Semantic-Improved Color Imaging Applications: It Is All About Context

Albrecht Lindner and Sabine Süsstrunk, *Senior Member IEEE*

***Abstract*—Multimedia data with associated semantics is omnipresent in today's social online platforms in the form of keywords, user comments and so forth. This article presents a statistical framework designed to infer knowledge in the imaging domain from the semantic domain. Note that this is the reverse direction of common computer vision applications. The framework relates keywords to image characteristics using a statistical significance test. It scales to millions of images and hundreds of thousands of keywords. We demonstrate the usefulness of the statistical framework with three color imaging applications. 1) semantic image enhancement: re-render an image in order to adapt it to its semantic context 2) color naming: find the color triplet for a given color name 3) color palettes: find a palette of colors that best represents a given arbitrary semantic context and that satisfies established harmony constraints.**

***Index Terms*—semantic gap, image processing, semantics, enhancement, color naming, color palette**

I. Introduction

THE semantic gap is a major challenge to solve in the multimedia community. The *gap* is often understood as the difficulty to automatically infer semantic information from the multimedia domain such as in face recognition or video classification. In this article we investigate the gap in the reverse direction, i.e. we are interested in applications that have the semantic domain as a source and the multimedia domain as a target.

Bridging the gap in reverse direction (see Fig. 1) might seem counterintuitive at first sight because the classic forward direction is an ubiquitous problem in our daily digital life: we want computers to gain a semantic understanding of digital content in order to ease search, classification and so forth. However, today's social community websites often have plenty of user generated semantic information already attached to multimedia content such as tagged images on Flickr or user comments for posted content on Facebook. It is thus reasonable to explore applications that take existing semantic information as input and infer meaningful information and actions in the image domain.

Many aspects of the research in this article are similar to classic computer vision. We use large databases of annotated photos and characterize the photos with image descriptors. We then learn relations between the keywords and the image descriptors. However, our work differs in two key aspects: the type of descriptors used to describe the images and the mapping function to relate the two domains.

Our image descriptors, which represent our output domain, are chosen so that they are meaningful to a human observer. Consequently, descriptors that are popular in classic computer vision such as SIFT or HoG cannot be used. Instead we use simple color histograms or Fourier domain histograms because they characterize visible aspects in the image domain, i.e. colors and sharpness, respectively. As mapping function we can use a simple and very scalable statistical framework capable of associating numeric characteristics to semantic content, because we do not have to discriminate similar semantic concepts.

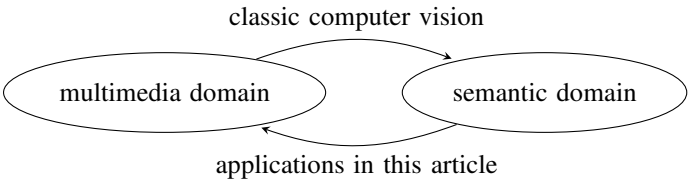


Fig. 1. Classic computer vision aims at inferring semantic information from the multimedia domain, e.g. image classification or face recognition. In this article we investigate the reverse direction, i.e. using the semantic domain to infer meaningful actions and information in the multimedia domain. This is similar as we operate on the same domains, yet different as source and target domains are swapped.

The statistical framework is based on the Mann-Whitney-Wilcoxon (MWW) ranksum test that assess whether the values in one set are significantly larger or smaller than the ones in another set. The test is non-parametric and thus does not require assumptions about the input values' distribution, which is important to guarantee usability for versatile inputs. The MWW test is also considerably robust, because it uses ranks instead of the absolute values to compute the test statistic. Finally, it can be implemented very efficiently, so that it can handle millions of images and hundreds of thousands of keywords on a single-processor machine.

The first application we present is semantic image enhancement: given an input image and a semantic expression the image is re-rendered in order to optimize it for the given semantic concept (Sec. IV). We implement two types of enhancements, which are a color tone mapping and a depth-of-field adaptation. Adapting an image to a specific semantic context usually requires a manual interaction and a photo editing tool, but our approach realizes it fully automatically leveraging a large database of annotated images.

The second application is color naming: given a color name, estimate its color triplet (Sec. V). This is usually done in psychophysical experiments where users have to manually name colors. Our automatic approach allows us to estimate color values for over 9000 color names in 10 different European and Asian languages. As there is no user interaction required we can even estimate color names for languages that we do

Both authors conducted this research at EPFL, Switzerland. Mail contact: albrecht.lindner@alumni.epfl.ch sabine.susstrunk@epfl.ch

not speak ourselves. The results are published in an online color thesaurus: www.colorthesaurus.com.

The third application is color palette extraction: given a semantic expression, extract a corresponding palette of five harmonic colors (Sec. VI). Color palette creation usually requires user interaction such as on Adobe Kuler, one of the web's most popular color palette tools [1]. The scalability of our framework allows us to pre-compute associations for a large vocabulary of 100,000 distinct English words and then propose color palettes fully automatically. The palettes can be explored online: www.koloro.org.

We further demonstrate in Section VII that among several alternatives the Mann-Whitney-Wilcoxon test is the optimal choice for the statistical framework in terms of both computational complexity and accuracy . The low computational complexity is essential to scale the methods to a large unrestricted vocabulary as realized for the color palette application.

II. Related Work

Our work is inspired by and related to research in the areas of computer vision, image mining, image enhancement and color naming. In this section we give a brief overview of other relevant work and put it into perspective with ours.

A lot of research in the the overlapping fields of imaging and semantics focusses on inferring semantic information from an image. Examples are automatic image classification [2], [3] or automatic labeling of objects in images [4], [5]. Our work also lies in the fields of semantics and imaging, but differs in one fundamental way. Instead of building systems that take images at the input and provide inferred semantic information at the output we work in the opposite direction: we consider semantic information as the input and automatically infer knowledge and actions in the image domain (see Fig. 1).

Deselaers and Ferrari [6] show that images with semantically similar annotations also have similar visual attributes and vice versa. This suggests that it is possible to enhance an image according to its semantic content by altering relevant image characteristics based on its annotated keywords. This is demonstrated in Section IV leveraging the MIR Flickr database [7] containing 1 million annotated Flickr images.

Automatic image enhancement is a widely explored topic and several approaches are being explored by the community. One approach is to enhance an image based on one or several example images such as Reinhard et al.'s color transfer [8] or Kang et al's personalized image enhancement [9]. Our enhancement also uses example images, but we select them according to keywords and estimate significant characteristics for those keywords. Another approach is to classify an image and then apply an algorithm explicitly designed for that class. Examples are Moser and Schröder [10] or Ciocca et al. [11], but the authors limit to 7 and 3 classes, respectively, because the different algorithms have to be manually implemented one by one. In our case we can regard each keyword as a class and the optimal processing is automatically derived from the example database; we thus are able to handle thousands of classes in one generic workflow.

A recent work in semantic image editing is PixelTone [12]. The authors propose a system where users can edit images with touch input and spoken commands such as *"sharpen the mid-tones at the top"*. Recognized commands such as *sharpen mid-tones* are mapped to the appropriate operation that is applied globally or locally in the relevant region that is determined by a keyword such as *top* or by the user's touch input. The main difference between this project and our proposed semantic image enhancement framework (Sec. IV) is that PixelTone requires the user to explicitly verbalize the desired image processing operation. In contrast, our aim is to exploit any semantic information that comes along with images such as keywords, image titles or community comments. This semantic information is originally not meant to be used for image enhancement, but harnessing it is a promising approach due to its omnipresence in social networks and photo sharing platforms.

