Octahedral Filters for 3D Image Processing

Reiner Lenz
Linköping University, Norrköping, SE-601 74, Sweden, Email: reile@itn.liu.se.

***Abstract*—The octahedral group is one of the finite subgroups of the rotation group in three-dimensional Euclidean space and a symmetry group of the cubic grid. We demonstrate filtering of three-dimensional volumes as application examples of its representation theory. We summarize properties of the octahedral group and basic results from its representation theory. Linear filter systems are defined as projection operators and symmetry-based filter systems are generalizations of the Fourier transforms. The algorithms are implemented in Maple/Matlab worksheets and functions. We illustrate the nature of the different types of filter systems, their invariance and transformation properties.**

I. Introduction

All digital image processing methods ultimately operate on discrete structures but many processing methods are based on continuous theories. This requires that the results obtained have to be adapted to the discrete case. Very often this step is essentially ignored using some ad-hoc implementations. In this paper we use an approach based on the application of finite groups. We assume that the measurements are given on a geometrical structure with a group-theoretically defined regularity. Based on this regularity, and only on this regularity, we will then develop our image processing tools and show how they perform for real data. This is the approach used in [1], [2] to construct two-dimensional filter methods. Here we derive the corresponding tools for the three-dimensional case where the discrete data is defined on a cubic grid. We implemented the mathematical framework in a combination of Maple and Matlab programs. Maple is used for generating the abstract description of the group involved. This description is then translated to a matrix-vector based framework which is the basis of a Matlab toolbox. We use typical MRI-volumes to illustrate some applications of the octahedral filter systems for three-dimensional image processing. The theory and, to a large extend, also the code can be directly adapted to other similar cases, for example to problems involving the icosahedral group.

II. The Octahedral Group

In [1], [2] we used the representation theory of the dihedral groups to develop fast filter systems for image processing on square and hexagonal grids. In the following we will generalize this to 3D image processing. These corresponding groups are the finite subgroups of the 3D rotation group $\mathrm{SO}(3)$. We first describe the detailed structure of one of these groups, the Octahedral group O and we will explain its relation to the cubic sampling pattern in 3D space.

We describe the 3D rotations by orthonormal matrices $\boldsymbol{R}$. They operate on the sphere and we define the transformation group as the pair $(\mathrm{SO}(3), \mathcal{S})$ with group action $(\boldsymbol{R}, \boldsymbol{x}) \mapsto \boldsymbol{R}\boldsymbol{x}$. If G is a subgroup of $\mathrm{SO}(3)$ and $\boldsymbol{x} \in \mathcal{S}$ is a unit vector then we define the orbit of $\boldsymbol{x}$ under G as: $G\boldsymbol{x} = \{\boldsymbol{R}\boldsymbol{x} : \boldsymbol{R} \in G\}$. It can be shown, see [3], that the (truly 3D) finite subgroups of SO (3) have $12, 24$ or 60 elements. Here we consider the Octahedral group O, for the icosahedral group I we refer to [4].

The properties that we need in the following are collected in the following theorem:

Theorem 1 O *has 24 elements and can be described by two generators* $\boldsymbol{R}_1, \boldsymbol{R}_2$ *and four equations.*

Elements in O *map points on the cubic grid to other points on the cubic grid.*

It has five different types of orbits: (i) The single point orbit $\mathcal{O}_o$ *consisting of the origin. (ii) The axis orbit* $\mathcal{O}_a$ *of the six points on the coordinate axes. They are the centers of the faces of the cube. (iii) The corner orbit* $\mathcal{O}_c$ *consisting of the eight corner points of a centered cube. (iv) The vertex orbit* $\mathcal{O}_v$ *with the twelve points on the middle of the vertices. Finally (v) The general orbit* $\mathcal{O}_g$ *with 24 points.*

The generators are the rotation matrices $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$:

$$\boldsymbol{R}_1 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \qquad \boldsymbol{R}_2 = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (1)$$

