means (Euclidian distances between clusters) [1], and edgeflow (Gabor filter banks) [14].

The comparison is based on the idea that if human segmentation is available for a given class, then its N eigenregions provide a reference basis in N -dimension. An algorithm-based segmentation of the same data will, however, provide a *different* basis in N -dimension. Measuring the distance between these bases effectively quantifies the performance of the algorithm.

The results show a strong class-dependency in both the accuracy of segmentation and shape of the eigenregions. In fact, the proposed framework can be used to quantify, for a given class, the distance between any two algorithms and the influence of the algorithm's parameters. In addition, it yields class-specific features that can be used for classification tasks.

Background

Segmentation-wise assessment of class-specific data is scarce. In a more global setting, however, assessing the performance of automatic segmentation is not a new concern and several approaches have been presented that yield a measure of "closeness" or "agreement" with human segmentation. Martin et al. [17] first proposed the use of region consistency over a database of human-segmented images [16] to evaluate the performance of automatic segmentation algorithms. These measures of segmentation consistency turned out to be biased toward over- or under-segmentation, so in [15] the use of precision and recall on region boundaries was suggested instead. A benchmark of several segmentation algorithms based on precision and recall was published in [6]. A different, region-based consistency measure

was presented by Ge et al. in [12]. Their measure also depends on the overlap between automatic and human segmentations, but it was computed on images that contained only two regions: a salient object and its background. Overlap was measured after deciding (based on the human segmentation) which subset of regions in the automatic segmentation best matched any given human region. More recently, Unnikrishnan et al. [19] presented a benchmark based on the Normalized Probabilistic Rand index. This measure compares segmentations through a soft weighting of pixel pairs that depends on the variability of the ground truth data. Other measures of segmentation consistency have been proposed in [9], [18], and [7]. A concise survey of these measures is provided in [19].

Despite their potential usefulness, each of the above methods for evaluation has its own limitations. First of all, they are global methods designed for entire image agreement, we are here concerned about specific classes. Boundary based methods will give good scores to under-segmented images in which two or more distinct (and possibly large) image regions are connected through narrow “leaks”. Since most of the boundary is recovered, boundary matching may falsely indicate that the segmentation is accurate. Methods based on overlap such as Ge et al. can be biased toward high scores by over-segmenting. In addition, this method assumes some form of expert is available to decide which of the over-segmented regions should be merged together to match human segmentation. The benchmark by Unnikrishnan et al. [19] provides interesting insights about the performance of segmentation methods on natural images, however, the question remains of whether particular algorithms are better for specific segmentation tasks, which is one of the fundamental problems addressed in this paper.

Eigenregions

Eigenregions were first proposed in [11] as PCA-based features for image classification. They were obtained by first segmenting a great number of images into regions whose “coverage” was assessed. Working on region coverage allows eigenregions to encompass geometrical attributes, such as shape, size, and position. For the analysis to be tractable, the segmentations are performed on reduced-size images, which is not a concern since downsampling does not alter a region’s location or coverage. An illustration of this downsampling procedure is shown in Fig. 1.



Figure 1. Left: an image from our database with a blue sky region; middle: a binary representation of the sky region’s coverage in the original image size; right: the downsampling of the binary image to 6×8 pixels which is used to perform PCA. The grayscale values represent the relative coverage of the region at a given location: from 0% (black) to 100% (white)

Let I be an input image of size $n \times m$, R be a region of I and p a pixel in the image. For every region R , we have that:

$$\forall p \in I: I(p) = 1 \text{ if } p \in R; 0 \text{ otherwise} \quad (1)$$

Let (i, j) be the index of a pixel in the reduced-size image I_d and let $d_1 = \frac{n}{n_d}$ and $d_2 = \frac{m}{m_d}$ be the downsampling factors along the rows and columns of I , respectively. I and I_d are related by:

$$I_d(i, j) = \frac{1}{d_1 d_2} \sum_{k=1}^{d_1} \sum_{l=1}^{d_2} I(d_1(i-1) + k, d_2(j-1) + l) \quad (2)$$

The pixel (i, j) of I_d is assigned the value of the proportion of white pixels contained within the corresponding $d_1 \times d_2$ sized block in the original binary image.

