

# Compact color texture descriptor based on rank transform and product ordering in the RGB color space

Antonio Fernández

Universidade de Vigo

School of Industrial Engineering, Campus Universitario Lagoas Marcosende, 36310 Vigo, Spain

<http://antfdez.webs.uvigo.es/>

David Lima

Universidade de Vigo

School of Industrial Engineering, Campus Universitario Lagoas Marcosende, 36310 Vigo, Spain

[davidlimaastor@hotmail.com](mailto:davidlimaastor@hotmail.com)

Francesco Bianconi

Università degli Studi di Perugia

Department of Engineering, Via G. Duranti 93, 06125 Perugia, Italy

<http://dismac.dii.unipg.it/bianco/>

Fabrizio Smeraldi

Queen Mary, University of London

School of Electronic Engineering and Computer Science

Mile End Road, London E1 4NS, United Kingdom

<http://www.eecs.qmul.ac.uk/~fabri/>

## Abstract

*Color information is generally considered useful for texture analysis. However, an important category of highly effective texture descriptors – namely rank features – has no obvious extension to color spaces, on which no canonical order is defined. In this work, we explore the use of partial orders in conjunction with rank features. We introduce the rank transform based on product ordering, that generalizes the classic rank transform to RGB space by a combined tally of dominated and non-comparable pixels. Experimental results on nine heterogeneous standard databases confirm that our approach outperforms the standard rank transform and its extension to lexicographic and bit mixing total orders, as well as to the preorders based on the Euclidean distance to a reference color. The low computational complexity and compact codebook size of the transform make it suitable for multi-scale approaches.*

## 1. Introduction

Color texture analysis is routinely used in many computer vision applications such as image-based medical diagnosis, land cover classification from satellite imagery, automated quality inspection, grading of agricultural and food products, etc. It has been shown by many authors that taking into account color information generally improves the performance – at least under steady illumination conditions – see for instance [30] and [6]. In the last two decades rank features have emerged as one of the most robust tools for image analysis. Examples of this approach are the well-known local binary patterns and their variants [7, 29], rank transform [38], texture spectrum [18], ranklets [35], directional rank coding [34] and sudoku rank features [17], all of which rely on a “greater than” relation between pairs of gray level values. The extension of such methods to color images has so far been limited by the absence of a natural ordering in the color spaces, where statements like *pink is greater than brown* don’t make any sense.

In this work we present a novel approach to color texture analysis based on rank features in color space. The key idea

is to replace the order relation over intensity values by a suitable partial order in color space. By doing so we admit that some pairs of colors are not comparable, as will become apparent later. We present an implementation of this idea by extending a classic method, the rank transform (RT), to work with a partial ordering (the product order) in the RGB space. The result is a compact yet highly effective local descriptor for color images.

In the remainder of the paper, after providing some background on partial orders (Sec. 2), we briefly review how rank-based descriptors have been defined in the literature for both grayscale and color texture analysis (Sec. 3). We present the proposed approach in Sec. 4 and validate the method through a set of classification experiments over nine datasets of color texture images (Sec. 5). The results clearly show that our method is more effective at discriminating texture than the original rank transform, either based on gray levels, or on different orders or preorders on the RGB color space. Section 6 concludes the paper and discusses directions for future work.

## 2. Theoretical background

We recall in this section some basic notions about order relations and briefly review the existing approaches for the ordering of multi-dimensional data.

### 2.1. Order relations

A *total order*  $\leq$  is a binary relation on a set  $X$  for which the following four properties hold for any three elements and  $a$ ,  $b$  and  $c$  of  $X$ :

- Antisymmetry:  $a \leq b \wedge b \leq a \implies a = c$
- Transitivity:  $a \leq b \wedge b \leq c \implies a \leq c$
- Totality:  $a \leq b \vee b \leq a$
- Reflexivity:  $a \leq a$

Totality establishes that any two elements of  $X$  are comparable; it also implies reflexivity, as can be seen by setting  $a = b$ . By discarding totality for the weaker requirement of reflexivity we obtain a *partial order*. This differs from a total order in that there will be elements of  $X$  that are not comparable to each other.

Finally, a *preorder* or *quasiorder* is a binary relation that is both reflexive and transitive (but not necessarily total or antisymmetric). The lack of antisymmetry means there can be cases where both  $a$  weakly dominates  $b$  and  $b$  weakly dominates  $a$  without  $a$  and  $b$  being equal; see Sec. 3.2 for an example. Total orders and partial orders are special cases of a preorder. Tab. 1 summarizes the definitions just introduced.

Table 1. Overview of order relationships.

	Preorder	Partial order	Total order
Reflexivity	✓	✓	✓
Transitivity	✓	✓	✓
Antisymmetry	–	✓	✓
Totality	–	–	✓

### 2.2. Ordering multi-dimensional data

The application of rank features to color or multispectral images is anything but straightforward for multidimensional data lack a natural way of ordering. Research has so far focused on *restricted* ordering principles or sub-orderings. These can be broadly divided into the four categories summarized below [2].