Annotated image data can also be used to associate words with colors. An early and famous work is from Jones and Rehg where the authors propose a statistical model to find skin tones from annotated images [13]. Other related work proposes systems to find appropriate colors for given song lyrics [14] or using a PLSA model to find 24 English color names from annotated images [15]. Our color naming experiment presented in Section V is similar, but of considerably larger scale and incorporates more than 9000 color names from 10 different European and Asian languages. Also we discuss in Section VII what statistical measures give the best results in terms of speed and accuracy.

Instead of a single color, it is also possible to extract a color palette for a given topic. There are numerous web services that help artists and designers to find color palettes for a given topic [1], [16], [17], [18]. Adobe Kuler [1] is the best known platform. The users can create palettes either manually or extract them automatically from an uploaded image. The palettes can also be annotated with keywords so that other users can query for them in the database. Color Hunter [16] provides a service where users can type in a keyword that is used to query related images from Flickr. The software then extracts one palette per image and returns them to the user. Our approach to find color palettes based on semantic expressions (see Sec. VI) differs from the above solutions in the sense that we extract palettes using millions of annotated images fully automatically. The reliance on many more than just one image or keyword makes our approach more robust to annotation errors.

III. Statistical Framework

A. Mathematical Background

In the following we describe the statistical framework to compute links between image characteristics and keywords. A characteristic is a property of an image such as the percentage of pixels with a specific color, the brightness in a specific spatial region or the power of a specific spatial frequency in Fourier domain. This mathematical background is about the general concept and thus does not further precise the term *image characteristic*. Practical examples with real image characteristics are reproduced in Figures 2 and 3.

We denote I_{db} a database of image annotation pairs $(I_i, A_i) \in I_{db}$, where an annotation is a set of keywords

$A_i = \{w_{i1}, w_{i2}, \ldots\}$. For any given keyword w the database can be split into two subsets $\mathcal{I}_w = \{I_i, w \in A_i\}$ and $\mathcal{I}_{\overline{w}} = \{I_i, w \notin A_i\}$, which contain all images annotated with w and all remaining images, respectively. Given c_i, a characteristic of image I_i, we can further define the two sets $\mathcal{C}_w = \{c_i, w \in A_i\}$ and $\mathcal{C}_{\overline{w}} = \{c_i, w \notin A_i\}$ containing all characteristics of the images in sets $\mathcal{I}_w$ and $\mathcal{I}_{\overline{w}}$, respectively.

To assess the influence of a keyword w on an image characteristic, the values in the two sets $\mathcal{C}_w$ and $\mathcal{C}_{\overline{w}}$ have to be compared. We use the Mann-Whitney-Wilcoxon (MWW) ranksum test, which is a statistical significance test that assess whether the elements in one set are significantly larger or smaller than the elements in another set [19], [20]. The test statistic T is the sum of the positional indexes of the sorted elements of the combined set $\mathcal{C}_w \cup \mathcal{C}_{\overline{w}}$.

We explain the computation of the MWW test statistic at the example of the sets $\mathcal{C}_w$ = {1.8, 0.9} and $\mathcal{C}_{\overline{w}}$ = {-0.3, 1.5, 0.8, 1.3, 0.2}:

1) Union of both sets:
   $\{1.8, 0.9, -0.3, 1.5, 0.8, 1.3, 0.2\}$
2) Sort in increasing order (rank indexes stacked on top):
   $\{\overset{1}{-0.3}, \overset{2}{0.2}, \overset{3}{0.8}, \overset{4}{0.9}, \overset{5}{1.3}, \overset{6}{1.5}, \overset{7}{1.8}\}$
3) Sum of ranks of first set:
   $T = 4 + 7 = 11$

Under the null hypothesis that both sets contain equally large values the expected mean and variance of the test statistic T are:

$$\mu_T = \frac{N_w(N_w + N_{\overline{w}} + 1)}{2} \tag{1a}$$

$$\sigma_T^2 = \frac{N_w N_{\overline{w}}(N_w + N_{\overline{w}} + 1)}{12} \tag{1b}$$

where N_w and $N_{\overline{w}}$ are the cardinalities of the sets $\mathcal{C}_w$ and $\mathcal{C}_{\overline{w}}$, respectively. The standardized significance score z for the above example is then:

$$z = \frac{T - \mu_T}{\sigma_T} \approx \frac{11 - 8}{6.67} \approx 0.45 \tag{2}$$

The positive z score indicates that the values from the first set are larger than those from the second set. However, the small absolute value indicates that the statistical significance in this example is low.

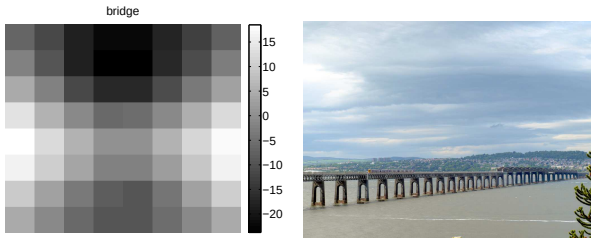
It is important to retain that the z score can be positive or negative. Its abolute value represents the statistical significance of a keyword's impact on an image characteristic. A positive (negative) sign indicates a presence (absence) of that characteristic given the keyword. The example in the next section illustrates this.

There are other statistical and non-statistical methods to assess the difference between the values of two sets. We give in Section VII an overview of alternative methods, a qualitative comparison between them and show that the MWW test yields the best results for the color naming application presented in Section V.

B. Example

We use the 1 million images from the MIR Flickr dataset [7] and compute histograms in CIELAB color space with $15 \times 15 \times 15$ equidistant bins along each axis. Each of the histogram bins counts as a characteristic j, which results in a significance distribution in color space. Figure 4(b) shows this distribution of significance scores for the keyword *periwinkle*. The three orthogonal planes intersect in the distribution's maximum, which is in the violet region of the color space as indicated by the ab color plane at the bottom. The maximum corresponds to the sRGB triplet 169, 167, 212.

The statistical framework not only handles colors, but any other characteristic that can be quantified numerically. We filter the images using Gabor filters with different orientations (horizontal, vertical, and diagonal) and average the filter outputs in each grid cell of a regular 8×8 spatial grid superposed on the image independent of its size or aspect ratio. This coarsely describes the spatial layout of structure in an image. Figure 2 shows the significance distribution for the keyword *bridge* and an example *bridge* image. It is visible that there is significantly more structure (positive z scores) in *bridge* images along a horizontal line in the lower third and on the very left and right hand sides of an image. In contrast, there is significantly less structure in the top middle part (negative z scores). More examples and other characteristics are shown in [21].



(a) significance distribution (b) example *bridge* image

Fig. 2. Left: Spatial layout of structure in images annotated with *bridge*. Structure is measured with Gabor filters and their coarse spatial location preserved due to an averaging in a regular 8×8 grid that is superposed on the image independent of its size or aspect ratio. As expected, *bridge* images have significantly more structure in the lower middle and the left and right sides (positive z scores), and significantly less structure in the top middle part (negative z scores). Right: Example *bridge* image, photo attribution: Joop van Dijk.