These downsampled images are the input to the PCA algorithm. In effect, each one is a N -dimension feature vector, where $N = n_d m_d$. If there are, for a given class, M regions in the database, then X is the input data matrix ($N \times M$) and we can write:

$$\bar{X} = \mu(X) : \text{the mean of } X \quad (3)$$

$$Y = X - \bar{X} \quad (4)$$

$$C = Y Y^T \quad (5)$$

C can then be expressed, using singular value decomposition, as: $C = V \Lambda V^T$, where V is the eigenvector matrix and Λ is the diagonal eigenvalue matrix of C . The Principal Component Analysis, in fact a translation and a rotation in the N -D space, finds an orthogonal basis that is optimal in a least square sense (i.e., it minimises the data projection’s distance onto the new basis). Moreover, the eigenvector that corresponds to the largest eigenvalue is in the direction of the largest variance [22].

The two important elements are the eigenvector and eigenvalues matrices: V and Λ . V defines the new basis vectors, i.e., their orientation, while Λ expresses the relative importance of each basis vector in reconstructing the data. The key insight is that if we are provided with a reference basis, we can calculate its similarity (i.e., distance) to any other basis in a space of identical dimension. In our framework, the reference basis is the eigenregions obtained by human segmentation, while the candidate bases are the eigenregions obtained via automatic segmentation algorithms.

We first note that if two vectors have many common components (that is, two regions’ coverage is almost identical), the angle they form is going to be small, i.e., the points they define will be close in space. This property is important since it guarantees that regions that are roughly similar will be located close to one another. Conversely, over- and under-segmented regions will be located much further apart since the number of components they have in common with an exactly segmented region is going to be small.

Let V_i^1 be the i^{th} eigenvector of a reference segmentation and V_i^2 be the i^{th} eigenvector of a candidate segmentation. Furthermore, let λ_i^1 and λ_i^2 be the eigenvalues associated with V_i^1 and V_i^2 , respectively. We can express the angle between the two vectors as:

$$\theta(V_i^1, V_i^2) = \cos^{-1}(\langle V_i^1, V_i^2 \rangle) \quad (6)$$

that is, the inverse cosine of the vectors’ inner product. Since we are working with PCA, orientation matters but direction does not, therefore we can further write:

$$\theta(V_i^1, V_i^2) = \min(\theta(V_i^1, V_i^2), 180 - \theta(V_i^1, V_i^2)) \quad (7)$$

where θ is expressed in degrees.

The distance between a reference segmentation method V^1 and a candidate one V^2 can then be defined as the weighted sum of each angle, i.e.,

$$\Delta(V^1, V^2) = \sum_i \lambda_i^1 \theta(V_i^1, V_i^2) \quad (8)$$

The eigenvalues from the reference method are used as weights because they express the importance of a given orientation in the human segmentation (the one an algorithm aims to be close to), and thus the importance of committing an error there. This weighting will have the effect of “denoising” the results, only preserving errors that are relevant to the reference segmentation.

Given a reference basis, the proposed distance measure, Equation (8), is effectively class, algorithm, and parameter independent since it only measures the dissimilarity of two bases in N -D space. It can thus be used to compare the accuracy of different segmentation algorithms using different parameters, and it can also indicate the relative “difficulty” of segmenting a class compared to others, as shown in the next section.

In [11], it was proposed that eigenregions were independent of the segmentation algorithm, and so were the underlying features. While we do not contest this, we point out that this argument was made in light of *general* regions, i.e., all regions were considered equal and were used. We argue, however, that most image classes have a much lower underlying dimensionality than general regions. As a result, their appearance in PCA space will vary significantly and, consequently, so will the outcomes of different segmentation methods.

Experimental Setup

The experimental protocol proceeds as follows: first, test images are selected from a database; these images are then segmented by hand according to chosen classes. In a third step, the images are segmented using several automatic algorithms and their output is assessed using a simple matching algorithm. Finally, once the data is collected, eigenregions are selected and distances measured.

