Joint Statistical Analysis of Images and Keywords with Applications in Semantic Image Enhancement

Albrecht Lindner
School of Computer and Communication Sciences
EPFL, Switzerland
Albrecht.Lindner@epfl.ch

Appu Shaji
School of Computer and Communication Sciences
EPFL, Switzerland
Appu.Shaji@epfl.ch

Nicolas Bonnier
Océ Print Logic Technologies
Créteil, France
Nicolas.Bonnier@gmail.com

Sabine Süsstrunk
School of Computer and Communication Sciences
EPFL, Switzerland
Sabine.Susstrunk@epfl.ch

ABSTRACT

With the advent of social image-sharing communities, millions of images with associated semantic tags are now available online for free and allow us to exploit this abundant data in new ways. We present a fast non-parametric statistical framework designed to analyze a large data corpus of images and semantic tag pairs and find correspondences between image characteristics and semantic concepts. We learn the relevance of different image characteristics for thousands of keywords from one million annotated images. We demonstrate the framework's effectiveness with three different examples of semantic image enhancement: we adapt the gray-level tone-mapping, emphasize semantically relevant colors, and perform a defocus magnification for an image based on its semantic context. The performance of our algorithms is validated with psychophysical experiments.

Categories and Subject Descriptors

L.01 [**Media Content Analysis and Processing**]: semantic concept detection

Keywords

semantic image processing, statistical analysis, image understanding, crowd sourcing, large-scale experimentation

1. INTRODUCTION

The abundance of data in online image-sharing communities gives new perspectives and possibilities to the multimedia research community. Photos and associated semantic tags can be analyzed on a large scale in order to gain mathematical models that relate human language (annotated keywords) to computer language (numeric pixel values). We present a statistical framework that estimates this correspondences between image keywords and characteristics, and design it to be fast even on large-scale databases. Such correspondence can be exploited for imaging applications that need to interpret semantic input, as is the case in semantic image enhancement for which we present three algorithms. First, we adjust an image's gray-level tone-mapping for a semantic concept. Second, we emphasize related colors in order to strengthen a semantic concept. And third, we magnify the defocus of an image to account for its semantic content.

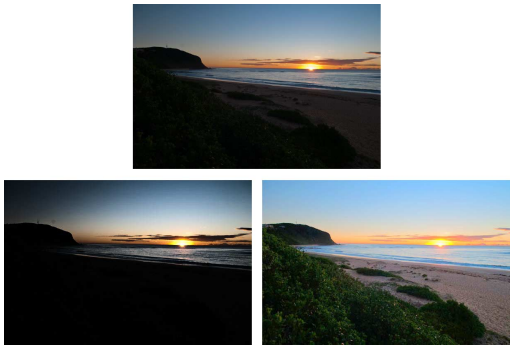
Figure 1 shows an example for each of the three implemented algorithms. Figure 1(a) shows the adaptation of an image to the semantic concepts *dark* and *sand*, respectively. Figure 1(b) shows how an image's colors can be re-rendered in order to emphasize its semantics – *strawberry*. Finally, Figure 1(c) demonstrates a defocus magnification to account for the semantic concept *macro*.

Our statistical framework (Sec. 3) is non-parametric and learns associations between any image characteristic and semantic tag as it is not necessary to assume an underlying distribution. This makes it fast in comparison to other learning techniques that depend on parameter-tuning (e.g., SVM or boosting). Hence, we are able to use a database of one million annotated images [7] and learn associations for thousands of keywords. In addition to that, a parameter- and tuning-free method opens the door to applications where it is difficult to numerically measure the quality of the result: for example the perceptual appeal of a semantically enhanced image in our case.

The philosophy behind our approach to semantic image enhancement is that it is not possible to optimize the visual appearance of an image based only on the pixel values. For an optimal result, it is indispensable to know its semantic context. This is demonstrated in Figure 1(a), which shows a beach scene (top) that can be enhanced in order to emphasize that it was almost dark (bottom left) or to show the sandy beach (bottom right). Conventional image-statistics based enhancement algorithms such as contrast stretching are not able to do this because they do not take into account the semantic context.

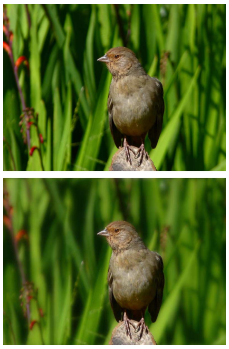
The three presented algorithms for semantic image en-

MM'12, October 29–November 2, 2012, Nara, Japan.



(a) gray-level re-rendering, *dark* and *sand*

(b) color re-rendering, *strawberry*



(c) frequency re-rendering, *macro*

Figure 1: Examples for the three semantic image enhancement algorithms presented in this paper. All algorithms take as inputs a single image and an associated semantic expression, which is indicated in the sub-caption in *italic*. Left: The gray levels of the input image at the top are tone mapped to adjust for the semantic concepts *dark* and *sand*. Middle: The input image's colors are re-rendered in order to emphasize the concept *strawberry*. Right: Defocus magnification to adapt to the semantic concept *macro*. The input image (top) is reproduced from Zhuo and Sim [30].

hancement (Sec. 4) are based on two components: 1) the semantic context as defined by a keyword 2) the image content as defined by the pixel values. The significance values from the statistical test allow to determine whether a specific processing of an image under a given semantic context is meaningful and, if yes, how the processing has to be done in order to achieve an optimal result. This is combined with information from the pixel values in order to tailor the semantic processing to a specific image. The combination of semantic and pixel-based information for image enhancement is another novelty presented here.

We evaluate the semantic gray-level enhancement with two psychophysical experiments. The first shows that our enhanced image is preferred over the original image. The second compares our algorithm against other algorithms that are not semantically adaptive. Our method performs on average 2.5 times better than the second best.

2. RELATED WORK

Our paper addresses image semantics and image enhancement. Here we present related publications that inspired this work.

Torralba and Freeman published a "tiny image database" [24] with 80 million 32×32 images associated with 75,062 non-abstract nouns from the WordNet lexical database [13]. It enables large-scale image classification, but is limited to non-abstract nouns. ImageNet [4] is a similar project aimed at populating each synset of WordNet with 1000 images on average. ImageNet is used by Deselaers and Ferrari [5] to show that images with semantically similar annotations have more visual attributes in common than images with dissimilar annotations. This work proves that an image's semantics can be associated with its low-level image descriptor, which we also exploit here. Ordonez et al. present an approach to leverage one million tagged photographs in order to automatically create a verbal image description for a new input image [17]. This can be seen as the inverse of our application, because we aim at changing image characteristics given a tagged keyword.

