Hierarchical S(3)-Coding of RGB Histograms

Reiner Lenz[1] and Pedro Latorre Carmona[2]

[1] Dept. Science and Engineering, Linköping University
SE-60174 Norrköping, Sweden
[2] Dept. Lenguajes y Sistemas Informáticos, Jaume I University
Campus Riu Sec s/n 12071 Castellón, Spain

Abstract. In this paper we introduce the representation theory of the symmetric group $\mathrm{S}(3)$ as a tool to investigate the structure of the space of RGB-histograms and to construct fast transforms suitable for search in huge image databases. We show that the theory reveals that typical histogram spaces are highly structured. The algorithms exploit this structure and construct a PCA like decomposition without the need to construct correlation or covariance matrices and their eigenvectors. A hierarchical transform is applied to analyze the internal structure of these histogram spaces. We apply the algorithms to two real-world databases (one from an image provider and one from a image search engine company) containing over one million images.

1 Introduction

The number and size of image collections is growing steadily and with it the need to organize, search or browse these collections. These collections can also be used to study the statistical properties of other large collections and to derive models of their internal structure.

In this paper we are interested in the understanding of the statistical structure of large image collections and in the design of algorithms for applications where huge numbers of images have to be processed very fast. We will therefore investigate the color properties of images using one of the simplest and fastest color descriptors available: the RGB histogram [18,8], which has a wide variety of applications in image processing, ranging from image indexing and retrieval [16,19,7,15] to object tracking [2] to cite a few. The approach we use is based on the observation that compression and fast-processing methods are often tightly related to the underlying structure of the input signal space. This structure can often be described in terms of transformation groups. The best-known class of algorithms of this type are the FFT-methods based on the group of shift operations. In the signal processing field these methods were generalized to the application of finite groups in filtering, pattern matching and computer vision. See [3,9,10,11,12,13] for some examples.

In the following we will first argue that a relevant transformation group for the space of RGB histograms is the group $\mathrm{S}(3)$ of permutations of three objects. We will describe the basic facts from the representation theory of $\mathrm{S}(3)$ and investigate the properties of the resulting transforms of histograms. We will see that the generated structures have a PCA like decorrelation property, leading to a separation of the histogram into blocks.

A. Ranchordas et al. (Eds.): VISIGRAPP 2009, CCIS 68, pp. 188–200, 2010.

Finally, we will analyze the internal structure of each block applying a hierarchical transform (with the help of bin quantization). These transforms will be applied to two image databases, one consisting of 760000 images representing the collection of an image provider and one with 360000 images from the database of an image search engine.

2 Notations and Basic Facts

We first summarize a few facts about permutations and representation theory and then we will describe how to generalize this to representations on spaces of RGB histograms. We only mention the basic facts and the interested reader should consult one of the books in the field, for example [14,4,6,5,1].

The permutations of three objects form the symmetric group $\mathrm{S}\,(3)$. This abstract group comes in several realizations and we will freely change between them. In the most abstract context the permutations π are just elements of $\mathrm{S}\,(3)$. We will use them to investigate color images. We describe colors in the RGB coordinate system described by triples (R, G, B). If we want to denote a triple with some numerical values then we write $(aaa), (aab), (abc)$ in the cases where all three, two or none of the values are equal. If a permutation changes the order within the triple we will simply use the new order of the generic RGB triple as a symbol for the permutation. The permutation (RBG) leaves the first element fixed and interchanges the other two. It should be clear from the context if we mean RGB-triples like (abc) or permutations like (RBG). We define the special permutations π_c as the cyclic shift $\pi_c = (BRG)$ and π_r as the reflection (RBG). These two permutations are the generators of $\mathrm{S}\,(3)$ and all others can be written as compositions of these two. The group $\mathrm{S}\,(3)$ has six elements and we usually order them as $\pi_c^0, \pi_c, \pi_c^2, \pi_r, \pi_c\pi_r, \pi_c^2\pi_r$ or in RGB notation

$$(RGB), (BRG), (GBR), (RBG), (GRB), (BGR)$$

We see that the three even permutations π_c^0, π_c, π_c^2 form a commutative subgroup with the same properties as the group of $0, 120, 240$ degrees rotations in the plane. The remaining odd permutations are obtained by preceding the even permutation with π_r.

If we consider the triples $(R, G, B)'$ as vectors $\boldsymbol{x}$ in a three-dimensional vector space then we can describe the effect of the permutations by a linear transformation described by a matrix. In this way the permutations π_c, π_r are associated with the matrices $T_G(\pi)$

$$T_G(\pi_c) = \begin{pmatrix} 0\,0\,1 \\ 1\,0\,0 \\ 0\,1\,0 \end{pmatrix}; \qquad T_G(\pi_r) = \begin{pmatrix} 1\,0\,0 \\ 0\,0\,1 \\ 0\,1\,0 \end{pmatrix} \tag{1}$$

This is the simplest example of a representation of $\mathrm{S}\,(3)$ which is a mapping from the group to matrices, so that group operations correspond to matrix multiplications. In this case the matrices are of size 3×3 and we say that we have a three-dimensional representation. The elements π_c, π_r generate $\mathrm{S}\,(3)$ and therefore we find that also all six permutation matrices are products of $T_G(\pi_c), T_G(\pi_r)$.

