Available online at www.sciencedirect.com
ScienceDirect
Computer-Aided Design 39 (2007) 460–476
COMPUTER-AIDED DESIGN
www.elsevier.com/locate/cad

Laplace spectra as fingerprints for image recognition

Niklas Peinecke, Franz-Erich Wolter*, Martin Reuter

University of Hannover, Welfenlab, Division of Computer Graphics, D-30167 Hannover, Germany

Received 12 May 2006; accepted 14 January 2007

Abstract

In the area of image retrieval from data bases and for copyright protection of large image collections there is a growing demand for unique but easily computable fingerprints for images. These fingerprints can be used to quickly identify every image within a larger set of possibly similar images. This paper introduces a novel method to automatically obtain such fingerprints from an image. It is based on a reinterpretation of an image as a Riemannian manifold. This representation is feasible for gray value images and color images. We discuss the use of the spectrum of eigenvalues of different variants of the Laplace operator as a fingerprint and show the usability of this approach in several use cases. Contrary to existing works in this area we do not only use the discrete Laplacian, but also with a particular emphasis the underlying continuous operator. This allows better results in comparing the resulting spectra and deeper insights in the problems arising. We show how the well known discrete Laplacian is related to the continuous Laplace–Beltrami operator. Furthermore, we introduce the new concept of solid height functions to overcome some potential limitations of the method.
© 2007 Published by Elsevier Ltd

Keywords: Laplace spectra; Image recognition; Laplace–Beltrami operator; Fingerprints; Watermarks; Copyright protection; Spectra; Laplace–Kirchhoff operator; Riemannian manifolds; Color images; Isospectrality; Features; Invariants; Image data bases

1. Introduction

One of the main tasks of computer science is to manage large collections of data. In general these collections are maintained using data base management systems. A variety of such systems exist, as well as different strategies for data base management systems to keep track of inserted data. All these strategies require chunks of data to be identified by a unique key, in order to distinguish them from other items stored in the data base. It is common practice to refer to such identifiers as *fingerprints*, in analogy to the way a human individual is identified by the prints of its finger tips.

Furthermore, with the area of *copyright protection* a new field of applications has developed recently. In order to identify unlicensed copies of protected material there have been efforts to develop *watermarks* intrinsic to the material in question. In the case of shape models such watermarks should be embedded in the geometry of the shape itself and they should be robust against distortions, caused e.g. by reconstructing shapes through scan processes. See [29,30] and [37,38] for recent developments in this area. For images this means that an image that is printed out and then scanned again is still identifiable by its fingerprint. Such fingerprints can be used as watermarks, too.

For some types of data constructing a suitable fingerprint is rather straightforward: for example, if the data is a collection of English words, it is sufficient to identify each word by its representation as an ASCII- or unicode-string. Well known techniques like *tries* (cf. [28]) or the like can then be applied easily. With the data consisting of images there is no such straightforward representation. A number of obstacles arise when trying to construct unique fingerprints for images, e.g.:

- Identical images can be represented in different ways. Even if we require all the images to be given as pixels, color images can be expressed in different color spaces. Thus we would need to restrict ourselves to a fixed color space, i.e. RGB space. For images given in other spaces a conversion needs to be applied.
- Unlike ASCII-strings, images may contain minor distortions, e.g. resulting from numerical errors during a color

* Corresponding address: Universitaet Hannover, Lehrstuhl Graphische Datenverarbeitung, Inst, Inf., Welfengarten 1, 30167 Hannover, Germany.
E-mail addresses: peinecke@gdv.uni-hannover.de (N. Peinecke), few@gdv.uni-hannover.de (F.-E. Wolter), reuter@gdv.uni-hannover.de (M. Reuter).

0010-4485/$ - see front matter © 2007 Published by Elsevier Ltd
doi:10.1016/j.cad.2007.01.014

space conversion or compression artifacts. We would like our fingerprints to be robust in the presence of such distortions.
- Pixel images can be given in different resolutions. For some applications it would be desirable that the same image in a different resolution would be identified by the same (or a very similar) fingerprint.
- Sometimes the content of an image is independent of the colors chosen, i.e. color rotation or inversion does not change the meaning of an image. For example, the negative of a photograph still represents the same content (although it may look strange). Thus we would need identical fingerprints in those situations.

