Document downloaded from:

http://hdl.handle.net/10251/201675

This paper must be cited as:

Agustí-Melchor, M.; Rodas Jordá, Á.; Valiente González, JM. (2011). Classification of Repetitive Patterns Using Symmetry Group Prototypes. Springer-Verlag Berlin Heidelberg. 84-91. http://hdl.handle.net/10251/201675

The final publication is available at

https://link.springer.com/chapter/10.1007/978-3-642-21257-4_11

Copyright Springer-Verlag Berlin Heidelberg

Additional Information

Classification of repetitive patterns using Symmetry Group Prototypes

M. Agustí-Melchor*, A. Rodas-Jordá, and J. M. Valiente-González

Computer Vision Group, DISCA, Escuela Técnica Superior de Ingeniería Informática,
Universidad Politécnica de Valencia, Camino de Vera, s/n 46022 Valencia, Spain
{magusti,arodas,jvalient}@disca.upv.es

Abstract. Symmetry is an abstract concept that is easily noticed by humans that designers make new creations based on its use. Images of these designs belong to a general group called wallpaper images that exhibit a repetitive pattern. We present a novel computational framework for automatic classification method by symmetries, based on Symmetry Group theory features for wallpaper images. The existing methods have several drawbacks because of the use of heuristics. These methods have shown low classification values when images exhibit imperfections due to the fabrication or the hand made process. Also, there is no way to give some computation of the *classification goodness-of-fit*. We propose to obtain an automatic parameter estimation for symmetry analysis. Thus, the image classification is redefined as distances computation to the prototypes of a set of defined classes. Our experimental results improves the state of the art in wallpaper classification methods.

Keywords: Symmetry, symmetry groups, prototype-based classification, adaptive nearest neighbour classification

1 Introduction

Symmetry is an abstract concept that is easily noticed by humans and as a result designers make new creations based on its use. In industrial sectors, the notion of symmetry is always present as an aesthetic element, indispensable in each new design. These designs, see examples in Fig. 1, are commonly referred as *regular mosaics*, *wallpaper images*, *wallpaper patterns*, or simply *Wallpapers*. Thousands of these samples accumulated over the years are stored in company storerooms or museums. But designers suffer serious limitations when searching and manage these format, and designers are accustomed to use other abstract terms related with perceptual criteria. Therefore, some kind of image analysis is necessary to extract information about the internal geometry and structure of patterns. Little effort has been made in their image analysis and classification, and so this work explores this direction.

A symmetry of any 2D pattern can be described through a set of geometrical transformations that transforms it in itself: isometries. These are: translations,

* This work is supported in part by spanish project VISTAC (DPI2007-66596-C02-01).

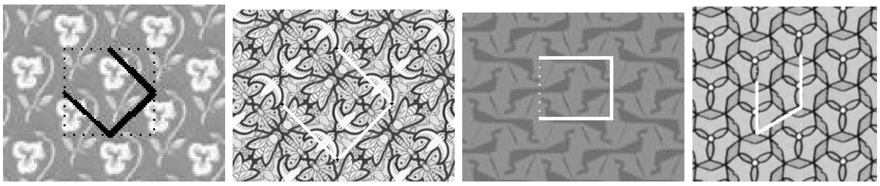

Fig. 1. Details of wallpaper images obtained from [10], [2], and [4] collections. These are images of real textile samples. The geometry of the UL is also shown.

rotations (n-fold), reflections (specular), and glide reflections (specular plus lateral displacement). Regarding translational symmetry, a wallpaper pattern is a repetition of a parallelogram shaped subimage, called *Unit Lattice* (UL). A lattice extraction procedure is then needed to obtain the lattice geometry. In this work we assume that the lattice geometry has been already obtained and the UL is known, e.g. using autocorrelation [5], Fourier or wavelets [1].