The database we used consists of 55’000 real-world images. They come in various original formats and quality, and depict (almost) every possible scene. Out of these 55’000 images, 9’000 have been manually annotated by photographic experts as containing either one of the memory colours: blue sky, green vegetation and skin tones. We randomly selected 900 images (300 per class) out of these 9’000 for our experiment. Since this database was used to perform quick calculations on images, image size was reduced to 64×48 pixels. This downsampling does not, however, alter the location of regions within an image. Examples of images in this database are shown in Fig. 2.

These images were segmented by hand. For every image, the user is asked to segment *only* the relevant class, which leads to a binary segmentation of the image (see Fig. 3 for an illustration).

The 900 images are also segmented using four different algorithms: k-means (with $k=8$), edgeflow (with $\sigma = 8$), FH (with $k=50$) and meanshift (with $\text{spatial}=6$ and $\text{range}=15$). For the first two algorithms, the parameters were chosen to match the ones from [11], while the latter two were chosen empirically by looking at the average size of regions and number of regions per im-

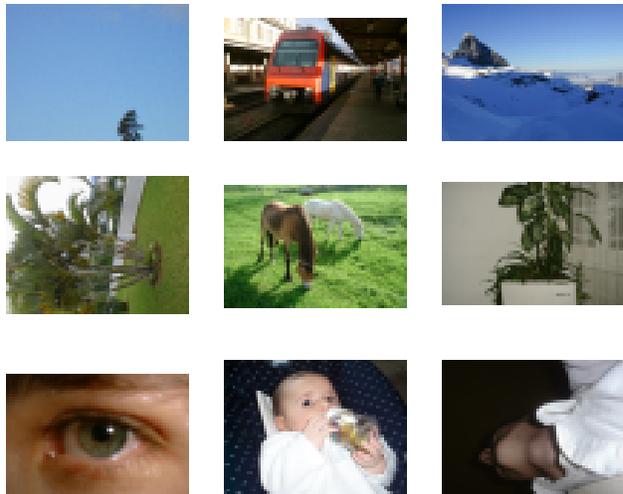


Figure 2. Example of images present in the database: blue-sky labelled (1st row), vegetation labelled (2nd row) and skin-tones labelled (3rd row), note the variety of content.

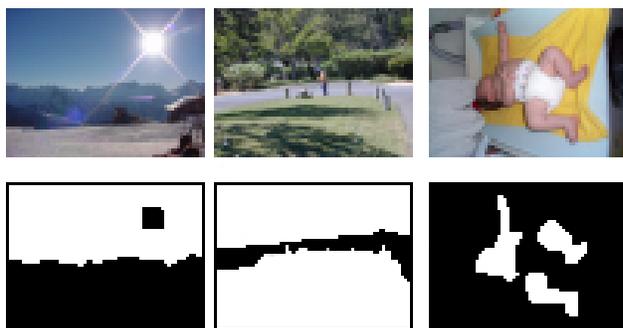


Figure 3. Human binary segmentations examples of the three considered classes. The original images containing sky, vegetation and skin (top row) and their segmentation (bottom row).

age. All 900 images were segmented using the exact same parameters, which we have not attempted to optimise.

To assess the segmentation results, we look at every region of the segmented image. If a region has a non-null intersection with the human segmentation, i.e., if a segmented region *contains* a given class, this region is deemed a positive match. A binary map is thus created where the region will appear in white and the rest of the image in black (akin to the ones shown in Fig. 1 and 3). After all the binary segmentations are obtained, they are reduced to a 6×8 image, according to Eqns. (1) and (2). From these output images, 15 sets of eigenregions are calculated: one for each algorithm-class pair.

Results

The results are reported in three categories. First, we assess the “localisability” of the memory colour classes. Indeed, if a class is not localised at all within images, a PCA-based framework will be of little help. The reconstruction rates, however, indicate that memory colour classes are in fact well localised. In a

second step, we show the class-specific eigenregions for the three considered classes obtained by manual segmentation and the four segmentation algorithms, and discuss these results in terms of image content and segmentation behaviour. Finally, we use our proposed distance measure to numerically quantify the distance of each of the algorithms to our ideal, manually segmented, images. The resulting distance agrees with both the shape of class-specific eigenregions, as well as with the opinion of the observers who looked at the human and automatic segmentations.