It is an intricate task to construct fingerprints suitable for image collections. In general it is impossible to take the image itself in its RGB-representation as its fingerprint. Even for a rather tiny image with 200×200 pixels the resulting fingerprint would be a vector with 40 000 entries. There is no efficient way to search a 40 000-dimensional space of objects. There has been some effort to overcome these dimensional restrictions recently; however, generally speaking the fewer dimensions that are involved, the more efficient the search that can be carried out. See [20] for a survey of search methods on higher dimensional spaces, and see [4] or [3] for examples of efficient data structures for such applications. Furthermore, there is the possibility to reduce the dimensionality of a feature space afterwards using multi-dimensional scaling (MDS); see [2] for an overview on different methods.

In other words we need to find a map from the space of images to a much lower dimensional space without losing information that is relevant for the content. It should be noted, that there are different approaches to automatically construct such a map from a given training set of images, e.g. by MDS. Apart from that, we are looking for fingerprints that are suitable for different setups without having prior training sets.

There have been different approaches to construct fingerprints, the best known of which are *feature vectors*. These vectors are constructed from a number of features that can be extracted from an image, including:

- brightness (i.e. mean pixel value) of an image
- contrast (i.e. variance of the pixel values) of an image (see e.g. [32] for an overview of these features and more)
- overall roundness of contained shapes
- approximate fractal dimension (see [36])
- Fourier transform and wavelet transform (see [41])
- skeletal transforms (e.g. discrete medial axis transforms, see e.g. [10]).

Some of these features obviously do not fulfill our requirements, e.g. changing the brightness or contrast of an image does not change important content of the image. Most of the classical techniques are only suitable for certain special classes of images and not for others; i.e. they cannot be applied in a general situation where there is no further knowledge about the nature of the images available. For example, computing the medial axis of a shape is impossible if the image in question does not contain any obvious shape at all, e.g. for gradient images. Of course there is a large number of more elaborate techniques but a detailed discussion would be beyond the scope of this short introduction. See e.g. [46,31,43] or [34] for an overview of different methods.

In this paper we will introduce a method that works in the general case where there is no restriction concerning the kind of images. We will develop criteria to be met in such a setup, and show that our fingerprints fulfill these criteria. We will explain how to obtain these fingerprints using the spectra of a family of operators known as Laplace or Kirchhoff operators. Only discrete versions of these operators have been used in image processing traditionally, while others (like the continuous Laplace–Beltrami operator) are relatively new in this area. In order to compute the operators' eigenvalues, we will interpret images as Riemannian manifolds. This is a novel approach in the area of image analysis. Based on the theory developed in [35] we will explain the techniques from the point of view of its application on images, although they can be used for more general shapes also. See [47] and [37,38] for an introduction to the theory of Laplace spectra in general shape matching.

2. Related work

Using the Laplace operator and more generally using eigenvalues of different operators and matrices derived from this operator is a well known and established technique in the community of shape and image recognition. Most of the applications mentioned in this section use discrete forms of the Laplacian directly, i.e. they are using some kind of admittance matrix.

One of the best known applications of the Laplace spectrum of a graph is graph partitioning. This is useful in areas where one has to find a segmentation of a given mesh, e.g. to identify different components of a scene. See [33,9] for an overview of the mathematical foundations. Closely related is the application of eigenfunctions to remesh given objects (see [15,16]).

Another popular application of the Laplacian can be found in image processing. A local version of the mesh Laplacian is well established for smoothing of images and meshes; see [19,21,23,7,18,45] for examples. A variant of the Laplace operator can be used for mesh partitioning and compression. This technique is known as spectral compression; details can be found in [26].