We rely on the Symmetry Groups theory [3] to formulate a mathematical description of structures. A symmetry group is the set of isometric transformations that brings a figure in coincidence with itself. In the 2D case, a *Plane Symmetry Groups* (PSG) is defined by means of two translational symmetries (lattice) and some other isometries. For example, Fig. 1 (left) has only translational. In contrast, the other patterns of Fig. 1 have more isometries, such as 180° rotations and reflections about two axes. The last pattern can be reconstructed using 120° rotations. The 'crystallographic constraint' limits the number of PSG to 17 cases, helping us to describe the pattern structure. Fig. 2 shows the details of each PSG as well as their standard notation. For example, the patterns in Fig. 1 belong, from left to right, to symmetry groups P1, PMM, PGG and P31M.

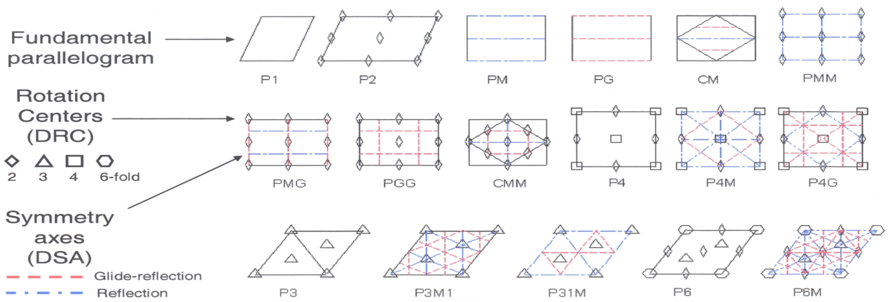

Fig. 2. Representation of the 17 Wallpaper Groups, their standard notation and their internal symmetries. The UL is referred as Fundamental Parallelogram.

The interest in the algorithmic treatment of symmetries, has been recognized by the recent tutorial at ECCV 2010 Conference [6]. It constitutes an extended discussion and comparison of state of the art Symmetry Detection algorithms. A global success of 63% over a test bed of 176 images is reported. The classical algorithms to catalogue wallpapers [3] are heuristic-based procedures to be used by humans. These are proposed in the form of decision trees whose branches are raised as questions to ask by looking at the design. Because of the ambiguities in the computer reformulation of these questions, their implementation is very complex. The main computer autonomous approach of this kind has been made by Liu et al. [5]. This model expresses the Schattschneider algorithm in the form of a *rule-based classifier* (RBC) where each symmetry group corresponds to a unique sequence of yes/no answers. It can be seen as a kind of *decision tree* classifier with binary symmetry values. Our experience confirms that this method can be tuned to obtain 100% hits for the Joyce [4] dataset, but this set-up is not successful for other image collections. In addition, the use of RBC obtains only one result without an associate measure of confidence. It becomes necessary to enhance this solution, as indicated in [7].

2 PROPOSED METHOD

We propose a novel computational framework based on continuous measures of symmetries for a distance-based classification into symmetry groups approach applied to real and synthetic images of two-dimensional repetitive patterns. The use of binary class prototypes describing the PSG mentioned above, adaptively adjusted on image conditions, assumed the high degree of variance of symmetry features, due to noise, deformations or just the nature of the hand made products. As this classification results in an ordered list of content similarity based on symmetry, it can be used as an result for Context-Based Image Recovery (CBIR) applications. We started by using a Nearest Neighbour Classifier (NNC) as this enabled us to obtain a measure of goodness for the classification. This kind of methods require the use of continuous-value feature vectors.