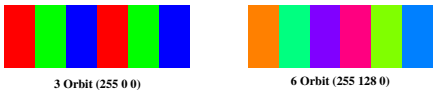

Fig. 1. Examples of a three- and a six-orbit

If we apply all six permutations to triples (abc) we obtain the so called orbits. For triples with different values for a, b and c we generate six triples, if we apply them to a triple (abb) we get three triples and the triple (aaa) is invariant under all elements in $\mathrm{S}(3)$. The orbits of $\mathrm{S}(3)$ have therefore length six, three and one respectively. We denote a general orbit by O and the orbits of length one, three and six by O_1, O_3, O_6. Two such orbits are illustrated in Fig.1 where each stripe shows one element in the orbit. For the three-orbit the colors are repeated for the odd permutations since the last two values in the RGB triple for the red image are identical. We can use the concept of an orbit to construct new representations similar to those in Eq. (1). Take the six-orbit O_6. We describe each element on O_6 by one of the six unit vectors in a six-dimensional vector space. Since permutations map elements in the orbit to other elements in the orbit we see that each permutation π defines a 6×6 permutation matrix $T_6(\pi)$ in the same way as those in Eq. (1). Also here it is sufficient to construct $T_6(\pi_c)$ and $T_6(\pi_r)$. The same construction holds for the three-orbits O_3. For the one-orbit the matrices are simply the constants $T_1(\pi) = 1$. We denote these vector spaces (defined by the orbits) by V_1, V_3, V_6.

The row- and column sums of permutation matrices are one and we see that $T(\pi)\mathbf{1} = \mathbf{1}$ where $T(\pi)$ is a permutation matrix and $\mathbf{1} = \begin{pmatrix} 1 \dots 1 \end{pmatrix}$ is a vector of suitable length with only elements equal to one. This shows that the subspaces V_k^t of $V_k, (k = 1, 3, 6)$, spanned by $\mathbf{1}$ are invariant under all permutations. These spaces define the trivial representation of $\mathrm{S}(3)$ [6,5].

Since V_k^t is an invariant subspace of $V_k, (k = 1, 3, 6)$ we see that the orthogonal complements are also invariant and we have thus decomposed the invariant spaces V_k into smaller invariant spaces and each of these subspaces defines a lower-dimensional representation (smaller matrices) of the group. The smallest such invariant spaces define the irreducible representations of the group (for definitions and examples see [14,6,5]).

The decomposition for the three-dimensional space V_3 is given by the matrix

$$P_3 = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 1 & 1 \\ \sqrt{2} & \sqrt{2}\cos(2\pi/3) & \sqrt{2}\cos(4\pi/3) \\ 0 & \sqrt{2}\sin(2\pi/3) & \sqrt{2}\sin(4\pi/3) \end{pmatrix} = \begin{pmatrix} \widetilde{\mathbf{1}} \\ \boldsymbol{P}_2 \end{pmatrix} \tag{2}$$

with the basis vector of the subspace V_3^t in the first row. The orthogonal complement is spanned by the remaining two basis vectors and it can be shown ([6]) that the space V_3^s spanned by these two cannot be split further. This defines another irreducible representation, the standard representation.

For the six-dimensional space V_6 it can be shown that the decomposition into irreducible representations is given by

$$P_6 = \begin{pmatrix} \widehat{\mathbf{1}} & \widehat{\mathbf{1}} \\ \widehat{\mathbf{1}} & -\widehat{\mathbf{1}} \\ \boldsymbol{P}_2 & 0 \\ 0 & \boldsymbol{P}_2 \end{pmatrix} \tag{3}$$

where $\widehat{\mathbf{1}}$ represents 3D vectors with entries $\frac{1}{\sqrt{6}}$ and $\boldsymbol{P}_2$ is the matrix with the two basis vectors defined in Eq.(2).

In the final stage of the construction we describe how the group operates on RGB histograms. We start with an orbit O with elements o. The permutations π are maps $\pi : O \to O$. Now take a linear function $f : O \to \mathbb{R}; o \mapsto f(o)$. We then define the new function f^π by $f^\pi(o) = f(\pi^{-1}(o))$. This defines a representation of $\mathrm{S}\,(3)$ on the space of functions on the orbit and we will also decompose them into irreducible parts.