The MIR Flickr database [7] contains one million high-quality photographs that are selected based on Flicker's "interestingness" score. The images are provided with the Flickr community's annotations, which sets this database apart from the previously mentioned. The annotations are of lower quality because they come from a large uncontrolled group of people (social tagging [22]), but they are abundant. Both the high quality of the photographs and the social tagging made us choose this database.

Associating image features with semantic tags has been investigated for several applications. In most cases, for example in image retrieval [23, 28], the goal is to link image features with a set of semantic classes. However, our method does the inverse and associates semantic labels with images. Generating training data for the former is cumbersome as it involves hand labeling of image regions, whereas we profit from user annotated images that are readily available from various crowd-sourcing data repositories. This allows us to scale to millions of images and to an unlimited vocabulary.

Our application scenario is image enhancement, which is a widely explored field. Image enhancement algorithms can be classified into different groups. One group of algorithms relies on a set of rules (defined by a human expert) that an enhanced image should satisfy. An input image is then modified so that it better respects these rules. These methods can work on a single image without any other input. A simple example is the rule that in an image's histogram, the bin counts should be more or less equal. This so-called histogram equalization process improves an image's contrast and has been known for a few decades [8]. Another example is unsharp masking, where the input image is convolved with a high-pass filter and added back to the original image in order to make it look sharper [19].

More recent and sophisticated examples are methods to increase region saliency from Fredembach [6] or to adjust color harmony in an image from Cohen-Or et al. [3] or Sauvaget and Boyer [21]. Wang et al. [26] and Murray et al. [15] present methods to adjust an image's color composition with predefined color themes, such as "nostalgic" or "spicy". However, their approaches are limited as the color themes are manually defined. On the contrary, our approach can

interpret any semantic expression at the input and deduce an appropriate image processing from it.

Another group of image enhancement algorithms are example-based. In this case, the algorithm adjusts the characteristics of an input image with those of one or more example images: depending on the example images, different enhancements can be achieved. Reinhard et al. [20] propose a system that transfers the colors from an example image to an input image. Kang et al. [10] develop a method where a user creates personal example images in a previous step. The parameters from the example set are then used to personalize the enhancement of a new input image. Wang et al. [27] present a framework to map colors and gradients. They use an example set of registered image pairs of scenes taken with a low-end and a high-end camera. The mappings from the low to the high-end images are then applied to an input image. Our approach can be seen as example-based, but instead of creating our own image examples, we use keywords whose dominant image descriptors are derived from a large number of freely available images.

Another approach to automatic image enhancement is to classify an image (or regions of it) into a fixed set of image categories before applying a class-dependent image enhancement. Such systems have been proposed by Moser and Schroeder [14] and Ciocca et al. [2]. They use common classes, such as "sky", "skin", or "vegetation". Both approaches are adapted to image semantics. However, only seven and three semantic concepts are distinguished, respectively, whereas our framework can deal with an arbitrary number of keywords.

A somewhat different group of image enhancement algorithms create artistic effects. An example of this is defocus magnification, where the goal is to additionally blur out-of-focus regions so that the object in focus is more accentuated [1, 30]. These algorithms first compute a defocus map [16] and then intentionally blur the image according to the estimated defocus level.

The main difference between our semantic image processing and all the above mentioned methods is that the semantic concept provides a second independent input to our workflow in addition to the image features. Other approaches only classify an image into a limited number of pre-selected semantic classes [2, 14] and then process according to that classification. Thus, these methods do not allow rendering the same input image for two different semantic concepts, as our approach is able to do (see Fig.1(a)).

3. STATISTICAL FRAMEWORK

In this section we present the framework that relates keywords with image characteristics. To do this, we learn correspondences between image characteristics and keywords by using the MIR Flickr database with 1 million annotated high-quality images [7].

3.1 Measuring Correspondence

Our database consists of image/annotation pairs $(I_i, A_i) \in I_{db}$. An annotation is an ordered set of one or more keywords $A_i = \{w_1, w_2, \ldots\}$. Given a keyword w, the database can be split into two subsets $I_w = \{I_i | w \in A_i\}$ and $I_{\overline{w}} = \{I_i | w \notin A_i\}$ that contain all images annotated with keyword w and all remaining images, respectively. It is $I_w \cap I_{\overline{w}} = \emptyset$ and $I_w \cup I_{\overline{w}} = I_{db}$.

For an image I, a characteristic $C \in \mathbb{R}$ can be computed from it. This can be anything we want to characterize in an image. Examples are the percentage of pixels that have a certain gray level or the output of Gabor filters. The set $\mathcal{C}_w^j = \{C_i^j | I_i \in I_w\}$ unites the characteristic j of all images annotated with keyword w. The set $\mathcal{C}_{\overline{w}}^j$ unites analogously the characteristic j from images in $I_{\overline{w}}$.

In order to assess how a keyword influences a characteristic j, the values in the sets $\mathcal{C}_w^j$ and $\mathcal{C}_{\overline{w}}^j$ have to be compared against each other. The task is to determine how the values of the two sets differ.

In the general case, the values do not follow a known distribution. Hence, we use methods from non-parametric statistics. A commonly used test is the Mann-Whitney-Wilcoxon (**MWW**) ranksum test [29, 12], which assesses whether two observations have equally large values, i.e., by how much their means differ.

For given characteristics (j)[1], let the computed scalar values from the positive set be $\mathcal{C}_w = \{C_1, C_3, C_4, \ldots\}$ and that from the negative set $\mathcal{C}_{\overline{w}} = \{C_2, C_5, C_6, \ldots\}$ for a given keyword w. We first sort the concatenated set $\mathcal{C}_w \cup \mathcal{C}_{\overline{w}}$. The ranksum is defined as the sum of positional indexes that the elements of $\mathcal{C}_w$ occupy in the sorted list of $\mathcal{C}_w \cup \mathcal{C}_{\overline{w}}$. Wilcoxon denoted this statistic with T

For example, consider the two sets $\mathcal{C}_w = \{17.5, -2\}$ and $\mathcal{C}_{\overline{w}} = \{23, -11.7, 3.1, 0.9, 42\}$. The ordered sequence of all C_i's is: $\overset{1}{-11.7}$, $\overset{2}{-2}$, $\overset{3}{0.9}$, $\overset{4}{3.1}$, $\overset{5}{17.5}$, $\overset{6}{23}$, $\overset{7}{42}$ (rank indexes stacked on top for convenience), and thus the ranksum $T = 2 + 5 = 7$. The expected mean and variance are [29, 12]:

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

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

where $N_w = |\mathcal{C}_w|$ and $N_{\overline{w}} = |\mathcal{C}_{\overline{w}}|$ are the cardinalities of either set, respectively.