One can quantify the impact of a keyword on a characteristic with a simple difference between a distribution's maximum and minimum values. For the two example distributions we get:

$$\Delta z^{CIELABhistogram}_{periwinkle} = 8.7 - (-4.0) = 12.7$$

$$\Delta z^{Gaborfilterlayout}_{bridge} = 18.5 - (-23.8) = 42.3$$

Figure 3 shows Δz values for more characteristics and keywords. Note that we use low-level characteristics because our output is the imaging domain and we thus need features that can be visually presented to a human observer.

If the number of samples increases, the ranksum statistic yields higher z scores because there is more evidence of a trend than with fewer samples. This is not a shortcoming of the method, but a general property of all significance tests: the more samples the stronger the evidence. This is a useful property because an observed impact may be due to noise if a rare keyword is poorly represented in the database. We will

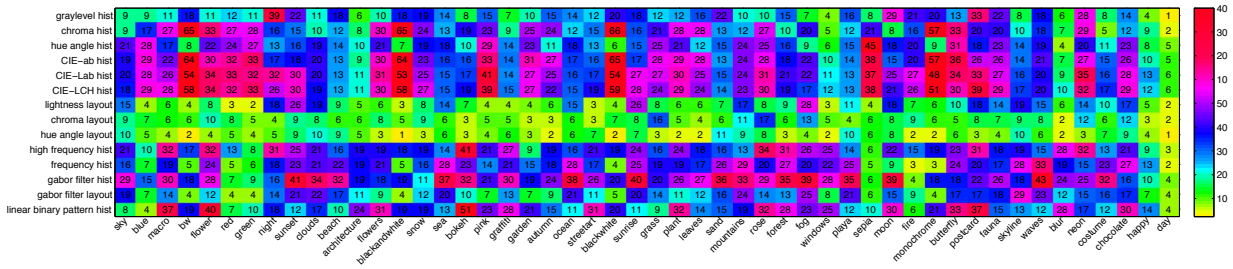


Fig. 3. Δz values for different combinations of characteristics and keywords. The histogram characteristics are computed from global histograms of a feature, e.g. a 1-dimensional histogram of chroma values (*chroma hist*) a 3-dimensional histogram of CIE Lab color values (*CIE-Lab hist*) or a 1-dimensional histogram of the image's graylevels after a high pass filter (*high frequency hist*). The layout characteristics use a 8×8 spatial layout of a specific as shown in Figure 2. The values show how significant a characteristic is for each keyword, e.g. color for *blackandwhite* or lightness and chroma layouts for *sunset* and *grass*, respectively.

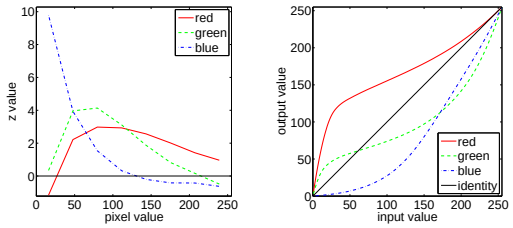
discuss alternative measures that do not depend on the number of samples in Section VII.

IV. Semantic Image Enhancement

The first application we present is semantic image enhancement, which aims at re-rendering an image to better adapt it to a semantic context. We define re-rendering as processing an image that has already been rendered in-camera or elsewhere. Figure 4(a) shows an example of semantic image enhancement for the semantic concept of *strawberry*.

A. Semantic Color Transfer

The method we present consists of two components that handle an image's semantic and visual contents, respectively. The semantic component uses the keyword and its z values to determine a processing that strengthens significant image characteristics. The image component then detects regions of the image where the processing has to be applied. Example source code and precomputed z scores for this application are available under http://ivrl.epfl.ch/Lindner_IEEE_MM_2015.



(a) z values (b) tone-mapping

Fig. 5. Left: significance values for *autmun* and the three color channels. Images annotated with *autumn* have significantly more pixels with a low blue content (positive z values) and significantly less pixels with a high blue content (negative z values). The red channel is the contrary. Right: derived tone mapping curves that decrease the overall blue content and increase the overall red content.

We use significance values from 8-bin histograms of the red, green, and blue channels in order to semantically enhance an image's colors. Figure 5(a) shows the significance values computed from 1 million images[1] for the keyword *autumn*. We can see that *autumn* images have significantly more pixels with a low blue content and a high red content (positive z values).

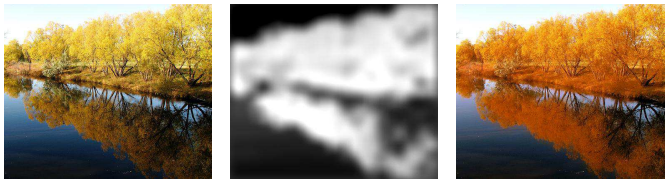
If the significance value of a histogram bin is positive (negative), the desired tone mapping curve has to increase (decrease) the number of pixels that fall into that bin. This is achieved with a varying slope m

$$m = \begin{cases} 1/(1+Sz) & \text{if } z \geq 0 \\ 1+S|z| & \text{if } z < 0 \end{cases} \tag{3}$$

where S is a scale parameter that can be chosen by the user to regulate the impact of the processing[2]. The slope is rather flat at a bin center where the z score is positive and thus accumulates more pixels into that bin. In contrast, a steeper slope at bin centers with negative z score decreases the numer of pixels in that bin.

We use Equation 3 to compute a slope value for all 8 bin centers of a channel. The slope values are linearly interpolated in between bin centers and then integrated to yield the tone mapping curve. Finally we scale the tone mapping curves of all channels to the interval [0 255] in order to preserve the image's white and black points.

Figure 5(b) shows the resulting tone mapping curves for the semantic concept of *autumn* and $S = 1$. These global tone mapping curves increase the red content of any given pixel and decrease its blue content; the green tone mapping curve is closer to the identity transform.

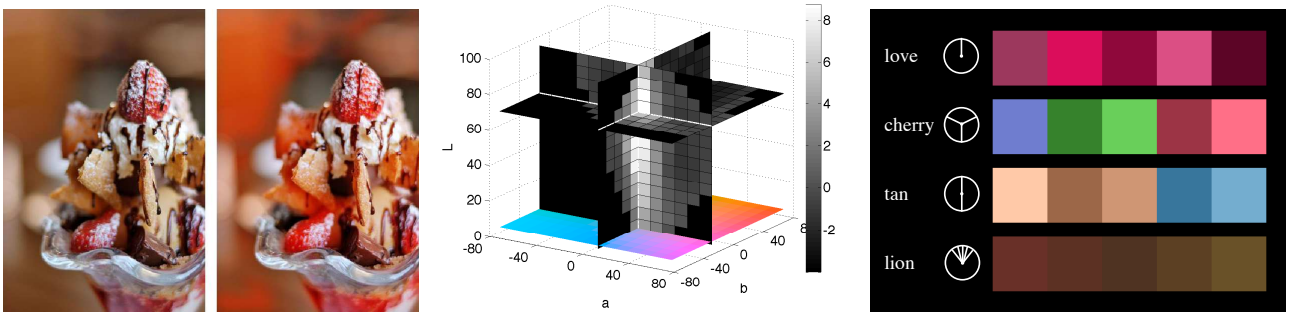


(a) input (b) weight map (c) re-rendered output

Fig. 6. Example image showing semantic image enhancement for the semantic concept of *autumn* and $S = 1$ (see Eq. 3). Left: input image. Middle: weight map indicating which regions are significantly relevant for the semantic concept of *autmun*. Bright regions are estimated as relevant and dark regions as not relevant. This map is used to weigh the influence of the semantic image processing according to Equation 5. Right: output image. Photo attribution: Nobodeez Business.