Consider a finite collection W of N points in space that is invariant under all operations of the octahedral group O. The scalar valued functions defined on W form an $N-$dimensional vector space. A function f defined on W can be described by an $N-$dimensional vector that we will also denote by f. These functions form the vector space $\mathbb{R}^N$. The rotations $\boldsymbol{R}$ in the group O operate on W and therefore also on the functions on W. We illustrate this by constructing the matrix acting on a six-point axis orbit $\mathcal{O}_a$. We denote the six points on the x-, y- and z-axis as $\boldsymbol{x}, -\boldsymbol{x}, \boldsymbol{y}, -\boldsymbol{y}, \boldsymbol{z}$ and $-\boldsymbol{z}$ and order them as $\boldsymbol{x}, -\boldsymbol{y}, -\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ and $-\boldsymbol{z}$. The ordering of these points is arbitrary and we only choose it to get a simple matrix description later. In this order we map the six points to the six canonical basis vectors $\boldsymbol{e_1}, \ldots, \boldsymbol{e_6}$ in a six-dimensional vector space. Thus: $\boldsymbol{x} \mapsto \boldsymbol{e_1}, -\boldsymbol{y} \mapsto \boldsymbol{e_2}, \ldots, \boldsymbol{z} \mapsto \boldsymbol{e_6}$. A simple calculation shows now that if we apply the rotation $\boldsymbol{R}_2$ to the coordinate vectors then the matrix $\boldsymbol{T}(\boldsymbol{R}_2)$ describing the transformation of the values on the orbit is (in this ordering) given by

$$\boldsymbol{T}(\boldsymbol{R}_2) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

In this way we see that for every function f, defined on an invariant set with $N-$ points, and every rotation $\boldsymbol{R}$ we get a transformed function $f^{\boldsymbol{R}} : f^{\boldsymbol{R}}(\boldsymbol{x}) = f(\boldsymbol{R}^{-1}\boldsymbol{x})$. Every rotation $\boldsymbol{R}$ defines therefore an $N \times N$ matrix $\boldsymbol{T}(\boldsymbol{R})$ such that $f^{\boldsymbol{R}} = \boldsymbol{T}(\boldsymbol{R})f$. We formalize this in the definition of a representation of a group:

Definition 1 *A matrix representation of a group is a mapping $\boldsymbol{T} : g \mapsto \boldsymbol{T}(g)$ with*

$$\boldsymbol{T}(gh) = \boldsymbol{T}(g)\boldsymbol{T}(h) \forall g, h \in G$$

where the $\boldsymbol{T}(g)$ are invertible matrices. If the matrices $\boldsymbol{T}(g)$ are of size $N \times N$ then we say that the degree of the representation is N.

A representation is called irreducible if there are no non-trivial invariant common subspaces for the transformations described by the matrices $\boldsymbol{T}(g)$.

It can be shown that there are five irreducible representations of O given by $\boldsymbol{T}_k, k = 1, \ldots, 5$ with dimensions $1, 1, 2, 3, 3$.

For a given orbit we can now construct the representation matrices $\boldsymbol{T}(\boldsymbol{R}_1), \boldsymbol{T}(\boldsymbol{R}_2)$ and since all rotations $\boldsymbol{R}$ are products of $\boldsymbol{R}_1, \boldsymbol{R}_2$ and their inverses and since $\boldsymbol{T}(\boldsymbol{R}_1)\boldsymbol{T}(\boldsymbol{R}_2) = \boldsymbol{T}(\boldsymbol{R_1}\boldsymbol{R_2})$ we see that we can generate all matrices $\boldsymbol{T}(\boldsymbol{R})$ as products of $\boldsymbol{T}(\boldsymbol{R}_1), \boldsymbol{T}(\boldsymbol{R}_2)$ or their inverses. For arbitrary sets W of points we partition W first into disjunct subsets consisting of orbits and then we construct the representations of each of the orbits separately. The representation on the whole set is the direct product of these orbit representations. In practice we have to construct 24 matrices, one for each element in O and for sets W with many grid points these matrices can be quite large (an $8 \times 8 \times 8$ cube requires matrices $\boldsymbol{T}(\boldsymbol{R})$ of size 512^2). We therefore compute them automatically in a series of Maple and Matlab programs.