**Marginal ordering (M-ordering)** Multivariate ordering consists of sorting the data separately by one or more components. Lexicographical ordering is an example of M-ordering: for instance in the RGB space one could sort the color data first by R, then by G and finally by B (or following any other sequence obtained by a permutation of the channels). The major shortcoming of this approach, however, is that colors close to each other in the color space can end up very far apart after the ordering.

**Reduced (aggregate) ordering (R-ordering)** In this case multivariate data are preliminarily converted to univariate through some *ad hoc* functions. One could for instance establish a reference point (*pivot*) in the multivariate space and sort the data by their distance to it.

**Conditional (sequential) ordering (C-ordering)** In C-ordering the members of a random multivariate sample are sorted according to the corresponding values of another (usually univariate) random sample. An example of C-ordering is provided by statistically equivalent blocks, which consists of splitting a multidimensional space into subspaces or blocks through the use of one-dimensional cutting functions [32].

**Partial ordering (P-ordering)** Partial ordering differs from total ordering in that the totality condition is not required: in practice we accept that there will be pairs of elements incomparable to each other. We have already introduced partial ordering in Sec. 2.1; we shall discuss the partial ordering used in this work (*i.e.* product order) in Sec. 4.2.

Interestingly, most approaches to deal with order in color space belong to the first two categories. For instance, rank-

ing based on the gray level can be either seen as a marginal ordering in the HSV space (along the V axis) or as a reduced or aggregate ordering in the RGB space, where the aggregating function is the standard color to grayscale conversion.

### 3. Rank-based texture features

Rank features are defined in terms of the relative order (rank) among pixels, with the actual pixel values being discarded. By definition rank features are invariant to any monotonic change in the illumination intensity, a feature that makes them particularly suited for dealing with noise and sensor non-linearities [20].

Though most of the rank-based descriptors proposed in the literature are intended for grayscale image analysis, a number of methods for color images have also been described. We briefly review the two classes in the following subsections.

#### 3.1. Grayscale analysis

Local image descriptors based on pixel intensity can either rely on a full ranking or thresholding. In the first case neighboring pixels are ranked according to their grayscale value and features are extracted from the rank signature. Examples of this approach are: rank transform [38], complete rank transform [15, 10], ranklets [35], n-tuples [19], directional rank coding [34] and local intensity order pattern [37]. In the second case features are computed from pairwise comparisons of pixel values. Most descriptors belonging to this group can actually be considered as LBP variants [7, 29].

#### 3.2. Color analysis

The extension of grayscale rank-based methods to color images usually involves computing the ranking on each color channel separately and/or on pairs of color channels [6]. A classic example are the opponent-color local binary patterns proposed by Mäenpää and Pietikäinen [31]. A similar approach was introduced by Bianconi *et al.* [5] to compute rotation-invariant color texture features from ranklets [35]. More involved procedures to compute intra- and inter-channel features have been proposed by Lee *et al.* [27] (local color vector binary patterns), and, more recently, by Cusano *et al.* [12] (local angular patterns).

An alternative strategy is that of inducing an arbitrary ordering (typically M- or R-ordering) on the multivariate color data. Ledoux *et al.* [26] is an up-to-date, thorough investigation on this class of methods. In the remainder of this section we describe the three approaches that we considered for the comparative study, *i.e.*: *lexicographic order*, *bit mixing* and *reference color*.

**Lexicographic order** This well-known color comparison relation is a total order based on priorities between the components of a vector [1]. Using the priorities  $R \triangleright G \triangleright B$  where  $\triangleright$  means “has priority over”, a color  $\mathbf{c}_0 = (r_0, g_0, b_0)$  in the RGB space is less than or equal to a color  $\mathbf{c}_1 = (r_1, g_1, b_1)$  according to the following rule:

$$\mathbf{c}_0 \leq \mathbf{c}_1 \iff (r_0 < r_1) \vee [(r_0 = r_1) \wedge (g_0 < g_1)] \vee [(r_0 = r_1) \wedge (g_0 = g_1) \wedge (b_0 \leq b_1)] \quad (1)$$

In the experiments we considered the following priority sequences:  $R \triangleright G \triangleright B$ ,  $B \triangleright G \triangleright R$ , and  $G \triangleright B \triangleright R$ . Obviously, the main shortcoming of this approach is that the color order induced strongly depends on how the priorities are defined.

**Bit mixing** This is a total order widely used in multivalued morphological filtering [11]. This paradigm associates an integer value to each color from the binary representation of their chromatic components. For convenience let us denote a color in the RGB space as  $\mathbf{c} = (c_0, c_1, c_2)$  and the  $n$ -th bit of the  $k$ -th component as  $c_{k,n}$ , with  $0 \leq k \leq 2$  and  $0 \leq n \leq 7$  (we have implicitly assumed that intensity is discretized to 8-bits). We can build up a 24-bit integer by mixing the bits of the three chromatic components like this:

$$N(\mathbf{c}) = \sum_{n=0}^7 \left[ 2^{3n} \sum_{k=0}^2 2^k c_{\pi(k),n} \right] \quad (2)$$

where  $\pi$  represents a permutation of the chromatic channels (represented by their indices  $\{0, 1, 2\}$ ). The bit mixing order is then defined as:

$$\mathbf{c}_0 \leq \mathbf{c}_1 \iff N(\mathbf{c}_0) \leq N(\mathbf{c}_1) \quad (3)$$

In the experiments we took into account the following bit-mixing permutations: RGB, BGR, and GBR. Similarly to the lexicographical order, the bit mixing order strongly depends on how the color components are permuted.