Furthermore, the Laplace operator is used for dimensionality reduction of high dimensional data spaces (cf. [1]). This method could be seen as a complement of the method developed in this paper. Belkin and Niyogi assume the existence of a manifold containing all the objects of a given set of objects represented as points in a feature space. One can assume that these points form a low dimensional manifold since for most applications the space of possible data depends continuously on few parameters. For example, given a set of images shot from an indoor scene by a movable camera the results depend solely on six parameters, i.e. placement and orientation of the camera. Belkin and Niyogi then use some interpolation technique to form a discrete mesh resembling the assumed manifold containing the objects representations in feature space

and compute eigenvalues and eigenfunctions of the associated mesh Laplacian. These eigenfunctions can be used to reduce the dimensionality of the embedding feature space to a lower dimension, e.g. six dimensions for the camera example.

The principal difference (aside from using the mesh Laplacian rather than the Laplace–Beltrami operator) is that Belkin and Niyogi compute the eigenvalues of the manifold defined by points in a given feature space, that is each of their points represents an entire manifold in our approach. We use the manifold's eigenvalues as features. The Belkin/Niyogi method is useful when there is a set of features present that has to be reduced for some reasons, while our method is useful when there is a manifold representation for each object possible but no efficient features are known. The Belkin/Niyogi technique can roughly be classified as a multi-dimensional scaling method (MDS) although it is implemented differently from the classical approaches (cf. [6] and [2]).

Another use of MDS is to map shapes to a canonical signature surface (cf. [17]). Ideally two isometric shapes should be mapped to an (almost) identical signature surface. It would be possible to use the Belkin/Niyogi technique for this application also.

These uses of the Laplacian show that the application of (non-Riemannian!) manifolds in feature space are well established in the image analysis community. On the other hand the duality of the image as a discrete height function and its continuous representation as a surface is also well known in the theory of image processing (see [22]). Nevertheless there seems to be no transfer between the two fields, i.e. the discussion of an image as a Riemannian manifold. We will see in Section 5 how to acquire a Riemannian metric for an image and use it for image classification. This discussion seems to be new.

Using the eigenvalues of a special matrix derived from point sets of two given objects in Euclidean space, one can compute a best match of these points. This technique, called *modal matching*, was developed by Sclaroff and Pentland [42] based on the classical well known technique of momentum matrices. No features are derived from the shapes in this approach; instead, the chosen points are matched to each other directly.

3. Features and invariants

In this section we will develop criteria for a fingerprint to be used in image identification. We will make precise what we mean by a feature and an invariant.

We call a map $\iota : A \to B$ a *feature map* if it makes some kind of higher level information from the objects in the set A available. Features may include any kind of meaningful information, e.g. placement of certain objects within a scene, overall contrast or brightness of an image, and so on. For purposes of efficiency one would expect a feature to take up less space than the object itself. Nevertheless some features may require more space for representation than the original dataset.

Definition 1. Let A, B, I be sets, $P = (P_i : i \in I)$ be a partition of A with representatives $p_i \in P_i$, and $\iota : A \to B$ be a map (and in our case even a feature map) with

$$\iota(a) = \iota(p_i) \Leftrightarrow a \in P_i \quad \forall i \in I.$$

Then ι is called a *P-invariant* for A.

If $|P_i| = 1$ for all $i \in I$ we call ι a *characteristic*. There is a relation between partitions and maps. Suppose we have a family of maps $F = (f_j)$ with $f_j : A \to A$, $j \in J$ for some index set J with $\cup_{i\in I} \cup_{j\in J} f_j^{-1}(p_i) = A$ and $f_j^{-1}(p_i) \cap f_j^{-1}(p_l) = \emptyset$ for all $j \in J$ and $i \neq l$. Then there exists a natural partition of A with

$$a \in P_i \Leftrightarrow \exists j \in J : f_j(a) = p_i.$$

It is therefore perfectly admissible to speak of an F-invariant, e.g. for F being the family of isometries of an isometry invariant.

We can now give some criteria for fingerprints feasible for image identification, based on arguments from the introduction:

[COMPRESSION] Fingerprints should have a shorter representation than the associated image.