2.1 Feature computation and Symmetry Groups classification

A close view to PSG description in Fig. 2 reveals that the minimum number of symmetry features needed to distinguish every PSG is twelve: four (R_2, R_3, R_4, and R_6) related to rotational symmetries; four (T_1, T_2, T_{1G}, and T_{2G}) to describe reflection symmetries (non-glide and glide) along axes parallel to the sides of UL; and four more features (D_1, D_2, D_{1G}, and D_{2G}) for reflection (non-glide and glide) with respect to the two UL diagonals. We defined a symmetry feature vector (SFV) of twelve elements that identifies the presence/absence of these symmetries as $(f_1, f_2, ..., f_{12})$. To obtain a symmetry feature f_i for a specific isometry, e.g 2-fold rotation, we apply this transformation to the original image $I(x, y)$ obtaining the transformed image $I^T(x, y)$. A piece of the transformed image, of the size of the bounding box of the UL (m), is taken. A score map is then computed as $Map(x, y) = 1 - SAD(x, y)$, where:

$$SAD(x, y) = \frac{1}{m} \sum_{x_0 \ y_0} |I(x - x_0, y - y_0) - BBox(x_0, y_0)| \tag{1}$$

If symmetry is present in the image, this map peaks at several positions indicating the presence of that symmetry, while revealing lower values in areas where the symmetry is not hold. The $|maximum - minimum|$ difference should then be a good measure to quantify the feature. However, there are patterns without internal symmetries, such as P1 (Fig. 1), so that max-min difference should be relative to any other value representing the presence of symmetry. The only symmetry always present in every pattern is the translational symmetry (S_T). Finally, we compute the normalized components of the SFV as follows:

$$f_i = \frac{max(Map) - min(Map)}{S_T - min(Map)} \quad 1 \leq i \leq 12 \tag{2}$$

The higher the value of f_i, the more likely the image contains symmetry. Table 1 shows the SFV vectors obtained for the four wallpaper samples in Fig. 1. As expected, these results partially confirm high values that indicate the presence of symmetry and low values otherwise. The bold values means a value that has to be considered as presence of symmetry to consider each vector as the group that it belongs to, while the others mean absence of others symmetries. Because these features were computed as gray level differences between image patches, their values will strongly depend on the particular arrangements of image pixels: the *image complexity*. As a consequence SFV requires a higher level of adaptation to the image conditions, i.e. taking into account on the contents of each image separately. This idea will be used later by an adaptive NNC.

Table 1. Symmetry feature vectors of the four wallpapers showed in Fig. 1.

Sample	SFV=(R_2, R_3, R_4, R_6, T_1, T_2, D_1, D_2, T_{1G}, T_{2G}, D_{1G}, D_{2G})	PSG
1	(0.62, 0.47, 0.69, 0.34, 0.65, 0.67, 0.80, 0.59, 0.37, 0.43, 0.80, 0.59)	$P1$
2	(**0.82**, 0.09, 0.20, 0.09, **0.88**, **0.83**, 0.20, 0.19, 0.27, 0.26, 0.2, 0.19)	PMM
3	(**0.95**, 0.42, 0.33, 0.46, 0.39, 0.45, 0.31, 0.48, **0.98**, **0.99**, 0.31, 0.48)	PGG
4	(0.46, **0.69**, 0.28, 0.49, 0.74, 0.65, 0.48, **0.72**, **0.74**, **0.65**, 0.48, 0.72)	$P31M$

To classify a wallpaper image, featured by SFV, we need a set of class samples. Fortunately, the number of classes and their structure are known in advance. For the sake of simplicity, we start by proposing the use of binary prototypes representing each one of the classes. Table 2 shows the resulting 23 prototypes. Some classes have two prototypes because there are two possibilities where reflection symmetry can appear.

After applying the NNC to several image collections we did not found significant improvements in comparison with RBC (see the Experiments section). This is probably due to the bias of the feature values: minimum values are not near '0', nor maximum values are near S_t. In that situation, the use of binary prototypes, with inter-class boundaries equidistant to each class, does not fit the problem. However, some advantage has been achieved. First, the Euclidean distance to the class prototype can be used as a measure of confidence. Second, the NNC produces an ordered set of outputs describing the class membership of

Table 2. Binary prototypes for the 17 PSG classes.