For our application the functions of interest are the histograms. We will however modify this idea slightly. We consider a simple example first. Select an orbit O with elements o and assume that we have a probability distribution on O. Since O has finitely many elements this is a histogram h with the properties that $h(o) \geq 0$ and $\sum_{o \in O} h(o) = 1$. Applying a permutation π to the orbit elements defines a new histogram h^π. In the usual framework of representation theory we have orthonormal matrices $T(\pi)$ transforming vectors according to $h \mapsto T(\pi)h$. We thus have two transformations $h \mapsto h^\pi$ and $h \mapsto T(\pi)h$. The first of this preserves the L_1-norm while the other preserves the L_2-norm. We avoid this conflict and consider the square-roots of the probabilities instead (see also [17]). In the following we use the square root $h(o) = \sqrt{p(o)}$ where p is the probability distribution and h is the modified "histogram".

We summarize the construction so far as follows:

- Split the RGB space into subsets X such that the split is compatible with the permutations in $\mathrm{S}\,(3)$. The elements $\boldsymbol{x} \in X$ are the bins.
- For a set of images compute the probabilities $p(\boldsymbol{x}), \boldsymbol{x} \in X$
- Convert and collect them in histogram vectors h with entries $h(\boldsymbol{x}) = \sqrt{p(\boldsymbol{x})}$.
- Collect bins $\boldsymbol{x}$ related by permutations in orbits O_i, defining partitions $X = \bigcup_i O_i$
- Every orbit O defines a representation of dimension one, three or six
- Split three-dimensional representations into two parts using P_3 from Eq.(2)
- Split six-dimensional representations into four parts using P_6 from Eq.(3)
- Leave the one-dimensional representations as they are
- The final decomposition is now:

$$V = V^t \oplus V^a \oplus V^s \tag{4}$$

where V is the space defined by the bins. The space V^t is the invariant subspace associated with the one-point orbits and the invariant parts (first rows in P_3, P_6) of the three and six point orbits. V^a is the subspace associated with the six-point orbits and depends on the even/odd properties (second row in P_6) of the six point orbits. The V^s part follows the P_2 parts in the three- and six-point orbit transforms, Eqs.(2, 3).

3 Implementation

In the derivation we only required that the split of the RGB space is compatible with the operation of $\mathrm{S}(3)$ but in the rest of the paper we will always split the R, G and B intervals into eight bins each, leading to a 512D RGB histogram. We will also use octal representations of the bin-number and write (klm) for the number $k + 8l + 64m$. One-point orbits are invariant under all permutations, therefore they represent gray-values (kkk). The three-orbits are given by bin numbers (kll). Consider as example the images given by the stripes in the left part of Fig. 1. The histogram for the first stripe has a one at position (700). Applying the six permutations we get the six stripes in this figure and six histograms. Applying the transformation $\boldsymbol{P}_3$ to the three-orbit section of the histogram space given by $(700), (070), (007)$ we find that the first entry is always one and the positions in the other two dimensions (the $\boldsymbol{P}_2$ part) transform as in an equal sided triangle. These two-dimensional vectors transform thus as 120 degrees rotations under permutations. The orbit of (740), representing the RGB vector (255,128,0), are the six stripes in the right part of Figure 1. Using the decomposition defined by $\boldsymbol{P}_6$ we get two two-dimensional vectors (from the last four rows of the matrix). The coordinates of the projections into the alternating and the standard parts are collected in Table 1

Table 1. Coordinates of projections for six-point orbits

Alternating Representation

RGB	GBR	BRG	RBG	GRB	BGR
0.4082	0.4082	0.4082	-0.4082	-0.4082	-0.4082

Standard Representation

RGB	GBR	BRG	RBG	GRB	BGR
0.8165	-0.4082	-0.4082	0	0	0
0	-0.7071	0.7071	0	0	0
0	0	0	-0.4082	0.8165	-0.4082
0	0	0	0.7071	0	-0.7071

We have now described how to reorganize the histograms so that the different components show simple transformation properties under channel permutations. This is one of the advantages of this approach. The other is the relation to principal component analysis (PCA) that we will explain now.

We start with a simple example. Consider a vector h defined on an three-point orbit. Generate all different versions h^π under permutations and compute the matrix $C = \sum_\pi h^\pi h^{\pi\prime}$. It is invariant under a re-ordering of the orbit since this will simply re-arrange the sum. This is the simplest example of an $\mathrm{S}(3)$-symmetric matrix. We generalize this to the definition of a wide-sense-stationary process as follows: Assume that we have vectors h in a vector space V and the permutations $\pi \in \mathrm{S}(3)$ operate on these vectors by $h \mapsto h^\pi$. Assume further that we have a stochastic process with stochastic variable ω and values in $h_\omega \in V$. We define the correlation matrix Σ of this process as $\Sigma = \mathrm{E}\left(h_\omega {h_\omega}'\right)$ where $\mathrm{E}(.)$ denotes the expectation with respect to the stochastic variable ω. Assume further that we have a representation $T(\pi)$ on V.