There are other tests such as the Kolmogorov-Smirnov test or the χ^2 test that additionally assess whether two distributions have different shapes [25]. As in our application, image enhancement, the absolute values are more important than the shape of their distribution, we us the **MWW**-test. We have found it to be more robust for our experiments.

Finally, we compute the standardized z value:

$$z = \frac{T - \mu_T}{\sigma_T} \tag{2}$$

We give more details on statistical tests and our choice to favor the **MWW**-test on the project webpage[2].

3.2 Interpretation of the z value

The z value is a useful measure to assess the relationship between keywords and low-level image features. The higher its magnitude, the more the corresponding characteristic is important for the keyword, and vice versa.

To give a better intuition for the z value, we consider an example where the tested image characteristic is a 16 bin gray-level histogram. For each of the equidistant bins, we calculate z_{night}^j from the two sets $\mathcal{C}_{night}^j$ and $\mathcal{C}_{\overline{night}}^j$, where $j = 1 \ldots 16$.

Figure 2 shows the distributions of all pairs, along with their corresponding z_{night}^j values. It is clearly visible that

[1] superscript ^{j} omitted in this paragraph for clarity.

[2] `http://ivrg.epfl.ch/SemanticEnhancement.html`

images annotated with *night* have more dark pixels ($z > 0$ for low gray levels) and less bright pixels ($z < 0$ for high gray levels). The z values smoothly vary between -130 and 124. The difference between $\mathcal{C}^j_{night}$ and $\mathcal{C}^j_{\overline{night}}$ is less significant for z values close to zero, which is the case for $j = 5$ (a medium gray level). Overall though, an image's "*night*ness" is strongly related to its gray-level distribution.

Figure 3 shows the same plots but for the keyword *statue*. The two distributions are much more similar, the z values are closer to zero. This tells us that an image's gray-level distribution and its "*statue*ness" are not related.

We can thus introduce a simple ranking criterion for a given descriptor and keyword, which is the difference between the maximum and the minimum z-value as indicated in Figure 2. According to the examples depicted in Figures 2 and 3, we obtain $\Delta z^{\text{gray-level hist}}_{night} = 124.3 - (-130.0) = 254.3$ and $\Delta z^{\text{gray-level hist}}_{statue} = 6.5 - (-1.2) = 7.7$.

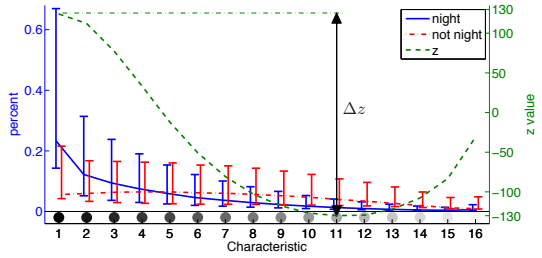


Figure 2: Left axis: The 16 characteristics of the sets $\mathcal{C}^j_{night}$ and $\mathcal{C}^j_{\overline{night}}$ measuring the percentage of image pixels falling into each bin. Each characteristic is represented with its median and its 25% and 75% quantiles. The markers at the bottom indicate the mean gray level of each characteristic. For visualization purposes the two curves have a small horizontal offset. Right axis: The corresponding z values indicate that images annotated with *night* contain more dark ($z > 0$) and less bright ($z < 0$) pixels than the other images not annotated with *night*.

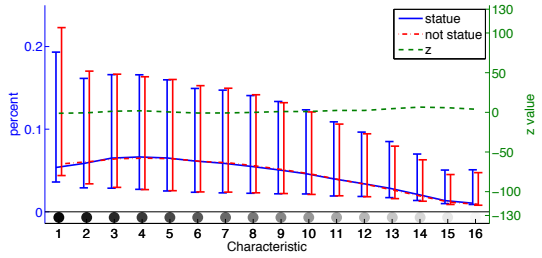


Figure 3: Same plot as in Figure 2 but for keyword *statue*. The distributions are more similar and the z values are closer to zero.

3.3 Comparing z values from Different Keywords and Characteristics

The z values can be computed for many keywords and characteristics. We use all keywords that occur in at least 500 images of the MIR Flickr database, 2858 in total. Additionally to the gray levels, we compute other image characteristics: lightness, chroma, hue angle (all three in CIELAB space [9]), linear binary patterns [11], responses of high-pass and Gabor filters (image details), and frequency distributions in the Fourier domain. They are either summarized in a 16-bin histogram or in a 64-dimensional layout descriptor[3].

3.3.1 Dependency on N_w

The z value depends on the number of images per keyword N_w as can be seen in Equations 1 and 2. This is an inherent property of any statistical test: more samples increase credibility and thus result in a higher significance value.

If the significance values from keywords with different numbers of samples have to be compared it is necessary to introduce a reference sample size N^*_w. All the variables from the statistical test can then be converted to this reference sample size as follows:

$$T^* = \frac{N^*_w}{N_w} \cdot T \tag{3a}$$

$$\mu^*_T = \frac{N^*_w}{N_w} \cdot \mu_T \tag{3b}$$

$$\sigma^{*2}_T = \frac{N^*_w N^*_{\overline{w}}}{N_w N_{\overline{w}}} \cdot \sigma^2_T \tag{3c}$$

$$z^* = \sqrt{\frac{N^*_w N^*_{\overline{w}}}{N_w N_{\overline{w}}}} \cdot z \tag{3d}$$

The better comparability can be demonstrated with the keywords *bw*, *blackandwhite* and *blackwhite*. The standard significance values are $\Delta z^{\text{chroma}}_{bw} = 502.1$, $\Delta z^{\text{chroma}}_{\text{blackandwhite}} = 379.0$ and $\Delta z^{\text{chroma}}_{blackwhite} = 230.1$, respectively. The unequal values are a consequence of the different sample sizes $N_{bw} = 30294$, $N_{blackandwhite} = 17092$ and $N_{blackwhite} = 6157$. The compensated values are $\Delta z^{\text{chroma}*}_{bw} = 63.5$, $\Delta z^{\text{chroma}}_{blackandwhite}{}^* = 64.3$ and $\Delta z^{\text{chroma}}_{blackwhite}{}^* = 65.4$, respectively. All three values are relatively equal, which is in accordance with the fact that they express the same semantic concept.

3.3.2 Examples and Implementation on a Large Scale

Figure 4 shows $\Delta z_w{}^*$ values for different combinations of characteristics and 50 selected keywords w The scores are intuitively clear; *night* relates strongly to the gray-level histogram as the respective images tend to be very dark. *Blue* and *flower* have strong correspondence with hue and chroma characteristics. Spatial layouts are significant for the keywords *sunrise* and *sunset* as they have a distinct spatial distribution of colors. The keywords *macro*, *flower* and *bokeh* strongly relate to high frequency content as these images often have a blurred background. However, there are also keywords that do not show strong correspondence with the tested characteristics, e.g. *happy* or *day*. Thus, our framework allows us to explicitly test if a given keyword has a predominant corresponding image characteristic or not.