**Reference color** This preorder relies on the Euclidean distance between a color and a reference color, referred to as  $\mathbf{c}_{\text{ref}}$ . The relation that compares two colors  $\mathbf{c}_0$  and  $\mathbf{c}_1$  with respect to the reference color is defined as [25]:

$$\mathbf{c}_0 \preceq \mathbf{c}_1 \iff \|\mathbf{c}_0 - \mathbf{c}_{\text{ref}}\| \leq \|\mathbf{c}_1 - \mathbf{c}_{\text{ref}}\| \quad (4)$$

Herein we used two reference colors: black (0, 0, 0) and a shade of gray (127, 127, 127). Needless to say that the reference color should be judiciously chosen since this choice may have a great impact on the resulting order.

## 4. Proposed approach

In the following subsections we provide a description and mathematical formulation of the image descriptor proposed in this paper. But before getting into the technicalities of the method we recall the basics of the rank transform (Sec. 4.1) and product order (Sec. 4.2).

### 4.1. Rank transform

The rank transform was first proposed by Zabih and Woodfill [38]. In this model the image is scanned through a sliding window of  $N \times N$  pixels and the feature value at each position is the number of pixels in the periphery of the local neighborhood the intensity of which is less than that of the central pixel. In the original implementation  $N = 3$ , but other values are also possible (see Fig. 1). The kernel function that induces an equivalence relationship between local patterns [16] can be written as:

$$f_{\text{RT}}(\mathbf{x}) = \sum_{j=0}^{P-1} \xi(I_c - I_j - 1) \quad (5)$$

where  $\mathbf{x}$  represents the set of the pixels' intensities in the neighborhood,  $I_c$  the intensity of the central pixel,  $I_j$  that of a generic peripheral pixel and  $P = 4 \times (N - 1)$  the total number of the peripheral pixels. Symbol  $\xi(x)$  indicates the binary thresholding function:

$$\xi(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \quad (6)$$

Clearly  $f_{\text{RT}}$  can return any integer in the  $[0, P]$  interval, which results in a set of  $P + 1$  possible patterns. Image features are the dense, orderless distribution over the set of the possible patterns. Note that the features so calculated are intrinsically invariant to rotation and to any permutation of the peripheral pixels.

### 4.2. Product order

Given an ordered set  $X$ , one can induce a partial ordering on the Cartesian product  $X \times X \times X$  [14]. This is referred to as the product order (also called component-wise order). If  $X$  is the set of intensity levels of one channel of an RGB image (for an 8-bit intensity scale such set would be formed by the integers from 0 to 255) this can be extended to order colors in a straightforward manner. Given two triplets  $\mathbf{c}_0 = (r_0, g_0, b_0)$  and  $\mathbf{c}_1 = (r_1, g_1, b_1)$  representing the RGB coordinates of two pixels, the product order relation establishes that:

$$\mathbf{c}_0 \preceq_{\times} \mathbf{c}_1 \iff r_0 \leq r_1 \wedge g_0 \leq g_1 \wedge b_0 \leq b_1 \quad (7)$$

Conversely, when two colors are not comparable we shall write:

$$\mathbf{c}_0 \approx_{\times} \mathbf{c}_1 \iff \neg(\mathbf{c}_0 \preceq_{\times} \mathbf{c}_1) \wedge \neg(\mathbf{c}_1 \preceq_{\times} \mathbf{c}_0) \quad (8)$$

Note that in the above equations we use symbol  $\preceq$  rather than  $\leq$  to emphasize the fact that this binary relation is a partial order, and subscript  $\times$  to signal that this particular partial order is the product order.

It is instructive to show that the product order in the RGB space has an intuitive geometric interpretation. Let us consider a generic color  $\mathbf{c}_0 = (r_0, g_0, b_0)$  in the RGB space as shown in Fig. 2, where points  $(0, 0, 0)$  and  $(1, 1, 1)$  respectively indicate colors 'black' and 'white'. We have that:

- Any color in the (closed) green region *dominates*  $\mathbf{c}_0$
- any color in the (closed) brown region *is dominated by*  $\mathbf{c}_0$
- any other color outside the two above regions is incomparable to  $\mathbf{c}_0$

Though the product order (as well as other types of partial orders) can be applied to any color space, in this work we restrict the study to partial ordering in the RGB space.