[1]Images are from the MIR Flickr database [7].

[2]We found $S = 1$ to be a good value and used it for all examples in this article.



(a) semantic recoloring: *strawberry* (b) color naming: *periwinkle* → 169, 167, 212 (c) color palette extraction

Fig. 4. Three applications for our large scale statistical framework. Left: An image is automatically re-rendered in order to emphasize an arbitrary semantic concept (*strawberry* in this example image). Photo attribution: openarms. Middle: Given an unconstrained list of arbitrary color names the framework can automatically estimate each color name's color value (*periwinkle* in this example). The three orthogonal planes show a heat map of the significance distribution and intersect at its maximum. Positive (negative) significance values indicate that this color is significantly present (absent) in images annotated with the related keyword. Right: Automatic extraction of color palettes for a given semantic expression and an optional hue template. Example for four keywords and hue templates are shown as indicated to the left of each palette.

The image component of the semantic image enhancement framework assures that the tone mapping is only applied locally in relevant image regions. This is important because tone-mapping by itself would alter all pixels independent ot their color and produce undesired color shifts in the non-relevant regions. In the example image in Figure 6(a) the relevant regions are the trees and their reflections in the water whereas the sky and blue water have to remain unaltered.

We compute for each pixel p a weight $\omega(p)$ that estimates the pixel's relevance for a given semantic concept

$$\omega(p) = \Big[g_\sigma * z_w\left(col(p)\right) \Big]_0^1 \quad \forall p \in \text{image plane} \qquad (4)$$

where g_σ is a small Gaussian blur with a sigma of 1% of the image diagonal, $z_w(\cdot)$ the significance value for a keyword w, $col(p)$ the color triplet at pixel position p and $[\cdot]_0^1$ a normalization operator that maps the 5% and 95% quantiles to the unit interval ($[q_{0.05}\ q_{0.95}] \rightarrow [0\ 1]$) and clips exceeding values to the borders.

Figure 6 shows the input image and the corresponding weight map ω for the keyword *autumn*. Bright regions have a weight close to 1 as they are estimated to be relevant for the enhancement, dark regions have a weight close to 0 and are estimated as not relevant.

We then produce an intermediary image I_{global} where the tone mapping is applied globally to each pixel. The final output image I_{out} is a weighted linear combination of the original input image I_{in} and the globally tone mapped image I_{global} that is computed at every pixel position independently:

$$I_{\text{out}} = (1-\omega)I_{\text{in}} + \omega I_{\text{global}} \qquad (5)$$

The output image I_{out} for the example image of this section is reproduced in Figure 6(c). Please note that tone-mapping operator successfully shifted the colors of the trees from yellowish to *autumn* in the areas where ω is large. The image's unrelated areas remain unaltered as ω is close to zero.

B. Beauty requires context

A key aspect of our semantic image enhancement is the observation that the pixel values alone are not sufficient to optimally enhance an image. The image's semantic context can give strong indications whether certain colors are required to better reproduce the artist's intent.

Figure 7 shows an example image where the same input image is enhanced for two different semantic contexts. The enhanced version for *gold* highlights the mountain's golden color emphasizing the setting sun whereas the version for *winter* brightens the image making the snow more salient. Even though there is just one input image, the different contexts result into two different output images. This demonstrates the importance of semantic context to enhance images.

(a) input

(b) *gold* (c) *winter*

Fig. 7. Example image showing a key aspect of semantic image processing: semantic information is crucial for optimal image enhancement. The single input image is enhanced for different semantic contexts: for the keyword *gold* and its respective z scores (left) and for the keyword *winter* and its respective z scores (right). Photo attribution: rickz

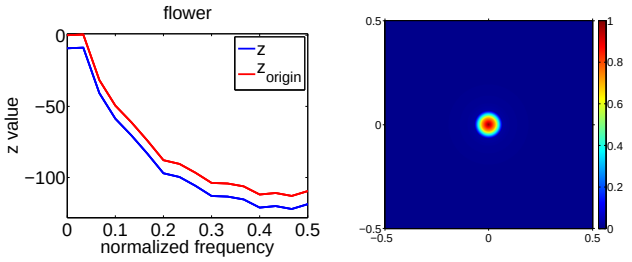
C. Semantic Depth-Of-Field Adaptation

The framework for semantic color enhancement presented in the previous section can be used for other types of image processing. We present an application to semantically enhance the depth-of-field of an image. This is an important characteristic that professionals use to direct an observer's focus to the image's in-focus area and away from the intentionally blurred surrounding.

For this purpose we compute significance values in the Fourier domain. Each image is transformed to the Fourier domain and the average signal per frequency band is computed. Figure 8(a) shows a 16-dimensional vector of z scores for the semantic concept of *flower*. The graph indicates that *flower* images contain significantly less signal power in the high frequency bins. This is reasonable as flower images often are close ups that have been shot with macro lenses or in macro mode.

In order to preserve the average brightness of the image we shift the z scores up to the origin and call it z_{origin}. The shifted significance values are then used to design a semantically adapted filter F in the Fourier domain similar to Equation 3 where S is again a scale parameter that can be used to regulate the impact of the processing[3]. The filter F for *flower* is depicted in Figure 8(b).

$$F = \begin{cases} 1/\left(1 + S \cdot |z_{\text{origin}}|\right) & \text{if } z_{\text{origin}} < 0 \\ 1 + S \cdot z_{\text{origin}} & \text{if } z_{\text{origin}} \geq 0 \end{cases} \tag{6}$$



(a) significance values (b) filter for *flower*

Fig. 8. Left: Significance values for a power spectrum in the Fourier domain. *Flower* images contain significantly less signal power in high frequency bins. Right: Derived Fourier domain filter F computed with Equation 6 and $S = 1$.

We compute the intermediary image I_{global} as a multiplication of the input image I_{in} and the filter F in Fourier domain:

$$I_{\text{global}} = \mathfrak{F}^{-1}\left[\mathfrak{F}(I_{\text{in}}) \cdot F\right] \tag{7}$$

where $\mathfrak{F}$ and $\mathfrak{F}^{-1}$ are the Fourier transform and its inverse, respectively.

The intermediary image I_{global} is globally blurred and does not provide the desired result. Just as in the semantic color enhancement, we need to estimate which regions of the image have to be blurred out. For this purpose we use an algorithm for defocus estimation from Zhuo and Sim [22]. Figures 9(a) and 9(b) show an example input image and its estimated defocus map, respectively.

[3]As stated before, all exmaples in this article use $S = 1$.

The final output image is a linear combination of the input image I_{in} and the intermediary image I_{global} according to Equation 5. Figure 9(c) shows the output image for the *flower* example. It is visible that the background is more blurred than in the input image while the foreground objects remain in focus.

(a) input (b) weight map (c) output for *flower*

Fig. 9. Left: example input image for the semantic depth-of-field adaptation. Middle: defocus estimation [22] to determine relevant regions for adaptive blurring. Right: final output image for the semantic concept of *flower*. Photo attribution: Shaojie Zhuo.