Localised classes

Figure 4 shows how localised our three considered classes are. For blue sky and green vegetation, 5 eigenregions (i.e., 10% of the available eigenvalues) suffice to explain 85% of the variance. Considering the prevalence of these two classes in landscape images, these results are unsurprising. Conversely, skin tones are not as localised. Since skin tones encompass all of face, hands, arms, body, etc., they are expected to be inherently less localised than sky or grass.

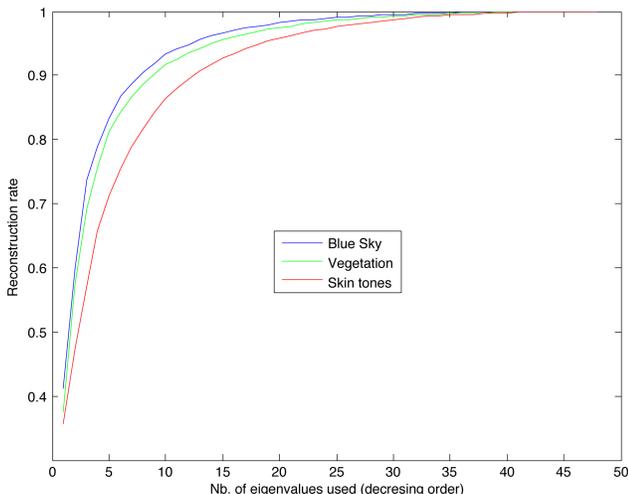


Figure 4. Reconstruction rates for human segmentation. Blue sky and green vegetation are fairly well localised, with 85% of the variance explained by 10% of the eigenregions, while the skin tones reconstruction rate is lower.

Reconstruction rates are important because they indicate whether it is judicious to use geometrical features for the detection of a given class. On the other hand, they do not provide a measure of accuracy. A segmentation algorithm that would deterministically partition images into two regions (say top and bottom) would have a very high reconstruction rate. It would, however, be a very inaccurate segmenter.

Class-specific eigenregions

After eigenvalues, we analyse the eigenregions given by the algorithms on our three classes. The first five eigenregions for each class and each algorithm are shown in Fig. 5-7, where their values have been normalised between 1 (white) and -1 (black) for better visualisation. These eigenregions provide important clues regarding the performance of a given segmentation algorithm over a class. First, they allow a visual comparison of class-localisation and differences across algorithms. Then, as pointed out in [11],

they can be used as features in image classification; the rationale is that our particular eigenregions should actually prove more useful than the general eigenregions since ours are readily tailored to a specific class. Classification in itself was not performed here, as it is outside the scope of this paper.

From the results, we observe that sky and vegetation eigenregions appear more coherent than the skin ones. This correlates well with the reconstruction curves shown in Fig. 4 and is easily explained by the fact that sky and vegetation are mostly found in landscape-type images that have a top/down decomposition (or left/right for pictures taken in a portrait orientation). These regions are therefore located in a smaller part of the 48-D space and thus are easier to cluster via PCA.

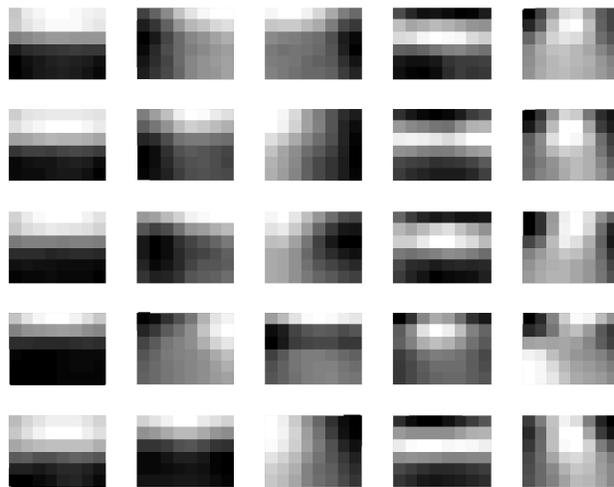


Figure 5. The first five eigenregions for the blue sky class. From top to bottom: Human segmentation, k-means, edgeflow, FH and meanshift.