Classes	Prototype Feature vectors
$P1$	(0,0,0,0,0,0,0,0,0,0,0,0)
$P2$	(1,0,0,0,0,0,0,0,0,0,0,0)
PM_1	(0,0,0,0,1,0,0,0,0,0,0,0)
PM_2	(0,0,0,0,0,1,0,0,0,0,0,0)
PG_1	(0,0,0,0,0,0,0,0,1,0,0,0)
PG_2	(0,0,0,0,0,0,0,0,0,1,0,0)
CM_1	(0,0,0,0,0,0,1,0,0,0,1,0)
CM_2	(0,0,0,0,0,0,0,1,0,0,0,1)
PMM	(1,0,0,0,1,1,0,0,0,0,0,0)
PMG_1	(1,0,0,0,1,0,0,0,0,1,0,0)
PMG_2	(1,0,0,0,0,1,0,0,1,0,0,0)
PGG	(1,0,0,0,0,0,0,0,1,1,0,0)

Classes	Prototype Feature vectors
CMM	(1,0,0,0,0,0,1,1,0,0,1,1)
$P4$	(1,0,1,0,0,0,0,0,0,0,0,0)
$P4M$	(1,0,1,0,1,1,1,1,0,0,1,0)
$P4G$	(1,0,1,0,0,0,1,1,1,1,1,0)
$P3$	(0,1,0,0,0,0,0,0,0,0,0,0)
$P31M_1$	(0,1,0,0,1,1,1,0,1,1,1,0)
$P31M_2$	(0,1,0,0,1,1,0,1,1,1,0,1)
$P3M1_1$	(0,1,0,0,0,0,1,0,0,0,1,0)
$P3M1_2$	(0,1,0,0,0,0,0,1,0,0,0,0)
$P6$	(1,1,0,1,0,0,0,0,0,0,0,0)
$P6M$	(1,1,0,1,1,1,1,1,1,1,1,0)

each sample. This latter consideration can enable an automatic adjustment of the prototypes in order to adapt them to the image variability.

2.2 Adaptive NNC (ANNC)

Recent works on NN classifiers have shown that adaptive schemes [9] outperform the results of classic NNC in many applications. In response to the ambiguities in computed symmetry values, we propose an adaptive approach based on establishing a merit function to adapt the inter-class boundaries to the specific image conditions. Fig. 3-a shows a simplified example of a 2D feature space including 4 binary prototypes. The inter-class boundaries are symmetric with respect to each prototype. In a real context, the $SFV(f_1, f_2)$ vectors never reach certain areas close to the prototypes, Fig. 3-b shows this forbidden areas. The distorted distances force to adapt the boundaries between classes.

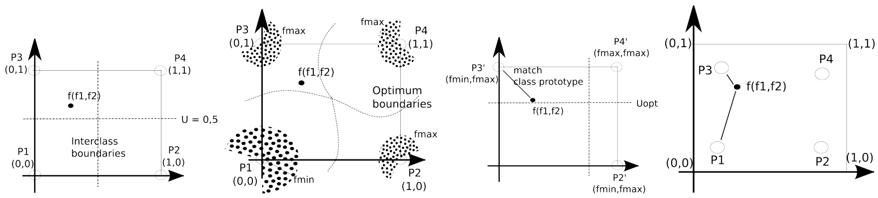

Fig. 3. From left to right: a) A 2D feature space and prototypes $P1$, $P2$, $P3$ and $P4$. b) Forbidden areas. c) Adaptation of class boundaries. d) Final disambiguation.

To do this, a transformation of the the feature space can be performed by normalizing these features. In this new space, the null-class $P1$ disappears, therefore this class should be treated separately. The new boundaries between classes can be searched in a way that maximizes a merit function. We use orthogonal boundaries defined by a single parameter U, the *Uncertainty Boundary of Symmetry*. We studied several merit functions and, finally, propose the distance ratio between the reported first and second classes after classifying the sample with respect to binary prototypes using a NN classifier. The result is the boundary U_{opt} that best separates the classes. Moreover, instead of moving the inter-class boundaries, the problem is reformulated to modify the class prototypes into new values $(H, L) \in [0, 1]$ that are symmetrical with respect to the middle value U (Fig. 3-c). Finally, the closest class to new prototypes (H, L) and the null-class $P1$ are disambiguated (Fig. 3-d). The algorithm is as follows:

Step 1 - The symmetry values are normalized, see Eq. 3, discharging the $P1$ class and resulting in a 16-class problem.

$$\begin{array}{c} SFV' = (f'_1, f'_2, ..., f'_{12}); \\ f'_i = \frac{f'_i - min(SFV)}{max(SFV) - min(SFV)}; \quad 1 \leq i \leq 12 \end{array} \tag{3}$$

Step 2 - The original prototypes are transformed into (H, L) prototypes for each of 16 classes. These values are defined with respect to parameter U as: $H = 1,\ L = 2 \cdot U - 1$ for $U \geq 0,5$ and $L = 0,\ H = 2 \cdot U$ otherwise.

Step 3 - For each U, ranging from 0.2 to 0.8, the H and L limits are computed and a NNC is performed using SFV' and the resulting prototypes. Repeating steps 2-3 for all U values, the value (U_{opt}) that maximizes the merit function is selected, and the corresponding class is also tentatively selected.

Step 4 - Finally, we disambiguate the candidate class from the previously excluded $P1$ class. To achieve this, we again re-classify the SFV but only using the $P1$ and candidate classes.

3 EXPERIMENTS

As indicated in [6], without a systematic evaluation of different symmetry detection and classification algorithms against a common image set under a uniform standard, our understanding of the power and limitations of the proposed algorithms remains partial. As image datasets reported in literature were not publicly available, we selected several wallpaper images from known websites, to carry out the comparison between the proposed ANNC and the reference RBC methods. We picked out image datasets from Wallpaper [4], Wikipedia [10], Quadibloc [8], and SPSU [2], resulting in a test bed of 218 images. All images were hand-labelled to make the ground truth. As the original RBC algorithm source code was unavailable, we implemented it using the original RBC decision tree reported in [5], but using our SFV feature vectors, and binarising the features using a fixed threshold (the average of the better classification results) for all image datasets. The results obtained with RBC, NNC and ANNC classifiers are shown in Table 3. For the shake of brevity, we only put here the percentage of successful classification, i.e. accuracy or precision results.

The first image collection is Wallpaper, a standard collection reported in previous works. In this case, both RBC and ANNC methods obtain a 100%

Table 3. Experimental classification results from RBC, NNC, and ANNC.

Collection	#img	Sub-set	RBC	NNC	ANNC	NNC2	ANNC2	NNC3	ANNC3
Wallpaper	17		100	82,35	100	100	100	100	100
Wikipedia	53		54,72	60,38	62,26	73,55	81,13	81,13	83,02
	17	WikiGeom	88,24	88,24	100	94,12	100	100	100
Quadibloc	46		71,74	69,57	82,61	82,61	91,30	91,3	95,63
	29	Quad0102	62,07	75,86	79,31	86,21	89,66	89,66	93,10
	17	Quad03	88,24	58,82	88,24	76,47	94,12	94,12	100
SPSU	102		49,02	62,76	59,80	71,57	76,47	76,47	82,35
Global	218		59,18	65,15	68,35	76,60	82,60	82,60	86,70

of success. The RBC achieves the same result as reported in [5], which means that our implementation of this algorithm has similar results as the original implementation. To take into account the varying complexity of the images, we separate the next two collections in sub-sets. In the WikiGeom dataset, which is a sub-set of Wikipedia formed by strongly geometrical patterns, the ANNC and NNC results outperformed the RBC.

In the case of the entire Wikipedia collection, which includes other distorted images, a decrease in results is evident. Similar results were obtained with Quadibloc image collection, which is of intermediate complexity. We studied it as two subsets: one formed by sketches over uniform background (Quad0102), and other (Quad03) is constituted by more complex motives with many highly contrasted colours. The ANNC obtains near 80% of success rate with this collections, clearly outperforming the NNC and RBC methods. The worse results were obtained with the more complex images in the SPSU collection. In this case, all results are below 60%. This lower values are due to the existence of noise and imprecise details (hand-made) in the images. Also, these exhibit several repetitions and illumination artifacts, which suggest the neccesity of pre-processing. It is remarkable that the ANNC algorithm is still 10 points up that RBC algorithm. The latest row gives a global hit success considering the number of images in each collection.