Definition 1. *The stochastic process with correlation matrix Σ is T-wide-sense stationary if $T(\pi)\Sigma = \Sigma T(\pi)$ for all $\pi \in \mathrm{S}(3)$.*

We will only consider representations for which the matrices T are orthonormal and in this case we have $\Sigma = T(\pi)\Sigma T(\pi)'$ for all $\pi \in \mathrm{S}(3)$. But $T(\pi)\Sigma T(\pi)'$ is the correlation matrix of the stochastic process h in the new coordinate system $T(\pi)h$ and we see that wide-sense-stationarity means that the correlation matrix is independent of a certain class of coordinate transforms.

The general theory (Schur's Lemma, [5]) shows that we can find a matrix $\boldsymbol{U}$ (defining a new basis in the vector space) such that the correlation matrix in the new space is block diagonal. This matrix $\boldsymbol{U}$ depends only on the group $\mathrm{S}(3)$:

Theorem 3.1. *For an $\mathrm{S}(3)$-symmetric process with correlation matrix Σ we can find a matrix $\boldsymbol{U}$ such that:*

$$\boldsymbol{U}\Sigma\boldsymbol{U}' = \begin{pmatrix} \Sigma^t & 0 & 0 \\ 0 & \Sigma^a & 0 \\ 0 & 0 & \Sigma^s \end{pmatrix} \tag{5}$$

This transformation $h \mapsto \boldsymbol{U}h$ defines a partial principal component analysis of the histogram space by block-diagonalizing the correlation matrix. It is given by the construction described in Section 2.

4 S(3) Experiments

We implemented the transform described above and used it to investigate the internal structure of two large image databases. The image databases denoted by **PDB** contains 754034 (watermarked) images from an image provider. The second database, **SDB**, contains 322903 images collected from the internet by a commercial image search engine. We computed first the distributions p and then the square-roots h. In all experiments we used eight bins/channel. We get 8 one-point orbits, 56 three-point orbits and 56 six-point orbits. The dimensions of the blocks (Eq.(5) are $(120, 56, 336)$. We apply the transform resulting in the new vector $v = (v^t, v^a, v^s)$ corresponding to the vector spaces in Eq.(4).

We describe first some of our experiments regarding the statistical properties of these databases and then we illustrate the compression properties of the group theoretical transform in Section 3.

We first computed the norms of the vectors v^t, v^a and v^s, their mean and max value of $\|v^a\|^2$ for **PDB** and **SDB**. The coefficients in v^a (see also Table 1) are given by differences between contributions from even permutations and odd permutations in a

Table 2. Contributions of the coefficients

Database	$\mathrm{E}\left(\|v^t\|^2\right)$	$\mathrm{E}\left(\|v^a\|^2\right)$	$\mathrm{E}\left(\|v^s\|^2\right)$	$\mathrm{Max}(\|v^a\|^2)$	zero vs. non-zero
PDB	0.606	0.042	0.350	0.312	0.057
SDB	0.678	0.031	0.291	0.278	0.144

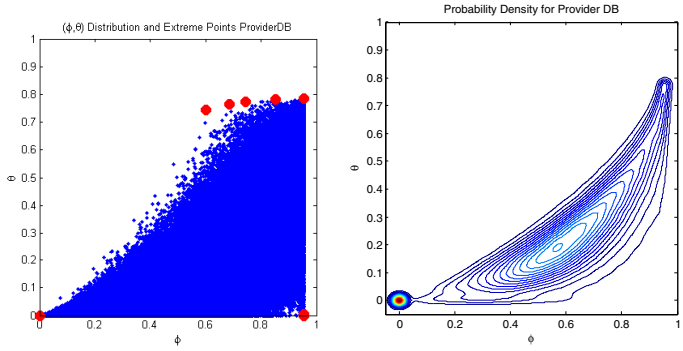

Fig. 2. Location and probability distribution from **PDB**

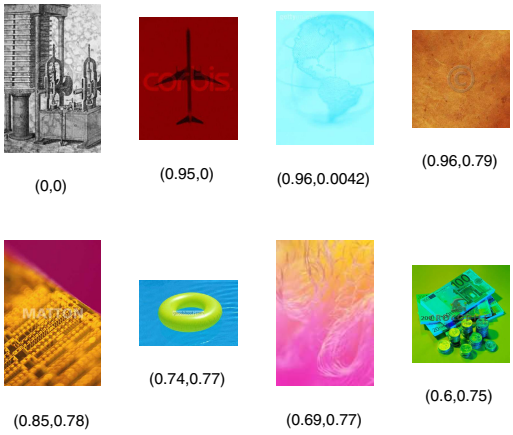

Fig. 3. Extreme images in **PDB**

six-orbit. If we assume that even and odd permutations are statistically equally likely then we expect the value of these coefficients to be small on average. We also computed the histogram (with 1000 bins) of these norms $\|v^a\|^2$ and computed the ratio between the probability of the first bin (with the small values of the norm) and the sum over the remaining bins (representing the non-zero norm values). The results are collected in Table 2.