The blue sky results, Fig. 5, show that while all algorithms correctly find the first eigenregions, k-means and edgeflow results appear, in general, much closer to the human segmentation than either FH or meanshift when looking at eigenregions 2-5. In general, however, the eigenregions correspond to our expectations, with clear top/down decomposition for the first one, and some variation in the subsequent ones that originate from images where the sky is partly occluded (trees, buildings, people).

Vegetation eigenregions, Fig. 6, start similarly with a landscape-type decomposition (top/down or left/right, depending on the camera's orientation), but this behaviour changes after the first three to indicate the presence of centred objects, e.g., trees or plants in indoor scenes. These latter positions are harder to accurately segment and few algorithms are able to correctly distinguish them. Both meanshift and edgeflow appear to be closer to the ground truth, but their results are still somewhat skewed. K-means performs well on the landscape-type images but is confounded by more complex scenes, while FH misses out one of the first eigenregion.

Finally, skin tones eigenregions, Fig. 7, exhibit various type of centre-surround interactions, i.e., the object of interest is small and located centrally within an image. Since we look for skin tones in general, as opposed to faces only, we expect the results to be somewhat noisy because of the greater location possibili-

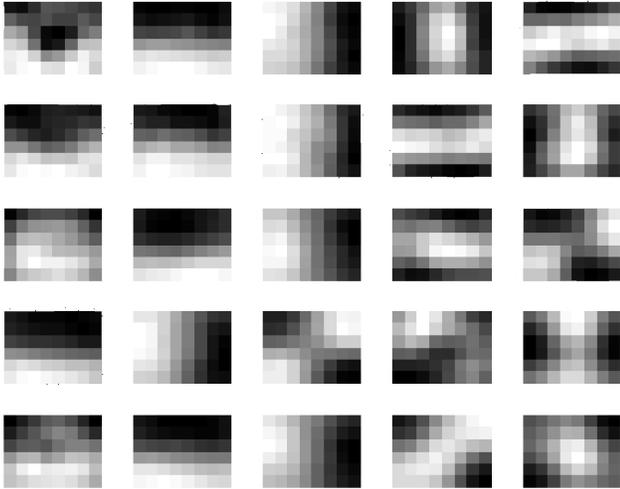


Figure 6. The first five eigenregions for the green vegetation class. From top to bottom: Human segmentation, k-means, edgeflow, FH and meanshift.

ties. The eigenregions express two aspects well: the position and the scale ambiguity. Indeed, while most of them are of centre-surround types, the size and the location of the “interest region” varies across eigenregions. Looking at the four algorithms, we see that meanshift is probably the closest to human segmentation while k-means is not too far behind. Edgeflow and FH appear to perform worse but for different reasons. In fact, their behaviour is complementary with FH not detecting the larger regions (eigenregion 1) and edgeflow wrongly detecting the smaller ones.

Looking at these results, we can draw the following conclusions: class-specific eigenregions have very distinct shapes that express the content of the images well; they are algorithm-dependent, and the closest algorithm to human segmentation does not appear to always be the same. Finally, the shape of the eigenregions correlates well with the reconstruction rates observed, i.e., the simpler the shape of the eigenregion, the better localised the underlying class is.

Which algorithm for which class?

The eigenregions themselves give useful information, still, assessing distance in a 48-D space, even when provided with visual cues, is difficult. Using our proposed measure (8), we measure the distance between the four algorithms and human segmentation for each class. The results, reported in Table 1, confirm what was visually inferred in the previous section. Looking at the distances as a whole, one sees that blue sky is the best segmented region (smallest distance), followed by vegetation and skin tones. This is expected given the much greater variety of position, size, shape, and colour of skin tones when compared to blue sky or vegetation, thus making them harder to segment. Also, we see that while *on average* meanshift performs better than the other algorithms, it is not necessarily the best performing one for *every* class.