### 4.3. Rank transform based on product order (RTPO)

Let  $\phi$  be a function that takes in two RGB triplets and returns 1 when the first one is dominated by the second as per the product order and 0 otherwise, and  $\psi$  another function that flags (with a 1) the situation in which the arguments are not comparable:

$$\phi(\mathbf{c}_j, \mathbf{c}_c) = \begin{cases} 1, & \text{if } \mathbf{c}_j \prec_{\times} \mathbf{c}_c \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

$$\psi(\mathbf{c}_j, \mathbf{c}_c) = \begin{cases} 1, & \text{if } \mathbf{c}_j \approx_{\times} \mathbf{c}_c \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

where  $\mathbf{c}_c$  is the RGB triplet corresponding to the central pixel and  $\mathbf{c}_j$ ,  $0 \leq j \leq P - 1$ , is the RGB triplet of the  $j$ -th peripheral pixel. Notice that in Eq. 10 we replaced the symbol  $\preceq_{\times}$  introduced in Eq. 7 by  $\prec_{\times}$  to be consistent with the strict inequality employed in the original implementation of the rank transform [38]. Those functions are handy to compute the number of peripheral pixels that are dominated by ( $\lambda$ ) and not comparable to ( $\nu$ ) the central pixel:

$$\lambda = \sum_{j=0}^{P-1} \phi(\mathbf{c}_j, \mathbf{c}_c) \quad (11)$$

$$\nu = \sum_{j=0}^{P-1} \psi(\mathbf{c}_j, \mathbf{c}_c) \quad (12)$$

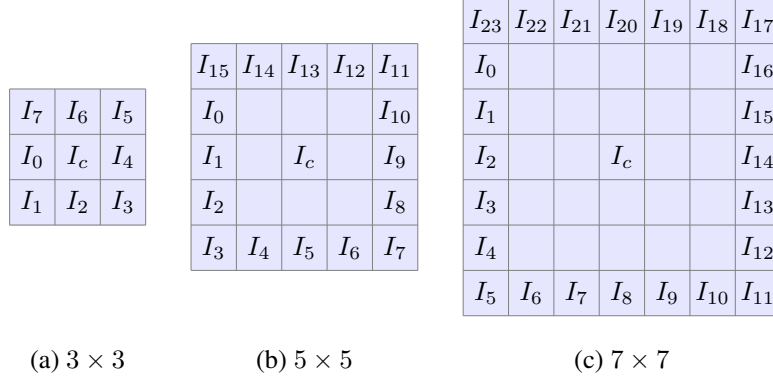


Figure 1. Different pixel neighborhoods considered in this study.

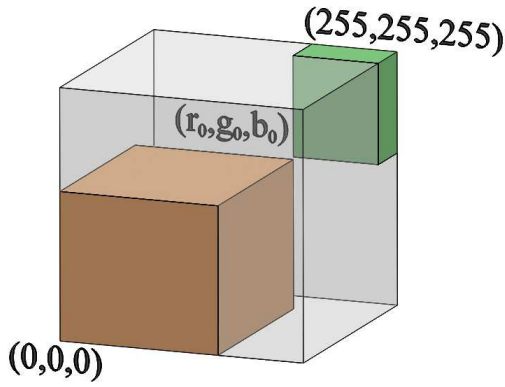


Figure 2. Volumes induced in the RGB color space by triplet  $(r_0, g_0, b_0)$  and the product order.

Both are non negative integer numbers, and are constrained to  $0 \leq \lambda + \nu \leq P$ . The kernel function that assigns a label to a local pattern can be expressed as a function of those numbers:

$$f_{\text{RTPO}}(\mathbf{x}) = \nu + \sum_{i=0}^{\lambda-1} \sum_{k=0}^{P-i-1} 1 = \nu + \frac{\lambda(2P+3-\lambda)}{2} \quad (13)$$

where  $\mathbf{x}$  in this case represents an array containing the RGB triplets of the pixels in the local neighborhood.

To further illustrate the kernel above, in Tab. 2 we display the pattern labels for a  $3 \times 3$  neighborhood as a function of  $\lambda$  and  $\nu$ . Clearly the number of possible features  $F$  can be computed as the number of ways to distribute  $n$  unlabelled balls into  $k$  labelled urns [3, Th. 4.2.6]; the balls being the peripheral pixels and the urns the possible outcomes of the comparison with the central pixel, *i.e.*: dominates, is dominated or non comparable. Since  $n = P$  and  $k = 3$  we have:

Table 2. Kernel of the rank transform based on product order as a function of  $\lambda$  and  $\nu$  for a  $3 \times 3$  neighborhood

		$\nu$								
		0	1	2	3	4	5	6	7	8
$\lambda$	0	0	1	2	3	4	5	6	7	8
	1	9	10	11	12	13	14	15	16	
	2	17	18	19	20	21	22	23		
	3	24	25	26	27	28	29			
	4	30	31	32	33	34				
	5	35	36	37	38					
	6	39	40	41						
	7	42	43							
	8	44								

$$F = \binom{n+k-1}{k-1} = \binom{P+2}{2} = \frac{(P+2)(P+1)}{2} \quad (14)$$

Thus the dimensions of the histograms that result from the neighborhoods of size  $3 \times 3$ ,  $5 \times 5$  and  $7 \times 7$  are 45, 153 and 325, respectively.

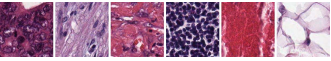

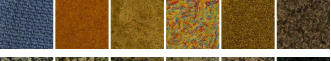

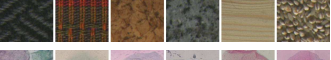
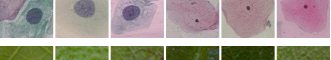



## 5. Experimental results

To assess the effectiveness of the proposed approach we ran a set of image classification experiments on nine datasets of color texture images from different sources. Tab. 3 summarizes the main properties of each dataset along with references/URLs.