This shows that the main contribution comes form v^t and the contributions from v^a are indeed low. We also see that there is a difference between the two databases where the contribution of the v^a is higher for **PDB**. One reason for this could be the higher proportion of cartoon-like images with distinct color distributions in **PDB** as compared with **SDB**.

From the construction we know that the modified histograms and their transforms are unit vectors $\|v\|^2 = \|v^t\|^2 + \|v^a\|^2 + \|v^s\|^2 = 1$. Since $\|v^a\|^2$ is small we conclude that the three-dimensional vectors $v = (v^t, v^a, v^s)$ are concentrated in the neighborhood of

one quarter of a great circle of the unit sphere. The length of v^t is also larger than the length of v^s. Based on these heuristic considerations we introduce the following polar coordinate system on the unit vectors given by the norms of the projection vectors:

$$\left(v^t, v^a, v^s\right) = (\cos\varphi\cos\theta, \cos\varphi\sin\theta, \sin\varphi) \tag{6}$$

The angle φ corresponds to the latitude and we think of it as an indication of the unbalance between the three channels (for a value of zero all the contribution is in the v^t part). The (longitudal) angle θ is a measure of the contribution of v^a. The (φ, θ)-distribution of the images (every dot corresponds to one image) in the **PDB** is shown in the left plot of Figure 2. The corresponding probability density distribution is shown in the right plot. This figure shows that the distribution of the images is concentrated around the origin and that the distribution has a banana-like shape in the (φ, θ)-space. The positions of the eight extreme points of the convex hull are marked with filled circles in the left plot of Figure 2. The images belonging to the eight extreme points of the convex hull are shown in Figure 3. Theorem 3.1. shows that wide-sense-stationary processes are partially de-correlated by the transform. In the remaining part of this section we will now investigate if the two databases define wide-sense-stationary processes.

We illustrate the effect of the transform on the correlation matrix in Figure 4 showing the contour plots of the correlation matrices computed from the square-root transformed histograms before and after the transformation. It can be clearly seen that the effect of the transformation is a concentration in the first 120 components given by the vectors v^t. In the following experiment we evaluated the approximation error introduced by reducing the correlation matrix computed from the transformed histograms (Figure 4) to the block-diagonal matrix with block-sizes (120,56, 336). We computed the first 20 eigenvectors of the full correlation matrices and found that they explain about 85% of the summed eigenvalues for both databases. We also computed the 20 eigenvectors for the block-structured correlation matrix. From the construction of the blocks we expect that these eigenvectors of the block-diagonal matrix are elements of the 120, 56 or 336-dimensional subspaces defined by the blocks. In Table 3 we illustrate the accumulated, normed eigenvalues $A_K = (\sum \gamma_{k=1}^{K})/(\sum \gamma_{k=1}^{512})$ where γ_k is the k-th eigenvalue and we list to which block the different eigenvectors belong. We see the minor role of the coefficients in v^a: the only eigenvectors from the second block are in positions nine and eleven. Also here we see that **PDB** has a higher contribution from v^a.

Table 3. Eigenvalues and Block-Number for the Block-Diagonal Eigenvectors

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SDB	0.40	0.51	0.57	0.62	0.65	0.67	0.70	0.72	0.74	0.75	0.76	0.77	0.78	0.79	0.80
Block	1	1	3	1	1	3	1	3	3	3	2	1	1	3	3
PDB	0.41	0.49	0.55	0.59	0.63	0.66	0.68	0.70	0.71	0.72	0.74	0.75	0.76	0.77	0.78
Block	1	3	1	1	3	1	3	3	2	1	3	3	1	3	3