III. Filtering

In this section we describe how the Octahedral group can be used in low-level filtering. One of the basic strategies in filtering is to use systematic variations of a given filter. Here we use the properties of the Octahedral group to design filter systems with special transform properties. As before we use a set W of N points in space that is invariant under all rotations of O. The functions defined on W are described by vectors in the N-dimensional vector space V. On V we have the representation $\boldsymbol{T}$ transforming functions f under the rotation $\boldsymbol{R}$ as $f \mapsto \boldsymbol{T}(\boldsymbol{R})f$. For filter design we consider a fixed function f and its orbit $\left\{f^{\boldsymbol{R}}, \boldsymbol{R} \in \mathrm{O}\right\}$. This defines a filter system with 24 filter functions. Depending on the nature of f these are, however, not necessarily all different. All functions $\left\{f^{\boldsymbol{R}}\right\}$ lie in V and they span an invariant subspace V_f of V. From the general theory we know that $\boldsymbol{T}$ defines a partition of V into smallest O-invariant subspaces of dimensions $1, 1, 2, 3, 3$ given by the five irreducible representations of O. We denote the transformation matrix from the ordinary, voxel-based, coordinate system to the system given by the irreducible representations by $\boldsymbol{B}$. This is a matrix of size $N \times N$ and we can assume that it is orthonormal. Furthermore, we order the matrix $\boldsymbol{B}$ such that the transformed vector $\boldsymbol{B}f$ has the form $\boldsymbol{B}f = \left(f^1, f^2, f^3, f^4, f^5\right)$ where f^k is the projection of the vector f to that part of the vector space that transforms like the $k-$th irreducible representation of O. To simplify notations we denote the two matrices that represent the group generators in the representation $\boldsymbol{T}$ by $\boldsymbol{A}$ and $\boldsymbol{D} : \boldsymbol{A} = \boldsymbol{T}(\boldsymbol{R}_1), \boldsymbol{D} = \boldsymbol{T}(\boldsymbol{R}_2)$ (see Equation (1)).

With these notations we now describe the implementation of a filter system $\left\{f^{\boldsymbol{R}}\right\}$ based on the transformation properties of the octahedral group. We start with a given filter function f that defines the orbit $\left\{f^{\boldsymbol{R}}, \boldsymbol{R} \in \mathrm{O}\right\}$. A typical example could be an edge filter in the x-direction. One method to construct the symmetry-adapted filter system is the following:

1) For f use the transformation $\boldsymbol{B}f$ and compute $\boldsymbol{B}f = \left(f^1, f^2, f^3, f^4, f^5\right)$
2) Select that part $\hat{f}$ in $\boldsymbol{B}f = \left(f^1, f^2, f^3, f^4, f^5\right)$ that has the highest norm of these five components
3) Compute the projection on the subspace with the highest projection norm by keeping component $\hat{f}$ and replacing the rest by zero-vectors. This gives the projected vector f^p
4) Reconstruct the corresponding vector in the original space as $f_p = \boldsymbol{B}' f^p$ (Note that an orthonormal $\boldsymbol{B}$ is used)
5) For the cases $\hat{f} = f^1$ and $\hat{f} = f^2$ use only $f_p = \boldsymbol{B}' f^p$ since the application of transformed versions of f_p will at most change sign of the filter result
6) For the case $\hat{f} = f^3$ the representation space is two-dimensional and therefore $f_p = \boldsymbol{B}' f^p$ and $\boldsymbol{D} f_p$ are used
7) For the case of the fourth irreducible representation $\hat{f} = f^4$ use the three filters $f_p = \boldsymbol{B}' f^p, \boldsymbol{D}\boldsymbol{A} f_p$ and $\boldsymbol{D} f_p$
8) For the remaining case $\hat{f} = f^5$ use the three filters $f_p = \boldsymbol{B}' f^p, \boldsymbol{D} f_p$ and $\boldsymbol{A} f_p$
9) The result of the construction in the last four steps is now a one-, two- or three-dimensional filter matrix $\boldsymbol{F}_p$
10) Finally update the filter vector f by projecting it to the orthogonal complement of $\boldsymbol{F}_p$
11) If the resulting projected vector is not zero then repeat the construction and compute the next filter matrix $\boldsymbol{F}_p$

Using this projection technique we can always assume that the computed filter vectors transform like the corresponding irreducible representation. We denote such a projection-based filter system by $\boldsymbol{F}^k$ where $0 \leq k \leq 5$ denotes the index of the irreducible representation. We now describe how to use the computed filter vectors in further processing based on their transformation properties.