Please note that the significance score controls the impact in both image processing applications (color and depth-of-field). High Δz scores cause a strong enhancement of the relevant characteristics and Δz values close to zero have no visual impact. Consequently, the strong out-of-focus processing in Figure 9 is a result of *flower*'s large Δz scores. A more thorough discussion of the results for semantic image enhancement and results of a psychophysical evaluation is given in Lindner [21].

V. Color Naming

The goal in color naming is to link color values with color names or vice versa. The statistical framework from Section III can also be applied to this problem as it links semantic to numeric data of images. As an illustrative example we can consider Figure 4(b), which shows the distribution of significance values in CIELAB color space for the semantic expression of *periwinkle*. The distribution's maximum is in the bin with center $L = 70$ $a = 10$ $b = -23$, which corresponds to 169, 167, 212 in sRGB color space.

A. Building a Database and Estimating Color Values

As the statistical framework infers information in the numeric image domain from semantic expressions we first need to prepare a list of color names. For this purpose we use the results from the XKCD color survey [23]. This study deduced 950 English color names during a large-scale online color naming experiment where observers were shown randomly colored patches that they had to name.

We use this list of English color names and let it translate by native speakers to 9 other European and Asian languages, namely Chinese, French, German, Italian, Japanese, Korean, Portuguese, Russian, and Spanish, respectively. In some cases it is not possible to find an exact expression in the destination language. For example the color names *burple*[4], *purpleish*

[4]An invented color name that combines *blue* and *purple*.

blue, *purpley blue*, and *violet blue* have all been translated to the same expression in Chinese: 紫色 .

Some of the color names are not frequently tagged on Flickr and we thus use Google Image Search to acquire enough images for the statistical framework. The query consists of the color name in quotation marks plus the word color in the respective language, e.g. "*mint green*" + *color* or "*vert menthe*" + *couleur* in English and French, respectively.

Additionally, we use Google's country and language restrict fields to concentrate the search to websites from a specific country and language as defined in the Custom Search API [24]. This is important in cases where a color has the same spelling in two or more different languages such as *rosa suave*, which means *soft pink* in both Portuguese and Spanish.

We download 100 images per color name and language, which totals to almost 1 million images. We compute CIELAB color histograms with $15 \times 15 \times 15$ equidistant bins in the range $[0 \;\; 100]$ on the L axis and $[-80 \;\; 80]$ on the a and b axes, respectively. Exceeding color values are clipped to the closest bin. The significance values are then computed with the statistical framework from Section III.

We estimate a color name's color value by a linear interpolation around the local neighbourhood $\mathcal{N}$ of the distribution's maximum.

$$L^{\text{est}} = \sum_{i \in \mathcal{N}} z_i L_i \Big/ \sum_{i \in \mathcal{N}} z_i \tag{8}$$

where z_i and L_i are the significance value and the CIELAB L value of bin center i. In our implementation the neighborhood $\mathcal{N}$ is the histogram bin with the maximum score plus the 26 adjacent bins. The CIELAB a^{est} and b^{est} values are computed in the same way.

The color names and their estimated values can be browsed in an online multi-lingual color thesaurus www.colorthesaurus.com. Users can browse through the color space to find similar colors, translate them to other languages and query the database for color names.

B. Analysis

It is not trivial to measure the precision of the estimated colors for two reasons. First, there is an inherent ambiguity in color naming because two different observers might not agree on how a specific color exactly looks like but are guided by personal taste. Second, we are not aware of any big color naming database that covers multiple languages. We thus only have ground truth for the English color names from the XKCD dataset [23]. A comparison of an estimation of a translated color name to its value in the original English version might be affected by translation imprecisions. The following analysis has to be interpreted with these two sources of imprecisions in mind.

We use the ΔE measure (Euclidean distance in CIELAB space) to compare our estimations to the English XKCD ground truth, because there is no ground truth available for other languages. Figure 10 shows distances for the color name *maroon* where the bars are colored in the estimated colors, respectively.

It is visible that there are roughly two groups of colors with ΔE distances of approximately 10 and 30, respectively. The estimation errors in the first group are relatively low because the translator was able to find a good translation of *maroon* in the respective language. In the second group the translators could not find an exact translation of *maroon* and thus constructed color names related to *chestnut*, i.e. the German word *kastanienbraun* literally means *chestnut brown*. Consequently, the estimations are rather brownish and do not have the reddish tint that *maroon* usually implies.

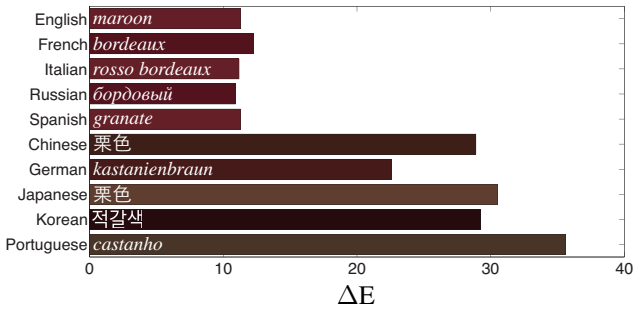


Fig. 10. ΔE distances between our estimations of maroon in 10 different languages and the English ground truth value from the XKCD dataset [23]. Languages that have a direct translation for *maroon* have lower deviations (top 5) than languages where the translations are related to *chestnut* (bottom 5).

Figure 11 shows ΔE distances between XKCD's ground truth value of *maroon* and values defined by other databases of English color names, namely Perbang [25], W3C [26], X11 [27], and Moroney [28]. The data in Moroney's database has been derived in an online psychophysical experiment [29].

When comparing Figures 10 and 11 we see that the ΔE distances are of the same order, indicating that our automatic framework's deviations are comparable to the disagreement between other databases. A more in-depth discussion about the estimated color values is presented in Lindner [21].

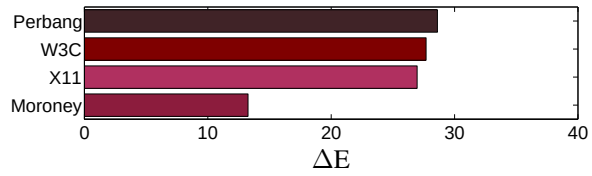


Fig. 11. ΔE distances between other databases (indicated along the vertical axis) and the XKCD ground truth. The disagreement between different databases is comparable to the deviations of our automatic estimation framework.

VI. Color Palettes

A color palette is a set of colors that represents a certain topic. Example palettes are reproduced in Figure 4(c) for different topics. The automatic extraction of palettes from text can also be realized with the statistical framework and is an extension of the color naming application. An automatic approach is clearly superior to a manual palette creation, which is common on popular color palette websites such as Adobe Kuler [1].

In order to allow users to freely describe the emotions they wish to evoke with a color palette we need an unconstrained

vocabulary. Thus we use Google n-grams, a freely available database of word n-tuples from Google's book scanning project [30]. In this case we only consider unigrams, i.e. single words, from the English database. We count the frequency of all words and keep the 100,000 most frequently used words. The last three words in this list are: *Bayswater*[5], *turbidite*[6], and *trabalho*[7]. Any word that is more common than these three is represented in our database. We download for every word 60 images using Google Image Search totalling to 6 million images.

We convert the images to HSV color space, because it is a cylindrical color space that is well suited for circular hue templates as explained in the next subsection. Then we re-map the hue values so that red-green and yellow-blue are opponent colors, which is more popular among artists and denote this $\bar{H}SV$ space[8]. We then compute histograms with 16 equidistant bins along each dimension for all 100,000 words in the database.