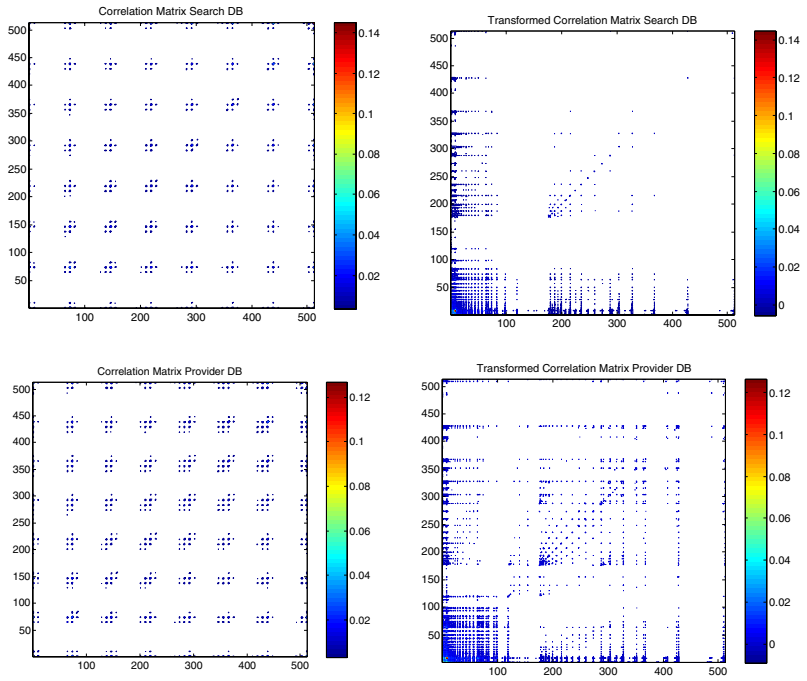

Fig. 4. Original and transformed correlation matrices

To get a more quantitative measure on how good the block-diagonal eigenvalues approximate the full matrix eigenvectors we computed the approximation values:

$$\gamma_{m,n} = \frac{\sum_{k=1}^{n} \left\| \widetilde{B}_m b_k \right\|}{n} \tag{7}$$

where $\widetilde{B}_m$ is the matrix of the m first eigenvectors of the block-diagonal approximation and b_k is the k-th eigenvector of the full correlation matrix. The value of $\gamma_{m,n}$ is a normalized measure on how large part of the n first eigenvectors of the full correlation matrix are projected into the space of the first m eigenvectors of the block-diagonal matrix. Some of the results are shown in Figure 5 where the solid line shows the values of $\gamma_{mm}, m = 1..20$ (equal number of full and block-diagonal eigenvectors) and the dashed line shows $\gamma_{20,n}$, the result for the full set of 20 eigenvectors of the block-diagonal matrix.

We see that the first 20 eigenvectors explain about 85% of the contributions from the first 20 eigenvectors of the full correlation matrix. Since the contribution from the last eigenvectors to the total variation (see also Table 3) is small we find that the approximation computed from the block-diagonal matrix is probably sufficient for most applications.

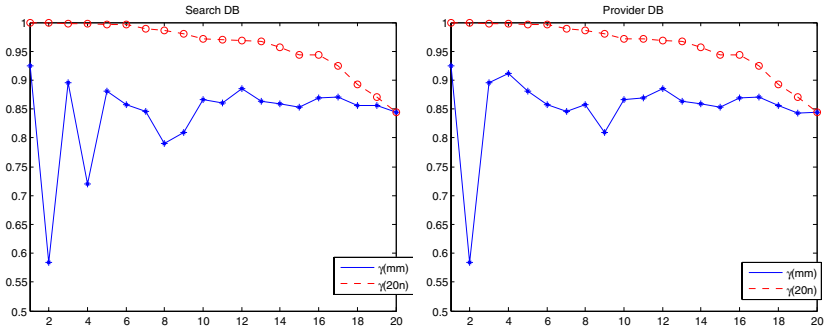

Fig. 5. Projection properties of the approximation.

5 Hierarchical Transforms and Bin Quantization

The experiments described so far show that the transform, derived from the assumption that permutations of the R, G and B channels are likely to occur with the same probabilities, lead to a separation of the histogram space into three clearly separated blocks. From an $\mathrm{S}(3)$ point of view nothing can be said about the internal structure of these blocks and if we want to transform them we have to use different sources of information about them. In the following we will introduce additional structure based on the bins defining the histogram. We consider only RGB histogram consisting of $2^n \times 2^n \times 2^n$ bins. We call such a histogram an level n histogram. In the experiments we will use histograms of size $1^3, 2^3, 4^3$ and 8^3, i.e. histograms of levels zero to three. A level-n histogram is the parent of a level-$(n+1)$ histogram.

We start with the parent node describing an orbit with label $(2k, 2l, 2m)$. There are eight possible children $\left\{(\widetilde{k},\widetilde{l},\widetilde{m}) = (2k+\Delta_k, 2l+\Delta_l, 2m+\Delta_m)\right\}, 0 \le \Delta_k, \Delta_l, \Delta_m \le 1$. Since $(\widetilde{k},\widetilde{l},\widetilde{m})$ is the label of an orbit they have to fulfil the conditions $\widetilde{k} \le \widetilde{l} \le \widetilde{m}$. For the case $k < l < m$ we see that $\widetilde{k} < \widetilde{l} < \widetilde{m}$ and all nodes with eight elements have thus eight-element children nodes. For a parent node $(2k, 2l, 2l), k < l$ the labels $\left\{(\widetilde{k},\widetilde{l},\widetilde{l}) = (2k+\Delta_k, 2l+1, 2l)\right\}$ are not allowed and they have therefore six-element orbits as children. For the case $(2k, 2k, 2k)$ we find four children $(2k, 2k, 2k)$, $(2k+1, 2k+1, 2k+1), (2k, 2k, 2k+1), (2k, 2k+1, 2k+1)$, the first two are one-point orbits, the last two are three-point orbits. We now consider the coefficients related to the trivial representations. Every orbit produces one such coefficient and we can collect the coefficients originating in the same parent orbit into vectors of size four, six and eight. In our experiments we transform these four, six and eight dimensional vectors using the matrices P_4, P_6, P_8:

$$P_4 = \begin{pmatrix} 1/2 & 1/2 & 1/2 & 1/2 \\ \sqrt{1/2} & -\sqrt{1/2} & 0 & 0 \\ 0 & 0 & \sqrt{1/2} & -\sqrt{1/2} \\ 1/2 & 1/2 & -1/2 & -1/2 \end{pmatrix}$$

$$P_6 = \begin{pmatrix} \sqrt{1/6} & \sqrt{1/6} & \sqrt{1/6} & \sqrt{1/6} & \sqrt{1/6} & \sqrt{1/6} \\ \dots & & & & & \end{pmatrix}$$

and

$$P_8 = \begin{pmatrix} \sqrt{1/8} & \sqrt{1/8} & \sqrt{1/8} & \sqrt{1/8} & \sqrt{1/8} & \sqrt{1/8} & \sqrt{1/8} & \sqrt{1/8} \\ \dots & & & & & & & \end{pmatrix}$$

For the following discussion only the first line in the three matrices is relevant. This choice ensures that we compute the average over the coefficients of the children. The remaining rows are chosen so that the transformation matrix is orthonormal. Only the average is used in the next stage. As a result we get a sequence of decompositions of the 120D space (related to the trivial representation) $120D \mapsto (20D, 100D) \mapsto (4D, 16D, 100D) \mapsto (1D, 3D, 16D, 100D)$ and a final decomposition of the 512D space into a sequence of subspaces:

$$V_1 \subset V_4 \subset V_{20} \subset V_{120} \subset V_{176} \subset V_{512}$$

We investigated the properties of the hierarchical decomposition on a smaller database consisting of 16734 images from the search engine. We call this database the "Small-Search-Engine Database" or **SSDB**. On the (square root) of the histograms we first applied the S(3) transform and then the decomposition of the 120D block. This results in a block-diagonal matrix consisting of six blocks of sizes $1, 3, 16, 100, 76, 336$ We compute the first twenty eigenvectors from this block-diagonal matrix and compute to which block they belong. We found that the first eigenvector comes from block one, the second from block four and the third from block three, etc.. The complete distribution is shown in Figure 6.

We also evaluated the influence of the transformation on the performance of a database search strategy. For this we first computed for each image the first 20 PCA coefficients. Then we choose 500 random images from the **SSDB** as query images. For each query image we computed the distance to all images in the **SSDB** using the usual Euclidean or L^2-norm based on the feature vectors computed from the PCA coefficients

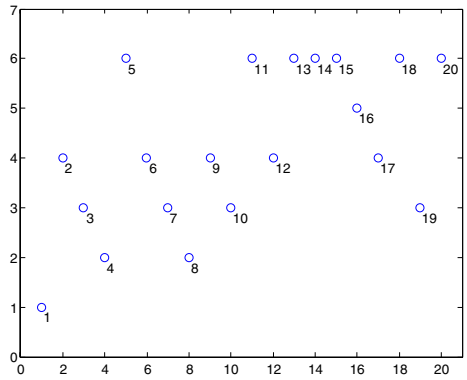

Fig. 6. Blocks and Eigenvectors

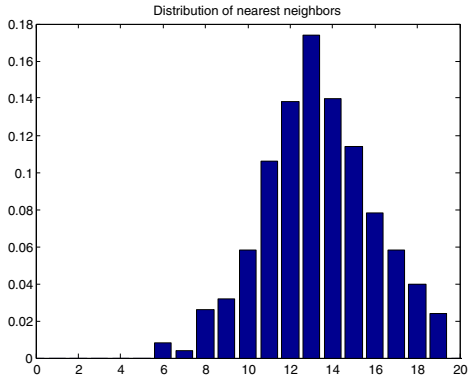

Fig. 7. Histogram over Ranks

in the original and the transformed coordinates. If k is the index of the query image and l denotes the l-th nearest neighbor of this query image computed from the feature vectors in the original space then we define $rank(k, l)$ as the retrieval rank computed from the transformed PCA coordinates. In other words: the image that comes in the l-th position in the ordered distances in the original feature space comes at position $rank(k, l)$ in the distance list computed from the transformed coordinates (Since the query image comes always first in this list we start indexing with zero). Some results obtained are shown in Figure 7 where we computed first for each query image how many of the first 20 retrieved images in the original coordinate space are located in the list of the first 20 images computed from the transformed vectors (which is the case when $rank(k, l) \leq 20$ for $l \leq 20$). We then computed the histogram over these numbers. We see that these numbers are always greater than five, which means that the approximation-based search always found the five most similar images from the full-PCA based retrieval. Most of the times we found twelve of them and never all twenty.