Our primary goal was to comparatively assess, using a set-up as simple as possible, the performance of RTPO against that of the RT based on gray levels (original RT) and on the orders and preorders described in Sec. 3.2. To this end we used a simple  $3 \times 3$  square neighborhood of pixels and the nearest neighbor classifier (1-NN) with Euclidean distance. The use of this classifier is standard practice in comparative studies – see for instance [22, 28, 13] –



Table 3. Datasets used in the experiments: round-up table

ID	Name	Classes	Samples per class	Sample size (px)	Variations in imaging conditions	Sample images	Ref.
1	Kather	8	625	$150 \times 150$	None		[23]
2	KTH-TIPS2b	11	432	$200 \times 200$	4 $\text{\textcircled{V}}$ , 9 $\text{\textcircled{Q}}$ , 3 $\text{\textcircled{E}}$		[8]
3	Kylberg-Sintorn	25	216	$1728 \times 1728$	9 $\text{\textcircled{R}}$		[24]
4	MondialMarmi	25	40	$1500 \times 1500$	10 $\text{\textcircled{R}}$		[4]
5	Outex-13	68	20	$128 \times 128$	None		[33]
6	PapSmear	2	204	$67 \times 92 - 300 \times 768$	Unspecified		[21]
7	PlantLeaves	20	60	$128 \times 128$	None		[9]
8	RawFoot	68	184	$400 \times 400$	46 $\text{\textcircled{V}}$		[13]
9	STex	476	16	$128 \times 128$	Unspecified		[36]

KEY TO SYMBOLS:  $\text{\textcircled{V}}$ = illumination,  $\text{\textcircled{R}}$ = rotation,  $\text{\textcircled{Q}}$ = scale,  $\text{\textcircled{E}}$ = viewpoint

due to the absence of tuning parameters, ease of implementation and other desirable asymptotic properties. Accuracy estimation was based on split-half validation with stratified sampling over 100 random splits. For each subdivision into train and test set we computed the fraction of samples of the test set correctly classified and averaged the results over the 100 splits.

The results obtained are summarized in the box plot of Fig. 3. As can be seen, in most datasets RTPO outperformed the other methods by a large margin. We also notice that the overall accuracy varied a lot from one dataset to another: this is due to the fact that the image datasets differ from each other in a number of significant factors such as the overall number of classes, the imaging conditions and the stationariness of the textures. To test whether there was significant difference between the accuracy of RTPO and that of the other descriptors we performed a two-sided Wilcoxon signed-rank test at significance level  $\alpha = 1\%$ . The results (Tab. 4) show that RTPO significantly outperformed the other descriptors in eight datasets out of nine, whereas in the remaining one the difference was very slight ( $<0.1\%$ ).

We also performed a second group of experiments to determine how much the accuracy of RTPO features could be improved by using a multi-resolution approach and more sophisticated classifiers such as decision trees and support vector machines. This experiment included datasets ‘Kather’, ‘KTH-TIPS2b’, ‘Outex-13’ and ‘RawFoot’. The

multi-scale approach consisted of concatenating the feature vectors resulting from employing the three different neighborhoods shown in Fig. 1. Optimal parameters  $C$  and  $\gamma$  for the rbf-SVM classifier were estimated, for each descriptors and image dataset, through exhaustive grid search. The parameters of the decision tree classifier (*i.e.*: maximum depth of the tree, minimum number of samples required to split an internal node and minimum number of samples required to be at a leaf node) were determined in a similar fashion. In the case of decision trees the dimension of the feature vectors was preliminary reduced through PCA before classification.

The results of this second group of experiments are graphically displayed in Fig. 4. Here the labels ‘1’, ‘2’ and ‘3’ respectively denote: the features obtained with the  $3 \times 3$  neighborhood, the concatenation of the features obtained with the  $3 \times 3$  and  $5 \times 5$  neighborhoods and the concatenation of the features obtained with the  $3 \times 3$ ,  $5 \times 5$  and  $7 \times 7$  neighborhoods. The total number of features by the three configurations respectively is 45, 198 and 523.

The plots in Fig. 4 clearly show that the overall accuracy significantly improved by adopting the multi-resolution approach and SVM classification. Decision trees, however, did not provide any benefit compared with nearest-neighbor classification.