This is important for image applications in general, as the absence of a significant characteristic implies that a given algorithm will not affect these images. With regards to our application, image enhancement, the algorithm will not try to automatically improve images where the keyword indicates that a certain characteristic is not important.

[3]We superpose a regular 8×8 grid over the image independent of its size or aspect ratio. Then we compute for each grid cell the average value of the respective characteristic, leading to a 64-dimensional layout descriptor.

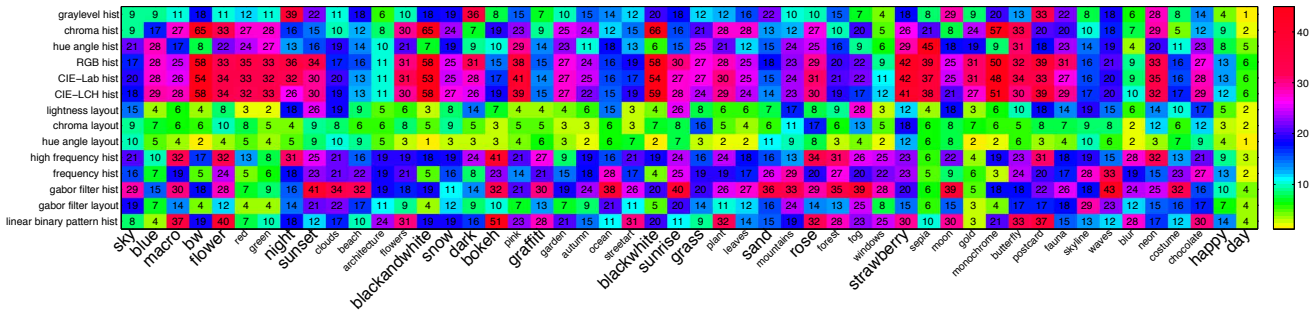


Figure 4: Δz^* values for 50 keywords and 14 characteristics. Note how the different keywords correspond to different characteristics. For instance, *bw* strongly equates with the chroma histogram (absence of high chromatic colors), *sunset* has a distinct spatial lightness layout (bright center and dark surrounding), and *graffiti* strongly relates to an image's high frequency content and linear binary patterns. *Day* and *happy* have very weak correspondence to any of the tested characteristics. Keywords that are referred to in the article have a larger font size. Access a complete listing of all 2858 keywords: http://ivrg.epfl.ch/SemanticEnhancement.html

If the one million example images are always the same, their characteristics can be computed and sorted offline. Then, to compute the ranksum statistic for a keyword, we need only to sum the corresponding elements' ranks in the pre-sorted list. Computing this indexed sum takes 35.9ms for 16 z values (e.g. gray-level histogram) on a MacBook Pro (2.5GHz Core 2 Duo). The code is written in Matlab and the core functions are implemented as mex-files. The main bottleneck of our current implementation is the query for a given keyword, as we parse text files with regular expressions in Matlab. This takes 50s per keyword, but we are confident that a standard MySQL implementation will reduce this time significantly.

4. SEMANTIC IMAGE ENHANCEMENT

The general framework presented in the previous section can be used for any application where image characteristics have to be linked to image semantics. In this publication, we focus on semantic image enhancement, which aims at re-rendering an image to adapt to a given semantic context. We define re-rendering as taking as input an image that has been processed in-camera or even enhanced afterwards and that we process to better visually match a semantic concept.

The proposed image enhancement is based on two components: 1) the image content as defined by the pixel values 2) the image semantics as described by a keyword. The first component uses standard image processing techniques. The novelty is the combination with the second component to make the processing semantically adaptive. We use the significance values in order to assess whether changing a characteristic is meaningful and if yes, how it has to be changed for an optimal adaption to the semantic context.

The significance values offer great potential to automatize semantic image processing, because they indicate whether a keyword and a characteristic are correlated. Keywords with lower significance values can be automatically discarded (e.g. *happy* or *day* as shown in Fig. 4) as they are not meaningful in terms of image processing. Also, we can automatically detect when images are "wrongly" annotated, i.e. no region in the image has significant characteristics corresponding to a particular keyword.

In the following, we present three semantic image enhancement algorithms beginning with a re-rendering for gray levels in Section 4.1. The efficiency and superiority over other commonly used methods is demonstrated with psychophysical experiments and described in Section 4.1.3. We further propose a method to re-render the colors in an image as shown in Section 4.2. Finally, we demonstrate in Section 4.3 the usage of the statistical framework for a very different type of enhancement: altering an image's frequencies in order to create artistic blurring effects that match the image's semantics.

4.1 Semantic Gray-Level Re-Rendering

For the first re-rendering application, a gray-level tone-mapping curve is computed that accounts for the image's semantic context. It is a global operation that maps an input pixel's gray level to a new gray level in the output image and thus alters the image's gray-level distribution.

4.1.1 Assessing a Characteristic's Required Change

To re-render an image for a specific semantic concept, its characteristic needs to be changed according to the two previously mentioned components: semantic context and image content. Hence we define two conditions that need to be fulfilled in order to alter the gray-level distribution: 1. the characteristic is significant for the semantic concept (i.e., high Δz as shown in Fig. 2); 2. the characteristic in the present image is too low or too high for the given concept. An image will not be altered if the characteristic is not influenced by the keyword or if the image is already a good example for it.

The first component is the significance of the semantic concept and is assessed via the z value from Equation 2. If the z value is positive (negative), the value of the corresponding characteristic has to be increased (decreased). We assume a linear relationship between the z values and the strength of the image processing; meaning that if the z value's absolute value is k times higher, the processing is k times stronger.

The second component is image dependent. We assess how well the given image already fulfills the desired characteristics for its semantic concept. We compare the image's characteristics to the characteristics of all images with the same keyword. Therefore, we compute the difference to a quantile:

$$\delta_{I,w}^{j} = \begin{cases} \max\left[0, Q_{1-p}\left(\mathcal{C}_w^j\right) - C_I^j\right] & \text{if } z_w^j \geq 0 \\ \max\left[0, C_I^j - Q_p\left(\mathcal{C}_w^j\right)\right] & \text{if } z_w^j < 0 \end{cases} \quad (4)$$

where $\delta^j_{I,w}$ signifies the difference measure for input image I with keyword w under characteristic j, C^j_I is image I's characteristic j, $Q_p(\cdot)$ measures a set's p-quantile and $\mathcal{C}^j_w$ are all characteristics j of images annotated with w.