The filter systems of type $\boldsymbol{F}^1, \boldsymbol{F}^2$ consist of one filter function only. The results of filtering with the first system $\boldsymbol{F}^1$ are independent of orientation changes of the underlying signal. We therefore use the filter results as they are. Filter results obtained from $\boldsymbol{F}^2$ are also scalar-valued. The filter results for the second filter type are invariant under one subgroup of O and change sign for the other coset of this subgroup. Therefore we describe the filter results by their absolute values and their signs. The sign indicates which coset of rotations was underlying the actual pattern. We describe the filter system derived from the fourth irreducible representation as an example of the higher dimensional filter systems. The filters are constructed by $f_p = \boldsymbol{B}' f^p$ and the two rotations $\boldsymbol{DA} f_p$ and $\boldsymbol{D} f_p$. The three-dimensional rotation matrices for the two non-trivial rotations are

$$\boldsymbol{T}_D = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \text{ and } \boldsymbol{T}_{DA} = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad (2)$$

The three-dimensional filter result vectors are first converted to polar coordinates given by the length of the vector and the unit vector for the direction. Then the following operations normalize the 3D-unit direction vector

1) First apply sequences of $\boldsymbol{T}_D$ such that the third component has the highest absolute value. There are three different cases depending on the origin of this component
2) Then there are two different cases, either the component of the highest absolute value was positive or negative. In the case of a negative value of the third component we apply $\boldsymbol{T}_{DA}$ after which we have a positive value in the third component. This gives six different labels and leaves four different cases for the first two components: $(x, y), (-y, -x), (-x, -y), (y, x)$
3) If the absolute value for the first component is larger than the absolute value for the second we exchange the two components. Combined with the previous six classes this results in twelve classes
4) Finally we code the sign of the second component applying the sign change $(x, y) \mapsto (-x, -y)$ if necessary. This gives 24 different labels

The previous labeling process defines a tiling of the unit sphere into 24 different regions. In the computation of the label we change the values in the filter vector correspondingly and we use the last version (after all normalization operations are applied) as a description of the location of the filter vector in a standard region on the sphere. We call this vector the residue vector.

The group theoretical transforms were implemented in a package of Maple worksheets and Matlab functions. These functions generate an abstract description of the group from the Maple definition. This description is then translated into a Matlab struct. Other parts of the Matlab toolbox implement the generation of arbitrary representations from two generators, the projection formulas and the generation of invariant subsets from a given sequence of points in 3D.

IV. Some Illustrations

We now illustrate some results obtained by the filter systems. We use a magnetic resonance angiography (MRA) volume from http://www.physionet.org/physiobank/database/images. It consists of coronal slices acquired from consecutive anteroposterior positions within the torso of a human body. It's size is $512 \times 512 \times 76$ and the slice thickness is $1.2mm$.

Usually one would use simple filter functions like edge patterns for filtering but here we use a unusual filter definition simply to show the flexibility of the filter design strategy. The filter is given by the product $f(\xi, \eta, \zeta) = P_3(\xi) P_3(\zeta)$ of a third order polynomial P_3. The window size was seven and all points in the cube were used, resulting in three filters with 343 coefficients each. The filter kernels are shown in Figure 1. The system consists of three filter functions and its transformation properties are those of the fourth irreducible representation.

In Figure 2 we applied this filter system to the 3D MRA volume. We show the result of applying the three filter kernels to a section in the middle of the volume. These raw filter results are then transformed using the procedure described above. The three new results, magnitude, label and residue are shown in Figure 3.

V. Summary and Conclusions

We started with the well-known fact that the group of three-dimensional rotations has only a few truly three-dimensional finite subgroups. The one that is related to a cubic sampling of space is the octahedral group O. We derived some of its most important properties and sketched a few basic facts from its representation theory. The result is a type of Fourier Analysis for signals defined on domains that are invariant under these octahedral transformations. We implemented the most important algorithms using a combination of Maple worksheets and Matlab functions and m-files. Using these tools it is easy to construct bases for vector spaces of signals defined on these domains that implement a type of Fourier-transform. We illustrated the usage of the tools by constructing filter systems for three-dimensional volumes.

The same tools can also be used to construct transform coding algorithms for three-dimensional volumes. We investigated the statistical properties of MRI-volumes and showed that for these volumes the predictions made by the group theoretical assumptions were essentially valid: we found that the transformed correlation matrices were block-diagonal, that the highest eigenvalue was related to a filter function of the averaging type and that the next few eigenvectors came in packages of one, two and three filter functions depending on their origin in the different blocks of the transformed correlation matrices. These results will be reported elsewhere.