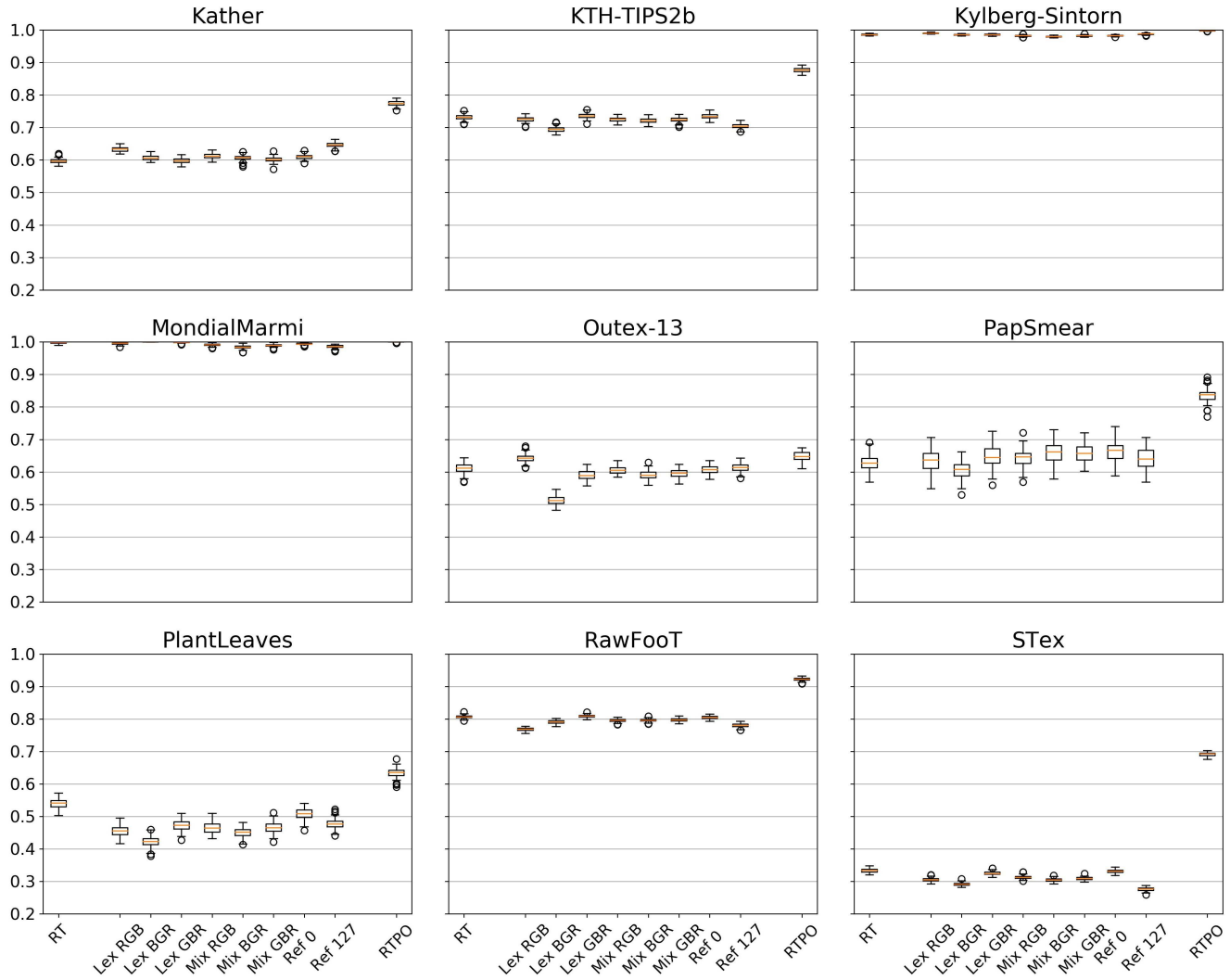


Figure 3. Box plots of the success rates obtained on the different datasets. RT: rank transform on gray levels; Lex: RT on lexicographic orders; Mix: RT on bit mixing orders; Ref: RT on reference color preorders; RTPO: the proposed enhanced RT over the product order.

Table 4. Results of the Wilcoxon signed-rank test at significance level 1%. For all the datasets the rank transform based on the product order is compared to the best performing benchmark descriptor.

ID	Name	Best preorder	Mean accuracy		p-value	Significant?
			Best preorder	Partial order		
1	Kather	Ref 127	64.67%	<b>77.40%</b>	3.88e-18	Yes
2	KTH-TIPS2b	Lex GBR	73.57%	<b>87.71%</b>	3.88e-18	Yes
3	Kylberg-Sintorn	Lex RGB	99.07%	<b>99.80%</b>	3.84e-18	Yes
4	MondialMarmi	Lex BGR	<b>100.00%</b>	99.94%	1.83e-05	Yes
5	Outex-13	Lex RGB	64.26%	<b>64.84%</b>	3.74e-03	Yes
6	PapSmear	Ref 0	66.24%	<b>83.48%</b>	3.85e-18	Yes
7	PlantLeaves	Ref 0	50.74%	<b>63.41%</b>	3.88e-18	Yes
8	RawFoot	Ref 0	80.51%	<b>92.29%</b>	3.89e-18	Yes
9	STex	Ref 0	33.14%	<b>69.15%</b>	3.89e-18	Yes

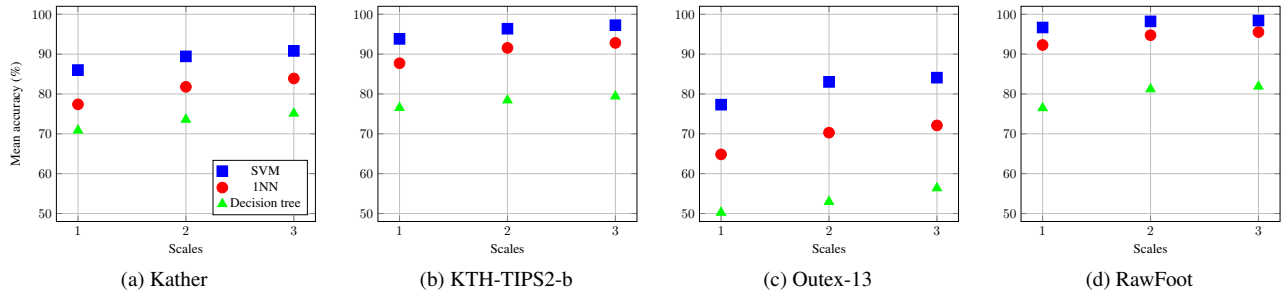


Figure 4. Results obtained through multi-scale analysis and different classifiers.