If we use the 50% quantile $Q_{0.5}$ to compute the difference in equation 4, the second condition is already fulfilled ($\delta = 0$) if the input image's characteristic is average for its semantic concept. If, however, we want to emphasize the significant characteristics more, a lower quantile has to be chosen. We found that a 25%-quantile is a good tradeoff between a desired enhancement and an extreme overshooting, which would happen for quantiles in the order of 5%.

Similarly to the dependency on the z values, we implement a linear relationship between the δ values and the strength of the enhancement. Thus, the image processing has to be proportional to the product of z and δ values.

Figure 5 shows an example for the semantic concepts *dark* and *snow*. The example input image is the one from Figure 6 that is annotated with both keywords. The top part shows the median and quantiles of the distributions of $\mathcal{C}^j_{dark}$ and $\mathcal{C}^j_{snow}$ and the input image's characteristics. The middle part shows the δ value calculated from Equation 4. The bottom part shows the product of z (significance related) and δ (image related) values. The values indicate that, for the concept *snow*, the image needs fewer dark and more bright pixels, whereas for concept *dark* it needs more dark pixels.

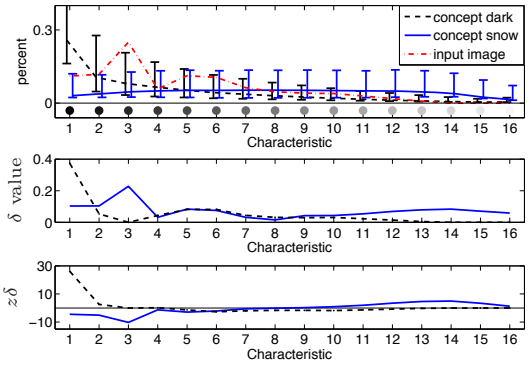


Figure 5: Top: Median and quantiles of the distributions for concepts *dark* and *snow* together with the used input image's characteristics (Fig. 1 right image). Middle: δ values according to Equation 4. Bottom: Product of z and δ values. For keyword *snow*, the image needs fewer dark pixels ($z\delta < 0$) and more bright pixels ($z\delta > 0$); the opposite is true for keyword *dark*.

4.1.2 Building a Tone-Mapping Function

We use the z value from Equation 2 and the δ value from Equation 4 to determine a tone-mapping of an image's gray levels. According to our previous assumptions, the product $z\delta$ is proportional to the change a processing introduces to an image.

In the case of a tone-mapping function, the strength is given by its slope. If at gray-level g, the slope is $m(g)$, the pixels in the interval around g are redistributed to a gray-level interval of $m(g)$ times the size. This holds for $m > 1$ (decreasing density) as for $m < 1$ (increasing density). A slope equal to one is the identity transform. As the $z\delta$ value indicates how strongly a characteristic has to be altered, the slope is:

$$m = \begin{cases} 1/(1 + Sz\delta) & \text{if } z\delta \geq 0 \\ 1 + S|z\delta| & \text{if } z\delta < 0 \end{cases} \quad (5)$$

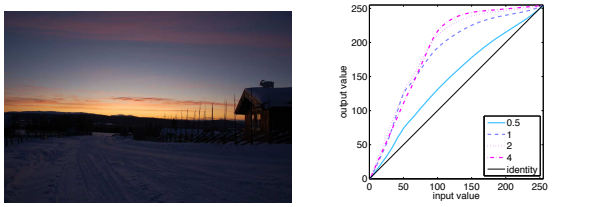
where S is a proportionality constant that controls the overall strength of the tone-mapping.

Extreme slope values are not desirable. A very steep mapping increases quantization artefacts and noise in homogeneous areas, and a very flat mapping reduces local contrast. Thus, the slope is cropped to a range $[1/m_{max} \;\; m_{max}]$. This is an inherent problem for any tone-mapping applications [18] and not specific to this approach. We used $m_{max} = 5$, which is a good compromise between limiting extreme tone-mappings and allowing visible changes.

The slope values from Equation 5 are linearly interpolated for 256 values in the interval [0 255] by using the representative mean gray level of each characteristic. Because these values specify the slope, they are the derivative of the tone-mapping function. An integration thus yields the desired function.

Due to the continuity of the slope values, the mapping function is continuous and differentiable. This guarantees a certain smoothness constraint that is beneficial for non-invasive processing. In a final step, we scale the mapping function to the interval [0 255] in order to maintain the image's black and white points.

The graph in Figure 6 shows tone-mapping functions for different proportionality constants S for keyword *snow*. The smaller the S is, the closer the mapping function is to the identity transform, which is depicted by the thin black line. Higher S values lead to a more extreme mapping. The three images show the input and the output for $S = 0.5$ and $S = 2$.



(a) input (b) tone-mapping, *snow*

(c) $S = 0.5$ (d) $S = 2$

Figure 6: Top: Input image and tone-mapping function to increase the semantic concept *snow* derived from the $z\delta$ values in Figure 5 ($S \in \{0.5\ 1\ 2\ 4\}$). Bottom: Two output images for $S = 0.5$ and $S = 2$, respectively.

4.1.3 Psychophysical Experiments

We evaluate the semantic gray-level enhancement with two psychophysical experiments. The first experiment shows that the semantically enhanced version is better than the

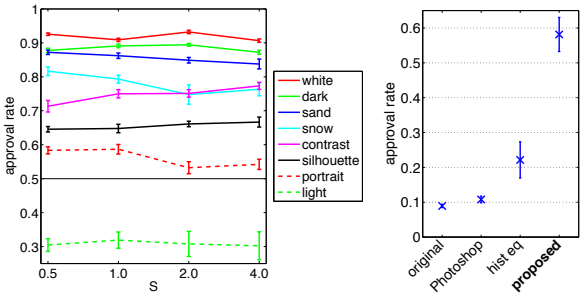
original image and in the second experiment we demonstrate that our algorithm outperforms other gray-level enhancement algorithms.

For the first experiment, we choose eight keywords with relatively high z values because low z values intentionally generate tone-mapping curves close to identity (see Eq. 5). The keywords w and their corresponding Δz_w values are *white* (88), *dark* (130), *sand* (72), *snow* (108), *contrast* (39), *silhouette* (79), *portrait* (80), and *light* (131), respectively.

For each keyword we selected 30 images from Flickr that have been annotated with the respective keyword, and we semantically re-rendered them with four different parameters $S \in \{0.5, 1, 2, 4\}$. Thus, we tested 960 images in total.

We set up a large-scale experiment using Amazon Mechanical Turk, where we showed the original and the enhanced image next to each other together with the corresponding keyword in the title. We asked 30 observers to select the image that best matches the keyword and payed them 1 cent per comparison.