6 Summary

In this paper we showed that the intuitive assumption that R, G and B channels can be interchanged on average motivates the application of tools from the representation theory of the symmetric group $\mathrm{S}\,(3)$. We implemented a fast transform using tools from representation theory and used them to investigate the structure of two large image databases and to develop fast PCA-like compression methods. We further combined the permutation-based transformation with a hierarchical decomposition based on the quantization level of the RGB histogram. The properties of the resulting transformations were illustrated with the help of two large image databases. The quantitative performance of the different approximations shows that the compressed feature vectors are probably sufficient for many retrieval tasks.

Acknowledgements. The support of the Swedish Research Council, the Knowledge Foundation, Sweden and the Ministry of Education and Science of the Spanish

Government through the *DATASAT* project ($ESP - 2005 - 00724 - C05 - C05$) and the *MIPRCV* project ($CSD2007 - 00018$) are gratefully acknowledged. Pedro Latorre Carmona is a *Juan de la Cierva* Programme researcher (Ministry of Education and Science). The databases were provided by Picsearch AB, Stockholm and Matton AB, Stockholm.

References

1. Chirikjian, G.S., Kyatkin, A.B.: Engineering applications of noncommutative harmonic analysis: with emphasis on rotation and motion groups. CRC Press, Boca Raton (2000)
2. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Trans. Pattern Analysis and Machine Intelligence 25, 564–577 (2003)
3. Cooley, J.W., Tukey, J.W.: An algorithm for machine calculation of complex Fourier series. Mathematical Computations 19, 297–301 (1965)
4. Diaconis, P.: Group representation in probability and statistics. Institute of Mathematical Statistics, Hayward, Calif. (1988)
5. Fässler, A., Stiefel, E.: Group theoretical methods and their applications. Birkhäuser, Basel (1992)
6. Fulton, W., Harris, J.: Representation Theory. Springer, Heidelberg (1991)
7. Geusebroek, J.M.: Compact object descriptors from local colour invariant histograms. In: Proc. British Machine Vision Conference, BMVC (2006)
8. Hafner, J., Sawhney, H.S., Equitz, W., Flickner, M., Niblack, W.: Efficient color histogram indexing for quadratic form distance functions. IEEE Trans. Pattern Analysis and Machine Intelligence 17(7), 729–736 (1995)
9. Holmes, R.B.: Mathematical foundations of signal processing. SIAM Review 21(3), 361–388 (1979)
10. Lenz, R.: Group Theoretical Transforms in Image Processing. Springer, Heidelberg (1994)
11. Lenz, R.: Investigation of receptive fields using representations of dihedral groups. J. Visual Communication and Image Representation 6(3), 209–227 (1995)
12. Lenz, R.: Crystal vision-applications of point groups in computer vision. In: Yagi, Y., Kang, S.B., Kweon, I.S., Zha, H. (eds.) ACCV 2007, Part II. LNCS, vol. 4844, pp. 744–753. Springer, Heidelberg (2007)
13. Rockmore, D.: Recent progress and applications in group FFT's. In: Computational noncommutative algebra and applications. Kluwer, Dordrecht (2004)
14. Serre, J.P.: Linear representations of finite groups. Springer, Heidelberg (1977)
15. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content based image retrieval at the end of the early years. IEEE Trans. Pattern Analysis and Machine Intelligence 22, 1349–1380 (2000)
16. Sridhar, V., Nascimento, M.A., Li, X.: Region-based image retrieval using multiple-features. In: Chang, S.-K., Chen, Z., Lee, S.-Y. (eds.) VISUAL 2002. LNCS, vol. 2314, pp. 61–75. Springer, Heidelberg (2002)
17. Srivastava, A., Jermyn, I., Joshi, S.: Riemannian analysis of probability density functions with applications in vision. In: IEEE Conf. Comp. Vision and Pattern Recognition, Minneapolis, MN, pp. 1–8 (2007)
18. Swain, M.J., Ballard, D.H.: Color indexing. Int. J. Comp. Vision 7(1), 11–32 (1991)
19. Yoo, H.W., Jang, D.S., Jung, S.H., Park, J.H.: Visual information retrieval system via content-based approach. Pattern Recognition 35, 749–769 (2002)