A. Palette Extraction

Palette creation tools such as Adobe Kuler [1] often have five color swatches per palette as default and propose predefined hue templates to ensure color harmony. In this study we focus on palettes with five colors and the following templates: "monochromatic" ①, "complementary" ②, "analogous" ③, and "triad" ④. Colors that respect these hue templates are known to be perceived as harmonic [31], [32] independent of the template's global orientation angle on the hue circle.

Color palettes are subject to taste and not an exact science. This is the reason why the same hue templates are sometimes even applied to different hue circles, e.g. CIELAB and HSB [33], indicating that templates are rather a rule of thumb. We use the HSV color space because it is a cylindrical space that makes it easy to optimize for the circular hue templates. Unfortunately, HSV does not have the opponent colors red-green and blue-yellow opposite to each other, which artists usually prefer because it better reflects human color perception. Thus we linearly re-map the HSV hue component so that green (originally 120°) is at 180°as described before.

Similar to Adobe Kuler we vary the colors' saturations and values to introduce some variety. Our palette definitions are summarized in Table I. All templates have three degrees of freedom which are the hue angle $\bar{h}$, the saturation s, and the value v. The ③ template has an additional degree of freedom α that determines the angle between the equally spread hue angles.

In order to find the best template parameters we define a score s that estimates the conformity of a palette $\mathbf{c}_n$:

$$s(\mathbf{c}_n) = \sum_{\mathbf{c}\in\bar{H}SV} z(\mathbf{c}) \cdot \max_{n=1\ldots5} \left[\exp \frac{(\mathbf{c}-\mathbf{c}_n)^2}{\sigma^2}\right] \qquad (9)$$

[5]An area of London.

[6]A geological formation from underwater avalanches.

[7]The Portuguese translation of *work*.

[8]We stretch the interval [0 120] to [0 180] and compress the interval [120 360) into [180 360) bringing green opposite to red and yellow opposite to blue.

TABLE I
DEFINITIONS FOR 4 HUE TEMPLATES. SIMILAR TO ADOBE KULER [1] WE VARY THE COLORS' SATURATIONS AND VALUES IN ORDER TO CREATE SOME DIVERSITY. EACH PALETTE THUS HAS THREE PARAMETERS h, s, AND v. THE ANALOGOUS TEMPLATE HAS AN ADDITIONAL PARAMETER α THAT DETERMINES THE ANGLE BETWEEN THE EQUIDISTANTLY SPREAD HUES.

①			②		
$\bar{h}$	$s-0.3$	$v+0.05$	$\bar{h}$	$s-0.1$	$v+0.3$
$\bar{h}$	s	$v+0.3$	$\bar{h}$	$s+0.1$	$v-0.2$
$\bar{h}$	s	v	$\bar{h}$	s	v
$\bar{h}$	$s-0.3$	$v+0.3$	$\bar{h}+0.5$	$s+0.2$	$v-0.2$
$\bar{h}$	s	$v-0.2$	$\bar{h}+0.5$	s	v
④			③		
$\bar{h}+0.33$	$s-0.1$	v	$\bar{h}-2\alpha$	$s+0.05$	$v+0.1$
$\bar{h}$	$s+0.1$	$v-0.3$	$\bar{h}-\alpha$	$s+0.05$	$v+0.05$
$\bar{h}$	s	v	$\bar{h}$	s	v
$\bar{h}+0.66$	$s+0.1$	$v-0.2$	$\bar{h}+\alpha$	$s+0.05$	$v+0.05$
$\bar{h}+0.66$	s	$v+0.3$	$\bar{h}+2\alpha$	$s+0.05$	$v+0.1$

where $\mathbf{c} = [\bar{h}\ s\ v]'$ is a color in $\bar{H}SV$ color space, $\mathbf{c}_n, n \in \{1\ldots5\}$ are the five colors that form a palette, $z(\mathbf{c})$ is the significance distribution for a semantic expression, and σ the variance of the local averaging. The hue angles wrap around the interval borders [0 1]. The optimal palette's colors $\mathbf{C}^P$ and score $s(\mathbf{C}^P)$ for a given template $P \in \{①, ②, ④, ③\}$ are:

$$\mathbf{C}^P = \underset{\mathbf{c}_n\in P}{\operatorname{argmax}}\, s(\mathbf{c}_n) \qquad s(\mathbf{C}^P) = \max_{\mathbf{c}_n\in P} s(\mathbf{c}_n) \qquad (10)$$

Example palettes are reproduced in Figure 4(c) along with their hue templates.

Please note that our choices of color space (re-mapped HSV) and palette types {①, ②, ④, ③} are a pragmatic adaptation of common color palette usage to our statistical framework. A study to evaulate different color spaces and palette types would require a large psychophysical experiment, which is out of the scope of this article. Yet, the fact that our framework is fully automatic allows to use it with other color spaces and palette types depending on user preference. For a fair evaluation we make the palette database accessible online: www.koloro.org.[9]

VII. STATISTICAL EVALUATION

The applications presented in the previous sections require to infer an information in the image domain from the semantic domain. This is the opposite direction than common computer vision applications such as image classification and thus we do not aim for discriminating power to avoid confusion between semantic classes. Hence, we do not use common machine learning techniques such as support vector machines or neural networks but use less complex statistical methods instead. This also makes the framework scale more easily to larger datasets.

Section III introduces the statistical framework to measure the impact of a keyword on an image characteristic. Given a keyword w, the characteristics c_i of all images I_i with annotations A_i are split into two sets $\mathcal{C}_w = \{c_i, w \in A_i\}$ and

[9]*koloro* is the Esperanto word for *color*.

TABLE II
ORDERS OF COMPUTATIONAL COMPLEXITY OF DIFFERENCE MEASURES THAT COMPARE TWO SETS OF SCALAR VALUES. MIDDLE LINE: SOME METHODS ALLOW PRE-COMPUTATIONS THAT HAVE TO BE DONE ONLY ONCE FOR A GIVEN DATABASE. BOTTOM LINE: COMPLEXITY TO COMPUTE THE DIFFERENCE FOR ONE SINGLE KEYWORD USING THE EVENTUALLY PRE-COMPUTED INTERMEDIATE RESULT. $N_I \approx 10^6$ DENOTES TOTAL NUMBER OF IMAGES, $N_{IW} \approx 10^2 - 10^3$ THE AVERAGE NUMBER OF IMAGES PER KEYWORD, AND $N_B \approx 10^1 - 10^2$ THE NUMBER OF HISTOGRAM BINS.

	Mann-Whiteny-Wilcoxon	Kolmogorov-Smirnov	Chi-square	Student's t-test	Hodges-Lehmann	Earth mover's distance
pre-computation	$N_I \log(N_I)$	$N_I \log(N_I)$	N_I	–	–	N_I
one keyword	N_{IW}	$N_{IW} \log N_I + N_I$	$N_{IW} + N_B$	N_I	$(N_I - N_{IW})N_{IW}$	$N_{IW} + N_B$

$\mathcal{C}_{\overline{w}} = \{c_i, w \notin A_i\}$ that contain all characteristics of images annotated with keyword w and all other images, respectively. A keyword's impact is determined by how much the values in the sets $\mathcal{C}_w$ and $\mathcal{C}_{\overline{w}}$ differ from each other. In this section we demonstrate that the Mann-Whitney-Wilcoxon test is optimal in terms of both computational complexity and accuracy.