Figure 7(a) shows the results of the psychophysical experiment. The S parameter is plotted on the horizontal axis and the approval rate for the enhanced image on the vertical axis. The approval values for all parameters S and all, except one, keywords are above 50%. Overall, the enhanced images are preferred and images in the *white* category have the highest rate (93%). This is not surprising as it is directly related to the gray-level characteristics.



(a) 1st experiment (b) 2nd experiment

Figure 7: Results from two psychophysical experiments. Left: Approval rates from 30 observers for different S values. Images of all except one semantic concepts are enhanced with success rates of up to 93%. The approval rate for *light* jumped to 62% in another experiment where we invited only artists. Right: Approval rate from 40 observers comparing the proposed method against other contrast enhancement methods (histogram equalization and Photoshop's auto contrast function). The proposed method scores more than 2.5 times better than the 2nd best. The error bars in both figures show the variances across different images.

The approval rate for images with *light* is surprisingly low and the variances are relatively large, which is due to the fact that there are two interpretations for this semantic term: 1) the image is bright in general 2) the image shows a light source that is visually important due to the dark surrounding. We reason that photographers and artists have rather the second point of view and carried out another experiment. We invited 20 photographers to judge the 30 images with keyword *light* ($S = 1$) and the resulting approval rate significantly jumped to 62% in favor of our algorithm.

The second psychophysical experiment compares our semantic image enhancement against other image only based contrast enhancements. We used four versions of each image, which were the original and three enhanced versions from Photoshop's auto contrast, Matlab's histogram equalization and our semantic framework for $S = 1$, respectively. In order to show the benefit of our semantically adaptive image enhancement we selected all images of the previous experiment that were annotated with more than two of the eight keywords. In total there are 29 such cases as the examples for *sand* and *dark* in Figure 1. The images were judged by 40 non-expert observers.

Figure 7(b) visualizes the results: 58.1% voted for our version, 22.2% for the histogram equalization, 10.8% for Photoshop's auto contrast and 8.9% for the photographer's original. The variances across the different images is shown in the form of vertical error bars. We see that our semantic enhancement has significantly higher approval rates and scores on average more than 2.5 times better than the 2nd best method. This is due to the fact that our semantic enhancement is the only method able to adapt to an image's semantic context.

4.2 Semantic Color Transfer

On the same lines as the gray-level tone-mapping, we can implement a semantic color transfer. As before, this requires two components that adapt to the image keyword and to the image pixels, respectively. The goal is to emphasize the colors in an image that are related to a given semantic concept.

However, it is important not to apply a global color shift to the entire image as this would look unnatural in certain image regions, such as a human face or a blue sky. Therefore, the image dependent component (δ in the gray-level case) has to be spatially varying in the color case.

This requirement is accounted for with a spatial weight map ω that encodes how much each pixel belongs to the semantic concept (see Fig. 8(b)). The map is simply the z value for each pixel color $\mathrm{col}(p)$ at position p in the image under the given semantic concept w. To assure smooth transitions, the map is blurred with a Gaussian blurring kernel with a sigma σ of 1% of the image diagonal. Further, the 5% and 95% quantiles ($Q_{0.95}$ and $Q_{0.05}$) are linearly mapped to 0 and 1, respectively, to remove noise.

$$\tilde{\omega} = g_\sigma * z_w\big(\mathrm{col}(p)\big), \qquad \forall p \in \text{image plane} \tag{6}$$

$$\omega = \min\left(1, \frac{\max\big(0, \tilde{\omega} - Q_{0.05}\big)}{Q_{0.95} - Q_{0.05}}\right) \tag{7}$$

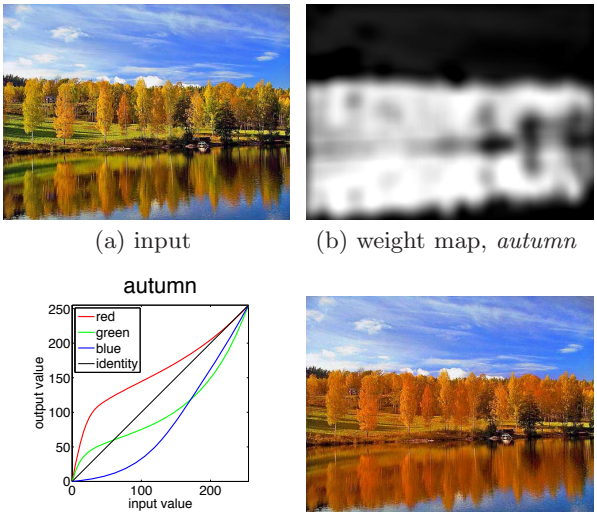
The semantic component is again based on z values, but this time with an $8 \times 8 \times 8$ histogram in sRGB color space. The tone-mapping curve is derived as before with Equation 5 and for each color channel separately. The only difference is that the δ value is omitted as this is accounted for by the weight map (Fig. 8(b)). The three tone-mapping curves derived for the semantic concept of *autumn* are reproduced in Figure 8(c).

We apply the derived tone-mapping on each color channel of the input image I_{in} resulting in a globally processed image I_{tmp}. As explained before, the final output has to show processed pixels only in those regions that belong to

the semantic concept. The output image I_{out} is thus a linear combination of the input image and the intermediary globally processed image I_{tmp}, and the weights are taken from the weight map ω:

$$I_{\text{out}} = (1 - \omega) \cdot I_{\text{in}} + \omega \cdot I_{\text{tmp}} \quad (8)$$

The resulting output image I_{out} is reproduced in Figure 8(d). Note that the image does not have a global color cast, but the semantic concept is emphasized only in image regions that are already part of it in the input image. Figure 9 shows another example, but for the semantic concept of *grass*.



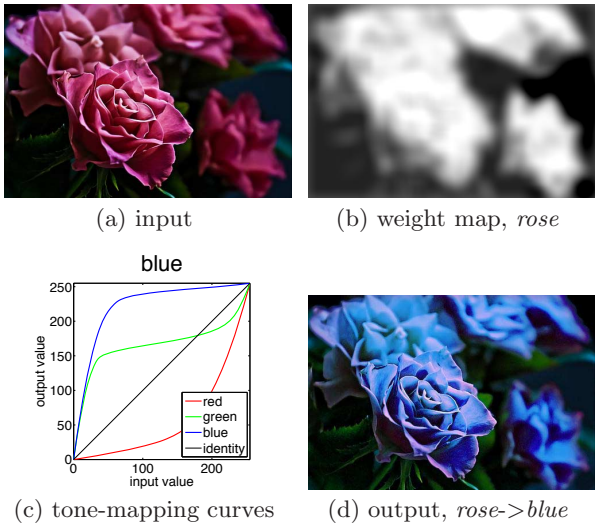
(a) input (b) weight map, *autumn*

(c) tone-mapping curves (d) output, *autumn*

Figure 8: Top: input image and associated weight map for the semantic concept of *autumn*. The bright regions in the map indicate regions that belong to the concept in the input image. Bottom row: Tone-mapping curves for the three color channels and the final output image. The semantic concept is emphasized only in those regions that already belong to it in the input image; other regions remain unprocessed.