## 6. Conclusions and future work

In this paper we have introduced RTPO (rank transform based on product order), a compact yet effective local image descriptor for color textures. Our approach generalizes a classic descriptor (rank transform) to RGB color images via a particular type of partial order, *i.e.* the product order.

In a set of color texture classification experiments RTPO proved more effective than the original rank transform – either based on grayscale on other arbitrary orders and pre-orders on the RGB space. We found that the improvement was statistically significant. The highest success rates were obtained with a multi-scale approach and SVM classification. Remarkably, the proposed method provides a compact representation of color texture, the number of features for square neighborhoods of size  $3 \times 3$ ,  $5 \times 5$  and  $7 \times 7$  respectively being 45, 153 and 325.

As far as we know the use of partial orders is entirely novel in this context. The results established so far seem very promising and encourage further investigation. In particular, we plan to utilize the product order with other rank features to make them suitable for color texture analysis. Another direction for future research is the investigation of other types of partial orders (different than the product order) that may better integrate color information into rank features.

## References

- [1] E. Aptoula and S. Lefèvre. A comparative study on multivariate mathematical morphology. *Pattern Recognition*, 40:2914–2929, 2007. 3
- [2] V. Barnett. The ordering of multivariate data. *Journal of the Royal Statistical Society. Series A (General)*, 139(3):318–355, 1976. 2
- [3] R. A. Beeler. *How to count. An introduction to combinatorics and its applications*. Springer, 2016. 5
- [4] R. Bello-Cerezo, F. Bianconi, A. Fernández, E. González, and F. Di Maria. Experimental comparison of color spaces for material classification. *Journal of Electronic Imaging*, 25(6):061406–1–10, 2016. 6
- [5] F. Bianconi, A. Fernández, E. González, and J. Armesto. Robust colour texture features based on ranklets and discrete Fourier transform. *Journal of Electronic Imaging*, 18(4):043012–1–8, 2009. 3
- [6] F. Bianconi, R. Harvey, P. Southam, and A. Fernández. Theoretical and experimental comparison of different approaches for color texture classification. *Journal of Electronic Imaging*, 20(4):043006–1–17, 2011. 1, 3
- [7] S. Brahmam, L. C. Jain, L. Nanni, and A. Lumini, editors. *Local binary patterns: new variants and applications*, volume 506 of *Studies in computational intelligence*. Springer, 2014. 1, 3
- [8] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV’05)*, volume II, pages 1597–1604, 2005. 6
- [9] D. Casanova, J. J. de Mesquita Sá Jr., and O. Martinez Bruno. Plant leaf identification using Gabor wavelets. *International Journal of Imaging Systems and Technology*, 19(3):236–243, 2009. 6
- [10] C. H. Chan, F. Yan, J. Kittler, and K. Mikolajczyk. Full ranking as local descriptor for visual recognition: A comparison of distance metrics on  $s_n$ . *Pattern Recognition*, 48:1328–1336, 2015. 3
- [11] J. Chanussot and P. Lambert. Bit mixing paradigm for multivalued morphological filters. In *Sixth International Conference on Image Processing and Its Applications*, volume 2, pages 804–808, Dublin, Ireland, July 1997. IET. 3
- [12] C. Cusano, P. Napoletano, and R. Schettini. Local angular patterns for color texture classification. In *ICIAP 2015: New Trends in Image Analysis and Processing – ICIAP 2015 Workshops*, volume 9281 of *Lecture Notes in Computer Science*, pages 111–118. Springer, 2015. 3
- [13] C. Cusano, P. Napoletano, and R. Schettini. Evaluating color texture descriptors under large variations of controlled lighting conditions. *Journal of the Optical Society of America A*, 33(1):17–30, 2016. 5, 6
- [14] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, Cambridge, second edition, 2002. 4
- [15] O. Demetz, D. Hafner, and J. Weickert. The complete rank transform: A tool for accurate and morphologically invariant matching of structures. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2013. 3