The fact of working with a distance-based classifier offers an additional advantage because it delivers a value defining the degree of proximity to the prototype chosen ($d_i = dist(SFV, P_i)$). This (P_i, d_i) description, which can be extended to the whole set of prototypes, can be used as a high level semantic image descriptor, useful in areas such as *Content Based Image Retrieval*. This is particularly helpful in the presence of complex images that, due to various factors (manufacturing, noise, damaged parts, small details, . . .), present an ambiguity about the symmetry group they belong to, exhibiting characteristics of several of them being even complicated their labelling by experts. Thus taking, the first two (NNC2 & ANNC2) or three (NNC3 & ANNC3) classification results, the success rates are considerably higher (see Table 3) and the distance to the second and third candidate are near the first result. That shows that many of the classification errors were due to the above-mentioned factors.

This idea can be conveniently exploited in the context of CBIR.

4 CONCLUSIONS

This paper had presented a proposal for a novel computational framework for classification of repetitive 2D pattern images into symmetry groups. The feature vector is composed of twelve symmetry scores, computationally obtained from the image gray level values. A main issue is the use of binary class prototypes to represent the 17 PSG classes. However, the absence of symmetry is never computed as '0', nor the presence of symmetry is computed as '1', even assuming perfect image conditions. The RBC and the NNC behave poorly, because of the ambiguities in symmetry computation. This leads to the use of some adaptive approach, implemented by an adaptive classifier. The ANNC is non-parametric, so there is no need to adjust the parameters involved, and it is also non-supervised, so no learning stages are needed. The experimental results show that the ANNC outperforms the other methods, even with very complex image collections.

As future work we are now on looking for a new way of computing the symmetry features, because the used approach seems to have a limited sensitivity. We are also extending the test beds, and the method to colour images. Moreover, the results can be useful in recovery tasks using an extended version of ANNC - which produces a list of similarity to every group that can be sorted from highest to lower values, and so for example, detect images that are near to several groups.

References

1. Agustí, M., Valiente, J. M., and Rodas, A. (2008). Lattice extraction based on symmetry analysis. In Proceedings of 3rd. International Conference on Computer Vision Applications (VISAPP-08), volume 1, pages 396..402.
2. Edwards, S. (2009). Tiling plane & fancy. http://www2.spsu.edu/math/tile/index.htm.
3. Horne, C. (2000). Geometric Symmetry in Patterns and Tilings. Woodhead Publishing, Abington Hall (England).
4. Joyce, D. (2007). Wallpaper groups (plane symmetry groups). http://www.clarku.edu/∼djoyce/.
5. Liu, Y., Collins, R., and Tsin, Y. (2004). A computational model for periodic pattern perception based on frieze and wallpaper groups. Trans. on PAMI, 26(3).
6. Liu, Y., Hel-Or, H., Kaplan, C. S., and Gool, L. V. (2010). Computational symmetry in computer vision and computer graphics. In Foundations and Trends in Computer Graphics and Vision, volume 5, pages 1..195.
7. Reddy, S. and Liu, Y. (2005). On improving the performance of the wallpaper symmetry group classification. CMU-RI-TR-05-49, Robotics Institute, CMU.
8. Savard, J. G. (-). Basic tilings: The 17 wallpaper groups. http://www.quadibloc.com/math/tilint.htm.
9. Wang, J., Neskovic, P., and Cooper, L. N. (2007). Improving nearest neighbor rule with a simple adaptive distance measure. Pattern Recognition Letters, 28(2):207..213.
10. Wikipedia. (-). Wallpaper group. http://www.wikipedia.org.