Figure 9: Other example of the semantic color transfer for keyword *grass*.

Additionally, our algorithm for the semantic color transfer is able to handle different semantic concepts for the tone-mapping curves and the weight map, respectively. Hence, it can be used to exchange two semantic concepts in an image. Figures 10(a) and 10(b) show an image of a rose and the associate weight map for the concept of *rose*. However, the tone-mapping we apply stems from the keyword *blue*, as shown in Figure 10(c). Figure 10(d) shows the output image in which the roses are colored in blue. This is similar to other color transfer methods [26, 15]. Note, however, that our method handles an arbitrary semantic expression.



(a) input (b) weight map, *rose*

(c) tone-mapping curves (d) output, *rose->blue*

Figure 10: color transfer using two different semantic concepts. Top: input image and weight map for the semantic concept of *rose*. Bottom: tone-mapping curves for semantic concept *blue* and final output image. The roses are colored in blue.

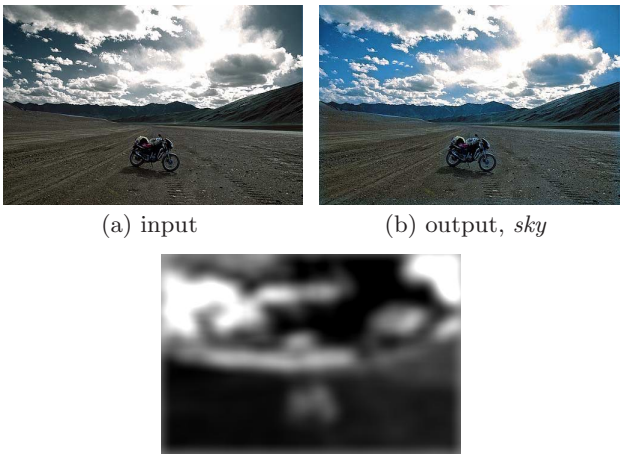
Figure 11 shows a failure case for the semantic concept of *sky*. The algorithm re-colored the cloud in the upper right corner in blue, which looks unnatural. The reason for the failure is an erroneous weight map as shown in Figure 11(c). Our detection method to find regions that correspond to a semantic concept is based on colors only. In this case the very dark clouds are classified as part of *sky* because it correlates also with dark grays in the MIR Flickr database. A more robust image region detection should improve results, but this is out of the scope of this publication.

4.3 Semantic Defocus Magnification

Defocus magnification is important in cases where a photographer intends an artistic blur of the background in order to accentuate the object in focus. In order to demonstrate the versatility of the presented statistical framework, we show how the significance values can be used in this context.

To account for the semantics, we compute z values describing the spatial frequency content in the Fourier domain. We do not distinguish between different orientations and thus obtain a radially averaged one-dimensional descriptor with 16 bins. The first bin describes the DC component and the lowest frequencies and the following bins describe increasing frequencies, respectively. The example plot in Figure 12(a) shows that the keyword *macro* relates to an absence of high frequencies, as indicated by the negative z values.

As we do not want to alter the brightness of the image, we shift the curve up with an additive constant so that the first z value (representing the DC component) is equal to



(a) input (b) output, *sky*

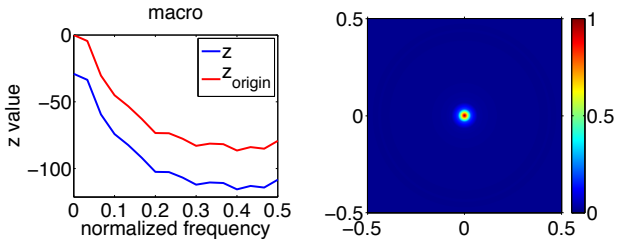
(c) weight map, *sky*

Figure 11: Failure case for the semantic concept of *sky*. The cloud in the top right corner has been mistaken for *sky* and re-colored in blue. The reason is an erroneous weight map based on color information. Other computer vision techniques can improve detection results, but this is out of the scope of this publication.

zero. These shifted values are denoted z_{origin} as their graph starts at the origin. We then compute the necessary change in the frequency domain similar to Equation 5:

$$F = \begin{cases} 1/\left(1 + S \cdot |z_{\text{origin}}|\right) & \text{if } z_{\text{origin}} < 0 \\ 1 + S \cdot z_{\text{origin}} & \text{if } z_{\text{origin}} \geq 0 \end{cases} \tag{9}$$

where S is a proportionality constant that controls the overall strength, and F is the filter in the Fourier domain. In order to multiply it with the Fourier transform of an image we generate a radially symmetric version with a simple linear interpolation as shown in Figure 12(b) for $S = 1$.



(a) significance values (b) F

Figure 12: Left: z and z_{origin} values for semantic concept *macro*. The negative values indicate an absence of high frequencies. Right: Corresponding multiplier in the Fourier domain computed with Eq. 9 and $S = 1$; it has a strong low-pass behavior.

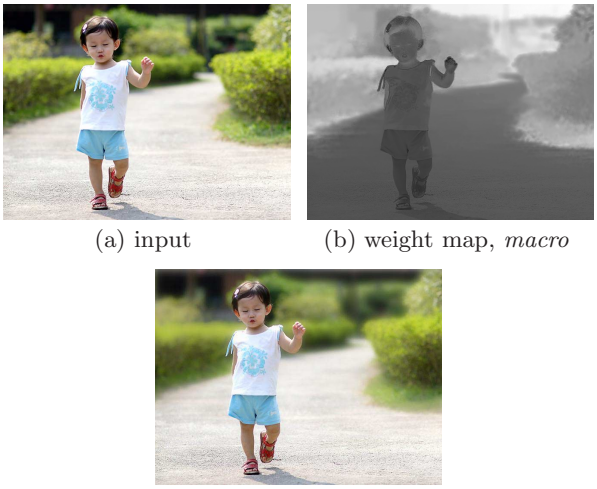
Similar to our two previous image enhancement examples, we not only implement a semantic component, but also an adaption to the input image itself. In this case, we need a map indicating regions with only low frequency content as it is done in defocus estimation [30]. Figures 13(a) and 13(b) show an image and its corresponding defocus map reproduced from Zhuo and Sim [30].

We compute an intermediary image I_{tmp} using the input image I_{in} and the filter F:

$$I_{\text{tmp}} = \mathscr{F}^{-1}\Big(\mathscr{F}(I_{\text{in}}) \cdot F\Big) \tag{10}$$

where $\mathscr{F}(\cdot)$ and $\mathscr{F}^{-1}(\cdot)$ denote the Fourier transform and its inverse, respectively.