A. Difference Measures and their Computational Complexity

There are numerous ways to assess how, and how much, values in one set differ from values in another set. The difference measure we need has to be non-parametric as we want it to work with any image characteristic and we thus cannot make assumptions about a possible underlying distribution of the values in either set.

This section gives an overview of possible difference measures and discusses them in terms of computational complexity. Without loss of generality we consider only one scalar characteristic because a vectorial characteristic is processed by considering each of its scalar values independently (e.g. one bin of the color CIELAB histogram reproduced in Figure 4(b)).

We define N_I the total number of images in the database, N_{IW} the average number of images per keyword in the database, and N_B the number of bins[10]. The computational complexities of all methods are summarized in Table II.

Mann-Whitney-Wilcoxon test [19], [20]: The values in both sets are combined $\mathcal{C}_w \cup \mathcal{C}_{\overline{w}}$ and then sorted in increasing order. The positional indexes of the first set's values are summed up, which yields the test statistic T. T is then normalized, but this does not add to the method's complexity.

Important is that the sorting, the more complex operation, has to be done only once for a given dataset – $\mathcal{O}(N_I \log N_I)$. To compute the test statistic for a new keyword, it is then sufficient to just sum up the rank indexes at the respective positions – $\mathcal{O}(N_{IW})$.

Kolmogorov-Smirnov test [34], [35]: A goodness-of-fit test that compares the cumulative distribution functions $F_w(x)$ and $F_{\overline{w}}(x)$ of two sets. The test statistic is the maximum difference between the two CDFs at any point x.

One way to compute this statistic (Matlab's implementation) is to sort the merged set and define each value as a histogram bin center – $\mathcal{O}(N_I \log N_I)$ in pre-computation. Then each set's values are binned into that histogram – $\mathcal{O}(N_{IW} \log N_I)$. The minimum of the absolute difference between the two CDFs is the statistic – $\mathcal{O}(N_I)$.

[10]The Chi-square test and the Earth mover's distance do not use continuous values, but bin the values first as explained in the respective descriptions.

Chi-square test [36]: This test requires both set's values to be binned into two separate histograms – $\mathcal{O}(N_I)$. The histograms are referred to as observed O and expected E distributions and stem from the sets $\mathcal{C}_w$ and $\mathcal{C}_{\overline{w}}$, respectively. The test statistic X^2 is then:

$$X^2 = \sum_{i=1}^{N_B} \frac{(O_i - E_i)^2}{E_i} \tag{11}$$

For a new keyword, a second histogram has to be computed, which is the observed histogram O – $\mathcal{O}(N_{IW})$. Its subtraction from the first combined histogram then gives the expected histogram E – $\mathcal{O}(N_B)$. The computation of the sum in Equation 11 has again complexity – $\mathcal{O}(N_B)$.

Student's t-test [37]: Due to the popularity of the test we include it in this overview even though it is parametric. Depending on whether equal variances in the two sets can be assumed or not, different estimators for the common standard deviation are used, but in any case one needs to compute for every word the means and variances of both sets – $\mathcal{O}(N_I)$.

Hodges-Lehmann estimator [38]: This estimator measures the effect size, which is, unlike the statistical significance, independent of the sample sizes (see discussion in Sec. III-B). This method first computes all differences $\Delta_{i1,i2} = c_{i1} - c_{i2}$ between any combination of characteristics of the first $\mathcal{C}_w$ and second set $\mathcal{C}_{\overline{w}}$– $\mathcal{O}((N_I - N_{IW})N_{IW})$. The estimated value is the median of all differences (same complexity using the *Median of Medians* algorithm).

Earth mover's distance [39], [40]: This distance metric compares two histograms to each other. Therefore, the values in both sets first have to be binned into a histogram each, just like with the Chi-square test. The distance is then defined as a transportation problem [39] where the bin counts are considered to be piles of dirt. This can be computed in linear time – $\mathcal{O}(N_B)$.

Comparison: The Mann-Whitney-Wilcoxon test has a comparatively low complexity, especially the computational cost to add an additional keyword. This can be done with only N_{IW} summation operations, where N_{IW} is the average number of images per keyword, in our case in the range of hundreds to thousands. This is the reason why we can easily compute significance scores for very large vocabularies The Chi-square and Earth mover's distance also have relatively little cost to compute a distance for an additional keyword, but they do not perform as well as we show in the following subsection.

B. Quantitative Comparison

We want to compare the difference measures in a quantitative experiment. We thus choose the color naming application because, unlike the others, it offers a mathematical way to

measure the precision of the estimations as shown in Section V-B. We focus on the English color names because this is the original data with ground truth from the psychophysical experiment [23]. We use the same images and the same CIELAB histograms, but compute the difference distributions using all difference measures presented in the previous subsection and Table II. For each difference measure we estimate a color name's color values using Equation 8.

To measure the accuracy we compute the ΔE distance between the estimated color and the XKCD ground truth value. Figure 12 summarizes the estimation errors of all tested difference measures. We see that the Mann-Whitney-Wilcoxon test yields the lowest errors, i.e. best performance, among all six tested methods at any quantile.

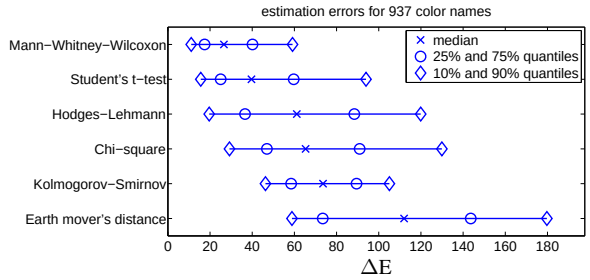


Fig. 12. Estimation errors for the English XKCD color names using six difference measures. The Mann-Whitney-Wilcoxon test has the highest accuracy, i.e. lowest error, as all its quantiles are lower than those of any other competing measure.

VIII. Conclusions

In this article we propose methods and applications that use the semantic domain to infer knowledge and actions in the image domain. This is the opposite direction of classic computer vision applications. This has a large potential considering that a lot of multimedia data in social online communities is annotated with plenty of semantic information such as keywords, user comments and so forth.

The core of the presented applications is a highly scalable statistical framework to compute associations between image characteristics and semantic information. We discuss multiple methods to realize the computation of the associations and conclude that the Mann-Whitney-Wilcoxon test is best in terms of both accuracy and computational complexity. The statistical framework handles millions of images and hundreds of thousands of keywords.

We presented three applications that leverage the statistical framework. First, semantic image enhancement where an input image is re-rendered in order to optimize it for a given semantic concept. We implemented two types of enhancement, which are color tone mapping as shown in Figure 6 and semantic depth-of-field adaptation as shown in Figure 9.

Figure 7 illustrates an important aspect of semantic image processing: enhancement requires context. An artist's intent can require certain dominant image features (e.g. colors or blur) for a given photo, which can be considered bad for another image. We propose to estimate the image's context from its keywords, if present.

The second application is automatic color naming. We estimate color values for over 9000 color names in 10 different European and Asian languages. The accuracy is comparable to other databases. The results are freely accessible in the form of an online color thesaurus: www.colorthesaurus.com.