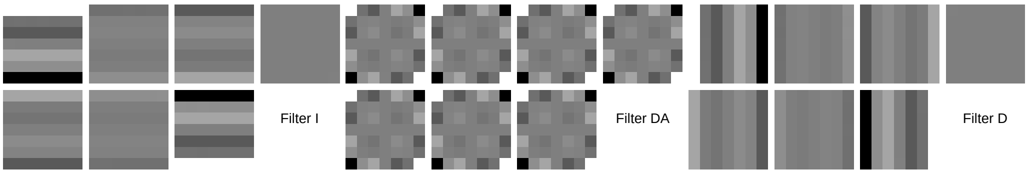
References

[1] R. Lenz, "Investigation of receptive fields using representations of dihedral groups," *J. Vis. Comm. Im. Rep.*, vol. 6, no. 3, pp. 209–227, 1995.

[2] R. Lenz, T. H. Bui, and K. Takase, "A group theoretical toolbox for color image operators," in *Proc. ICIP 05*. pp. III–557–III–560.

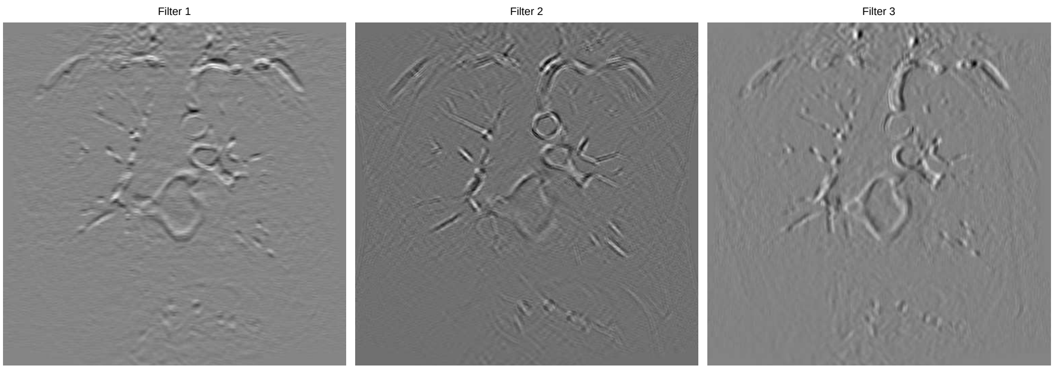
[3] S. Sternberg, *Group Theory and Physics*. Cambridge Univ. Press, 1994.

[4] R. Lenz, "Crystal vision-applications of point groups in computer vision," in *Proc. ACCV 2007*, LNCS, 4844. Springer, 2007, pp. 744–753.



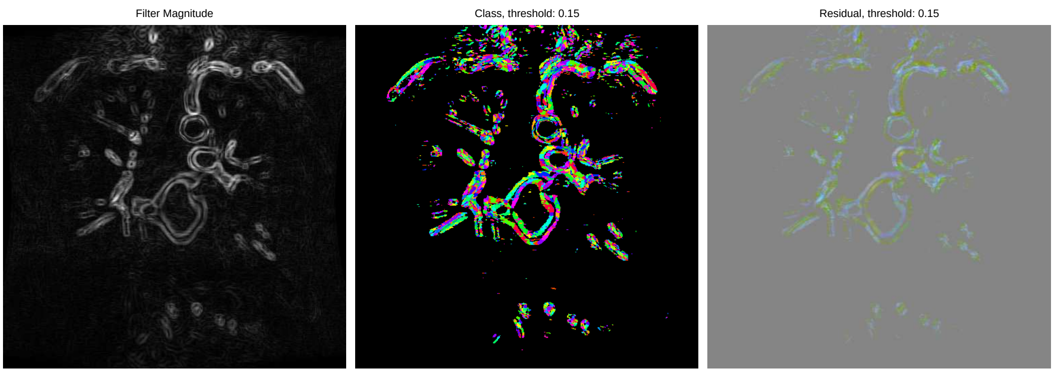
(a) First filter (b) Second filter (c) Third filter

Fig. 1: Filters related to fourth Irreducible Representation, All Points, Window Size 7



(a) First filter result (b) Second filter result (c) Third filter result

Fig. 2: The three filter results for the type 4 filters



(a) Magnitude of filter results (b) Class labels of filter results (c) Residue of filter results

Fig. 3: The magnitude, the class and the residue