We again use a linear weighting of the images I_{in} and I_{tmp} (Eq. 8), where the weights are taken from the defocus map. The final output is shown in Figure 13(c). Note that the background is more blurred than in the input image, whereas the boy remains in focus.



(a) input (b) weight map, *macro*

(c) output, *macro*

Figure 13: Example for the semantic concept *macro*. Top row: input and associated weight map. Images reproduced from Zhuo and Sim [30]. Bottom: Output image; note that the background is more blurred whereas the boy remains in focus.

5. CONCLUSIONS

We have presented a framework that learns associations between image characteristics and image keywords from a large database of annotated images [7]. The associations are quantified as a significance value from a simple non-parametric significance test. The framework needs no parameter tuning or optimization of a cost function, which makes it easy to scale to millions of images and thousands of semantic concepts.

The associations are then used for three semantic image enhancement methods where the goal is to re-render an image based on its semantic context. All methods are based on two components: 1) semantic context as described by the keywords 2) numeric content as described by pixel values. The novelty is the first component that enables an adaptation to an arbitrary semantic context; conventional enhancement methods lack this dimension.

We demonstrate three enhancement algorithms that all adapt to a given semantic context (a keyword): 1) gray-level tone-mapping to semantically adjust an image's gray-level distribution 2) semantic color transfer to emphasize a given

context 3) semantic defocus magnification creating artistic blur effects expressed by an associated keyword. Example images for the three semantic enhancements are reproduced in Figure 1 and throughout Section 4.

We demonstrate with psychophysical experiments that our semantic gray-level enhancement outperforms other enhancement methods that are based on pixel values only. Moreover, our method interprets semantic expressions from an uncontrolled vocabulary as opposed to other methods that restrict the user to a small number of predefined choices [26, 15].

Due to the simplicity and computational speed of the framework – even at large scales – we see potential to use it in other image-related areas that include semantics, such as annotation, labeling, or retrieval. For this reason we make the code publicly available for research purposes under `http://ivrg.epfl.ch/SemanticEnhancement.html`.

For future work we propose to use computer vision techniques in order to estimate more robust weight maps (see Figure 11). Also, assuming a linear relationship between the significance values and the processing strength proved to be good (Sec. 4.1.3), but other non-linear dependencies can be investigated to further improve the performance. It is also interesting to investigate other types of image enhancement algorithms that exploit different characteristics and validate them with more psychophysical experiments.

6. REFERENCES

[1] S. Bae and F. Durand. Defocus magnification. *Eurographics*, 26(3):571–579, 2007.

[2] G. Ciocca, C. Cusano, F. Gasparini, and R. Schettini. Content aware image enhancement. In *Artificial Intelligence and Human-Oriented Computing*, volume 4733/2007, pages 686–697, Rome, September 2007.

[3] D. Cohen-Or, O. Sorkine, R. Gal, T. Leyvand, and Y.-Q. Xu. Color harmonization. *ACM Trans. Graphics*, 25(3):624–630, 2006.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: a large-scale hierarchical image database. In *CVPR*, pages 248–255, Miami, 2009.

[5] T. Deselaers and V. Ferrari. Visual and semantic similarity in imagenet. In *CVPR*, pages 1777–1784, 2011.

[6] C. Fredembach. Saliency as compact regions for local image enhancement. In *CIC*, pages 14–18, 2011.

[7] M. J. Huiskes and M. S. Lew. The MIR flickr retrieval evaluation. In *ACM MIR*, 2008.

[8] R. Hummel. Image Enhancement by Histogram Transformation. *CGIP*, 6(2):184–195, 1977.

[9] ISO 11664-4:2008(E)/CIE S 014-4/E:2007. *CIE Colorimetry - Part 4: 1976 L*a*b* Colour Space*, 2007.

[10] S. B. Kang, A. Kapoor, and D. Lischinski. Personalization of image enhancement. In *CVPR*, pages 1799–1806, 2010.

[11] T. Mäenpää. *The Local Binary Pattern Approach To Texture Analysis – Extensions and Applications*. PhD thesis, University of Oulu, 2003.

[12] 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*, 18(1):50–60, 1947.

[13] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[14] S. Moser and M. Schroeder. Usage of DSC meta tags in a general automatic image enhancement system. In *EI*, volume 4669, pages 259–267, San Jose, CA, USA, January 2002.

[15] N. Murray, S. Skaff, and L. Marchesotti. Towards automatic concept transfer. In *ACM SIGGRAPH/Eurographics*, pages 167–176. ACM Press, 2011.

[16] V. Namboodiri. Recovery of relative depth from a single observation using an uncalibrated (real-aperture) camera. In *CVPR*, pages 1–6, 2008.

[17] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: describing images using 1 million captioned photographs. In *NIPS*, 2011.

[18] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *JCVGIP*, 39:355–368, 1987.

[19] A. Polesel, G. Ramponi, and V. J. Mathews. Image enhancement via adaptive unsharp masking. *TIP*, 9(3):505–510, March 2000.

[20] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE Computer Graphics and Applications*, 21(5):2–9, 2001.

[21] C. Sauvaget and V. Boyer. Harmonic colorization using proportion contrast. In *AFRIGRAPH*, pages 63–69, 2010.

[22] N. Sawant, J. Li, and J. Z. Wang. Automatic image semantic interpretation using social action and tagging data. In *Multimedia Tools and Applications*, volume 51, pages 213–246, 2011.

[23] I. K. Sethi, I. Coman, and D. Stan. Mining association rules between low-level image features and high-level concepts. In *Data Mining and Knowledge Discovery III*, volume 4384, pages 279–290, 2001.

[24] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: a large dataset for non-parametric object and scene recognition. *Trans. PAMI*, 30(11):1958–1970, 2008.

[25] R. Walpole, R. Myers, and S. Myers. *Probability and Statistics*, volume 6. Prentice Hall International, 1998.

[26] B. Wang, Y. Yu, T.-T. Wong, C. Chen, and Y.-Q. Xu. Data-driven image color theme enhancement. In *SIGGRAPH Asia*, volume 29, 2010.

[27] B. Wang, Y. Yu, and Y.-Q. Xu. Example-based image color and tone style enhancement. In *SIGGRAPH Asia*, volume 30, 2011.

[28] X.-J. Wang, W.-Y. Ma, and X. Li. Exploring statistical correlations for image retrieval. *Multimedia Systems*, 11(4):340–351, 2006.

[29] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.

[30] S. Zhuo and T. Sim. Defocus map estimation from a single image. *Pattern Recognition*, 44(9):1852–1858, 2011.