Taking the results separately, one observes that k-means and edgeflow are equivalent in their sky segmentation, meanshift and edgeflow are better for vegetation, and meanshift is best for skin tones; these results correlate well with the visual assessment done

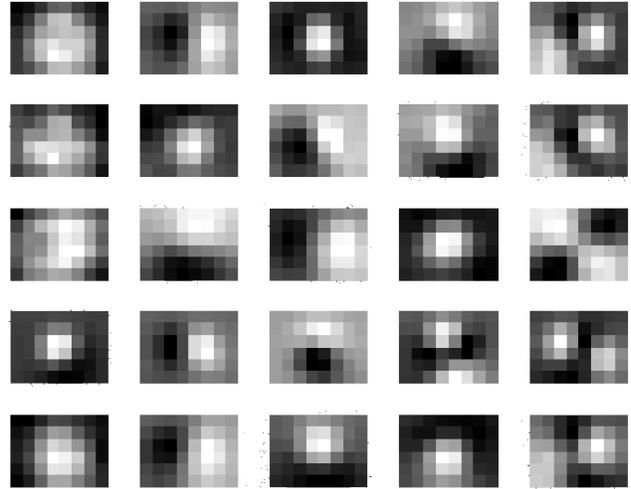


Figure 7. The first five eigenregions for the skin tones class. From top to bottom: Human segmentation, k-means, edgeflow, FH and meanshift.

| | Blue Sky | Green Vegetation | Skin Tones |
|-----------|-------------|------------------|-------------|
| k-means | 24.3 | 34.65 | 44.15 |
| edgeflow | 23.6 | 28.53 | 56 |
| FH | 40.62 | 73.2 | 70.8 |
| meanshift | 36.12 | 27.5 | 33.6 |

Distance between a given algorithm and human segmentation (smaller is better). The results are highly class-dependent and there is not a best algorithm overall.

in the previous section. For all classes, FH is rated as the worst performing algorithm. This comparison brings several questions that have to be answered: why is k-means so good, why is meanshift worse in the simplest class, and why does FH perform so badly?

K-means’ performance can be explained by the choice of classes. Indeed, memory colours are classes that are well located in colour space [2], so a cluster including them will usually be found. As a result, k-means is expected to be accurate. Its performance for vegetation and skin is, however, lower since these classes’ luminance and colour can be altered by lighting effects (such as shading), thus creating errors.

Edgeflow includes both colour and texture information, and is therefore expected to yield a good segmentation of our three classes. However, its accuracy for skin tones is not always high. Looking at both the distance and the eigenregions themselves, one sees that its regions are larger than they should be. This is, most likely, the consequence of the choice of σ that influences the scale at which variations are sought. Additionally, artefacts such as glasses, hats, occlusions or sometimes hair can induce a wrong segmentation.

Perhaps surprisingly, FH is the worst rated algorithm in our test, and this for all classes. The reason here is that the choice of parameters has given rise to chronic over-segmentation. While the number of regions in the image is not overly high (between 14 and 28 regions per image), it was very sensitive to noise, vignetting, and small level texture alterations. This is confirmed by looking

at the number of regions found for each class. For sky, vegetation and skin, FH has, on average, 3.2, 5.2, and 2.6 regions per image, respectively, compared to meanshift's 1.2, 2.3 and 1.5, indicating a strong over-segmentation issue. We have found no significant correlation between number of regions and performance for other algorithms than FH, whose over-segmentation was significant.

Finally, meanshift, the best overall algorithm, exhibits a rather unique behaviour: its worst class is sky, which is the opposite of every other algorithm. Again, this can be explained by the parameters used. While they were well-suited to vegetation and skin, they tend to under-segment sky, especially in the presence of softer gradients, such as clouds or haze.

Note that we do not advocate here that one algorithm is better than the others. Rather, the results show that a given algorithm (or a given choice of parameters) appears to be measurably better for segmenting a specific class, not all classes in general. It is therefore well possible that meanshift, with other parameters, would have a more accurate sky segmentation. However, this could be detrimental to its segmenting of skin or vegetation. A direct consequence of our results is that the proposed distance measure can not only be used to select one algorithm, but also to optimise a given algorithm's parameters in order to segment a specific class.