- [16] A. Fernández, M. X. Álvarez, and F. Bianconi. Texture description through histograms of equivalent patterns. *Journal of Mathematical Imaging and Vision*, 45(1):76–102, 2013. 4
- [17] G. Finlayson and S. Nixon. Sudoku texture classification. In *Proceedings of the IS&T International Symposium on Electronic Imaging Science and Technology. Image Processing: Machine Vision Applications IX*, pages 1–5, San Francisco, CA, USA, February 2016. 1
- [18] D.-C. He and L. Wang. Texture unit, texture spectrum, and texture analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 28(4):509–512, 1990. 1
- [19] L. Hepplewhite and T. J. Stonham. Texture classification using N-tuple pattern recognition. In *Proceedings of the 13th International Conference on Pattern Recognition (ICPR'96)*, volume 4, pages 159–163, August 1996. 3
- [20] R. M. Hodgson, D. G. Bailey, M. J. Naylor, A. L. M. Ng, and S. J. McNeill. Properties, implementations and applications of rank filters. *Image and Vision Computing*, 3(1):3–14, 1985. 3
- [21] J. Jantzen, J. Noras, G. Dounias, and B. Bjerregaard. Papsmear benchmark data for pattern classification. In *Proceedings of NiSIS 2005: Nature inspired Smart Information Systems (NiSIS), EU co-ordination action*, pages 1–9, Albufeira, Portugal, 2005. 6
- [22] U. Kandaswamy, S. A. Schuckers, and D. Adjeroh. Comparison of texture analysis schemes under nonideal conditions. *IEEE Transactions on Image Processing*, 20(8):2260–2275, 2011. 5
- [23] J. N. Kather, C.-A. Weis, F. Bianconi, S. M. Melchers, L. R. Schad, T. Gaiser, A. Marx, and F. G. Zöllner. Multi-class texture analysis in colorectal cancer histology, 2016. 6
- [24] Kylberg Sintorn Rotation dataset, 2013. Available online at <http://www.cb.uu.se/~gustaf/KylbergSintornRotation/>. Last accessed on Apr. 6, 2017. 6
- [25] A. Ledoux, R. Noël, A.-S. Capelle-Laizé, and C. Fernandez-Maloigne. Toward a complete inclusion of the vector information in morphological computation of texture features for color images. In A. Elmoataz, O. Lezoray, F. Nouboud, and D. Mammass, editors, *Proceedings of the 6th International Conference on Image and Signal Processing (ICISP 2014)*, volume 8509 of *Lecture Notes in Computer Science*, pages 222–229, Cherbourg, France, 2014. Springer International Publishing. 3
- [26] A. Ledoux, O. Losson, and L. Macaire. Color local binary patterns: Compact descriptors for texture classification. *Journal of Electronic Imaging*, 25(6):061404–1–12, 2016. 3
- [27] S. Lee, J. Choi, Y. Ro, and K. Plataniotis. Local color vector binary patterns from multichannel face images for face recognition. *IEEE Transactions on Image Processing*, 21(4):2347–2353, 2012. 3
- [28] L. Liu, L. Zhao, Y. Long, G. Kuang, and P. W. Fieguth. Extended local binary patterns for texture classification. *Image and Vision Computing*, 30(2):86–99, 2012. 5
- [29] L. Liu, P. Fieguth, Y. Guo, X. Wang, and M. Pietikäinen. Local binary features for texture classification: Taxonomy and experimental study. *Pattern Recognition*, 62:135–160, 2017. 1, 3
- [30] T. Mäenpää and M. Pietikäinen. Classification with color and texture: jointly or separately? *Pattern Recognition*, 37(8):1629–1640, 2004. 1
- [31] T. Mäenpää and M. Pietikäinen. Texture analysis with local binary patterns. In C. H. Chen and P. S. P. Wang, editors, *Handbook of Pattern Recognition and Computer Vision (3rd Edition)*, pages 197–216. World Scientific Publishing, 2005. 3
- [32] K. E. Matthews and M. S. Taylor. Nonparametric methods for multivariate analysis using statistically equivalent blocks. Technical Report ARL-TR-992, U.S. Army Research Laboratory, April 1996. 2
- [33] T. Ojala, M. Pietikäinen, T. Mäenpää, J. Viertola, J. Kyllönen, and S. Huovinen. Outex - new framework for empirical evaluation of texture analysis algorithms. In *Proceedings of the 16th International Conference on Pattern Recognition (ICPR'02)*, volume 1, pages 701–706, Quebec, Canada, 2002. IEEE Computer Society. 6
- [34] F. Ouslimani, A. Ouslimani, and Z. Ameer. Directional rank coding for multi-scale texture classification. In *Proceedings of the 2016 International Conference of Computing for Engineering and Sciences (ICCES'2016)*, pages 1–12, Barcelona, Spain, July 2016. 1, 3
- [35] F. Smeraldi. Ranklets: orientation selective non-parametric features applied to face detection. In *Proceedings of the 16th International Conference on Pattern Recognition (ICPR'02)*, volume 3, pages 379–382, Quebec, Canada, August 2002. IEEE Computer Society. 1, 3
- [36] Salzburg texture image database (STex), 2009. Available online at <http://wavelab.at/sources/STex/>. Last accessed on Jul. 7, 2015. 6
- [37] Z. Wang, B. Fan, and F. Wu. Local intensity order pattern for feature description. In *Proceedings of the 13th International Conference on Computer Vision (ICCV 2011)*, pages 603–610, Barcelona, Spain, November 2011. 3
- [38] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In *Proceedings of the 3rd European Conference on Computer Vision (ECCV 1994)*, pages 151–158, Stockholm, Sweden, May 1994. Springer-Verlag. 1, 3, 4