The third application is color palette extraction where the goal is to determine a palette of five harmonic colors that represents a given semantic expression. Palettes can be searched here: www.koloro.org

Example code and raw significance distributions are available for download: http://ivrl.epfl.ch/Lindner_IEEE_MM_2015.

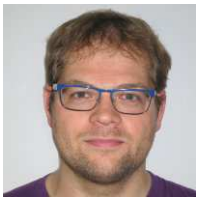
Acknowledgement

The authors would like to thank Nicolas Bonnier, Christophe Leynadier, and Maria-Valezzka Ortiz-Segovia for their support and the many interesting discussions.

References

[1] "Adobe Kuler," https://kuler.adobe.com, last checked Oct. 2014.
[2] ImageNet Large Scale Visual Recognition Challenge, http://www.image-net.org/, last checked Oct. 2014.
[3] Caltech 256, http://www.vision.caltech.edu/Image_Datasets/Caltech256/, last checked Oct. 2014.
[4] PASCAL Visual Object Classes, http://pascallin.ecs.soton.ac.uk/ challenges/VOC/, last checked Oct. 2014.
[5] LabelMe, http://labelme.csail.mit.edu/, last checked Oct. 2014.
[6] T. Deselaers and V. Ferrari, "Visual and semantic similarity in imagenet," in *CVPR*, 2011, pp. 1777–1784.
[7] M. J. Huiskes and M. S. Lew, "The MIR flickr retrieval evaluation," in *ACM MIR*, 2008, pp. 39–43.
[8] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer Graphics and Applications*, vol. 21, no. 5, pp. 2–9, 2001.
[9] S. B. Kang, A. Kapoor, and D. Lischinski, "Personalization of image enhancement," in *CVPR*, 2010, pp. 1799–1806.
[10] S. Moser and M. Schroeder, "Usage of DSC meta tags in a general automatic image enhancement system," in *IS&T EI*, vol. 4669, San Jose, CA, USA, January 2002, pp. 259–267.
[11] G. Ciocca, C. Cusano, F. Gasparini, and R. Schettini, "Content aware image enhancement," in *Artificial Intelligence and Human-Oriented Computing*, vol. 4733/2007, Rome, 2007, pp. 686–697.
[12] G. Laput, M. Dontcheva, G. Wilensky, W. Chang, A. Agarwala, J. Linder, and E. Adar, "Pixeltone: A multimodal interface for image editing," in *ACM Human Factors in Computing Systems*, 2013, pp. 2185–2194.
[13] J. M. R. Michael J. Jones, "Statistical color models with application to skin detection," *International Journal of Computer Vision*, vol. 46, no. 1, pp. 81–96, 2002.
[14] D. Sekulovski, G. Geleijnse, B. Kater, J. Korst, S. Pauws, and R. Clout, "Enriching text with images and colored light," in *IS&T EI*, vol. 6820, Multimedia Content Access: Algorithms and Systems II, 2008.
[15] J. van de Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications," *TIP*, vol. 18, no. 7, pp. 1512 – 1523, 2009.
[16] "Color Hunter," http://www.colorhunter.com, last checked Oct. 2014.
[17] "CSS Drive," http://www.cssdrive.com/imagepalette/, last checked Oct. 2014.
[18] "Pictaculous," http://www.pictaculous.com, last checked Oct. 2014.
[19] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.
[20] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The Annals of Mathematical Statistics*, vol. 18, no. 1, pp. 50–60, 1947.
[21] A. Lindner, "Semantic awareness for automatic image interpretation," Ph.D. dissertation, EPFL School of Computer and Communication Sciences, 2013.
[22] S. Zhuo and T. Sim, "Defocus map estimation from a single image," *Pattern Recognition*, vol. 44, no. 9, pp. 1852–1858, 2011.
[23] "XKCD color survey," http://blog.xkcd.com/2010/05/03/color-survey-results/, last checked Oct. 2014.
[24] Google Custom Search API, https://developers.google.com/custom-search/docs/xml_results, last checked Oct. 2014.

[25] Color Database, http://www.perbang.dk, last checked Oct. 2014.
[26] *CSS Color Module Level 3*, W3C Recommendation, World Wide Web Consortium, 7 June 2011.
[27] X11 color names, http://cvsweb.xfree86.org/cvsweb/xc/programs/rgb/rgb.txt, last checked Oct. 2014.
[28] N. Moroney, http://www.hpl.hp.com/personal/ Nathan_Moroney/, last checked Oct. 2014.
[29] N. Moroney, "Unconstrained web-based color naming experiment," in *IS&T EI*, vol. 5008, Color Imaging VIII: Processing, Hardcopy, and Applications, 2003, pp. 36–46.
[30] Google n-grams, http://books.google.com/ngrams/, last checked Oct. 2014.
[31] Y. Matsuda, *Color Design (in Japanese)*. Asakura Shoten, 1995.
[32] P. O'Donovan, A. Agarwala, and A. Hertzmann, "Color compatibility from large datasets," in *SIGGRAPH*, 2011, p. Article No. 63.
[33] B. J. Meier, A. M. Spalter, and D. B. Karelitz, "Interactive color palette tools," *IEEE Comput Graph Applications*, vol. 24, no. 3, pp. 64–72, 2004.
[34] A. Kolmogorov, "Sulla determinazione empirica di una legge di distributione," *Giornale dell' Istituto Italiano degli Attuari*, pp. 83–91, 1933.
[35] N. Smirnov, "Table for estimating the goodness of fit of empirical distributions," *The Annals of Mathematical Statistics*, vol. 19, no. 2, pp. 279–281, 1948.
[36] R. L. Plackett, "Karl pearson and the chi-squared test," *International Statistical Review*, vol. 51, no. 1, pp. 59–72, 1983.
[37] Student, "The Probable Error of a Mean," *Biometrika*, vol. 6, no. 1, pp. 1–25, 1908.
[38] J. L. Hodges and E. L. Lehmann, "Estimates of location based on rank tests," *Annals of Mathematical Statistics*, vol. 34, no. 2, pp. 598–611, 1963.
[39] F. L. Hitchcock, "The distribution of a product from several sources to numerous localities," *Journal of Mathematical Physics*, vol. 20, pp. 224–230, 1941.
[40] S. Shirdhonkar and D. W. Jacobs, "Approximate earth mover's distance in linear time," in *CVPR*, 2008, pp. 1–8.



Albrecht Lindner received Masters degrees in electrical engineering and signal processing from the University of Stuttgart, Stuttgart, Germany, and Telecom ParisTech, Paris, France in 2008, respectively. He received his Ph.D. degree from the École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 2013. Dr. Lindner was the recipient of the Fritz Kutter Best Thesis Award.

He is currently a Senior Engineer with the Multimedia Research and Development Team, Qualcomm, San Diego, CA, USA. Previously, he was a Postdoctoral Researcher with ADSC (University of Illinois), Singapore. His research interests include color imaging, image processing, and large-scale statistical image analysis.

Sabine Süsstrunk leads the Images and Visual Representation Lab (IVRL) in the School of Computer and Communication Sciences (IC) at EPFL since 1999. Her main research areas are in computational photography, color imaging, multimedia, and image quality. She has authored and co-authored over 150 publications and holds 8 patents. In 2013, she received a Best Paper Award at the IEEE International Conference on Image Processing (ICIP) and the IS&T/SPIE 2013 Electronic Imaging Scientist of the Year Award. Sabine is currently Associate Editor for the IEEE Transactions on Computational Imaging.