Conclusions/Future work

We have presented an eigenregion-based framework that evaluates class-specific image information. Using human segmentation and assessment of automatic segmentation algorithms, we were able to show, numerically, that naturally occurring classes in images were neither evenly distributed nor similarly localised. We argued that class-specific eigenregions can prove more accurate than general eigenregions in classification, since they better explain the observed behaviour of the considered classes. The classes we considered here were sky, vegetation, and skin tones memory colours, all of them being of critical importance for tasks such as colour rendering or correction.

Moreover, we have proposed a distance measure in N -D space that takes into account the relative weight of a given eigenregion. Using that distance, we showed that different algorithms segment different image classes with varying accuracy compared to human segmentation. Importantly, the algorithms' performance is strongly class-dependent, there is no single best algorithm. Future work involves using our distance measure to obtain optimal parameters for each segmentation algorithms, thus having specific settings for specific classes.

References

- [1] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.
- [2] P. Bodrogi and T. Tarczali. Colour memory for various sky skin and plant colours: effect of the image context. *COLOR research and application*, 25(4):278–289, 2000.
- [3] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE PAMI*, 24(8):1026–1038, August 2002.
- [4] J. Chen, T. Pappas, A. Mojsilovic, and B. Rogowitz. Adaptive image segmentation based on color and texture. In *IEEE 2002 International Conference on Image Processing (ICIP2002)*, pages 122–126, 2002.
- [5] D. Comaniciu and P. Meer. A robust approach toward feature space analysis. *IEEE Trans. on PAMI*, 24(5):603–619, 2002.
- [6] F. Estrada and A. Jepson. Quantitative evaluation of a novel image segmentation algorithm. In *CVPR*, pages 1132–1139, 2005.
- [7] M. R. Everingham, H. Muller, and B. Thomas. Evaluating image segmentation algorithms using the pareto front. In *ECCV*, pages 34–48, 2002.
- [8] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.
- [9] C. Fowlkes, D. Martin, and J. Malik. Learning affinity functions for image segmentation. In *CVPR*, volume 2, pages 54–61, 2003.
- [10] C. Fredembach, M. Schroeder, and S. Susstrunk. Region-based image classification for automatic color correction. In *Proceedings the 11th IS & T color imaging conference*, pages 59–65, 2003.
- [11] C. Fredembach, M. Schroeder, and S. Susstrunk. Eigenregions for image classification. *IEEE Trans. on PAMI*, 26(12):1645–1649, 2004.
- [12] F. Ge, S. Wang, and T. Liu. Image segmentation evaluation from the perspective of salient object extraction. In *CVPR*, volume 1, pages 1146–1153, 2006.
- [13] A. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall International, 1989.
- [14] W. Ma and B. Manjunath. Edgeflow: a technique for boundary detection and image segmentation. *IEEE Trans. on IP*, 9(10):1375–1388, 2000.
- [15] D. Martin. An empirical approach to grouping and segmentation, 2002. Ph.D. thesis, U.C. Berkeley.
- [16] D. Martin and C. Fowlkes. The Berkeley Segmentation Database and benchmark, 2001. <http://www.cs.berkeley.edu/projects/vision/grouping/seg>.
- [17] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, pages 416–425, 2001.
- [18] M. Meila. Comparing clusterings by the variation of information. In *Proc. Intl. Conf. on Learning Theory*, pages 173–187, 2003.
- [19] R. Unnikrishnan, C. Pantofaru, and M. Hebert. Toward objective evaluation of image segmentation algorithms. *IEEE Trans. on PAMI*, 29(6):929–944, 2007.
- [20] J. Perez-Carpinell, M. D. de Fez, R. Baldovi, and J. C. Soriano. Familiar objects and memory color. *COLOR research and application*, 23(6):416–427, 1998.
- [21] J. Da Rugna and H. Konik. Color coarse segmentation and regions selection for similar image retrieval. In *CGIV 2002: The First European Conference on Colour in Graphics, Image and Vision*, pages 241–244, 2002.
- [22] A. Webb. *Statistical Pattern Recognition*. Arnold, 1999.