[ISOMETRY] Fingerprints should be isometry invariants. This is a natural requirement if we are dealing with fingerprints of objects that are metric spaces themselves. Here we will interpret images as Riemannian manifolds (see Section 5); this discussion seems to be new in the area of image analysis. In the context of image classification this is motivated by the fact that most content preserving operations are isometries, i.e. this map and its inverse preserve the arclength of all curves (see [14]). This includes changes of brightness, rotations, mirror operations, color rotations and inversions.

[SCALING] Fingerprints should optionally be made scaling invariants, in order to identify different resolutions of the same image.

There is another important criterion that is not directly related to invariance:

[SIMILARITY] Fingerprints of similar images should be similar. To develop a notion of similarity one needs to have a metric structure (or at least semimetric structure) for both images and fingerprints. Similarity of images may be measured by simple Euclidean distances. Refer to Section 7 to find out more about the way similarity may be measured for fingerprints resulting from spectra. One should note that the metric structure generated by comparing fingerprints does not necessarily correspond to a direct comparison of images.

This cannot be taken for granted: even small almost isometric changes in the object's geometry may change some feature's values non-continuously. This is especially true for discrete features, e.g. the number of "dark" regions on a photo (the number of components in a level set) and similar features.

Concerning criterion [COMPRESSION], any method of compression could be used in order to generate fingerprints. This way a compressed representation of an image could be viewed as its fingerprint. Nevertheless, traditional compression schemes tend to change significantly for small changes in the

data, thus violating [SIMILARITY]. This behaviour is even wanted in some application, e.g. with MD5 hashes (see [39]), where small changes to a file are required to cause a significant change in the hash. Nevertheless, methods used especially for geometry compression often change continuously, e.g. if they are based on Fourier transformations.

4. The Laplace operator and the Kirchhoff operator

In this section we will describe the Laplace operator, also known as the Laplacian. First we will take a look at the traditional continuous case; then we will examine the analogous construction in the discrete case.

Definition 2. Given a compact Riemannian manifold M of dimension n, $f : M \to \mathbb{R}$ with $f \in C^k(M)$ and $k \geq 2$. Let grad denote the gradient and div the divergence on the manifold M. Then

$$\Delta f := \operatorname{div}(\operatorname{grad} f)$$

defines the *Laplace operator* Δ on M.

For M being a domain of the Euclidean plane $M \subset \mathbb{R}^2$ the Laplace operator reduces to

$$\Delta f = \left(\frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2} \right). \tag{1}$$

In the non-Euclidean case for a Riemannian metric given by (g_{ij}) and $n = 2$ the Laplace operator can be expressed as (cf. [8])

$$\Delta f = \sum_{i=1}^{2} \sum_{j=1}^{2} g^{ij} \left(\frac{\partial^2 f}{\partial x_i \partial x_j} - \sum_{k=1}^{2} \Gamma_{ij}^k \frac{\partial f}{\partial x_k} \right) \tag{2}$$

with $(g^{ij}) := (g_{ij})^{-1}$ and Γ_{ij}^k being the well known Christoffel symbols (of the second kind, see e.g. [14]):

$$\Gamma_{ij}^m = \frac{1}{2} \sum_k g^{km} \left(\frac{\partial}{\partial_j} g_{ik} + \frac{\partial}{\partial_i} g_{jk} - \frac{\partial}{\partial_k} g_{ij} \right).$$

Note that these are invariant against isometries and scaling.

One can easily check that, for the Euclidean case, Eq. (2) specializes to Eq. (1). We will sometimes refer to the Laplace operator in the non-Euclidean case as the *Laplace–Beltrami operator*.

As a special case we could identify the map generating the manifold and the function the operator is applied to, i.e. given coordinate functions $F(u, v) := (x(u, v), y(u, v), z(u, v))$ we calculate ΔF with respect to the manifold given by F. Note that we use a generalized version of the Laplacian here that is defined coordinate wise by $\Delta F := (\Delta x, \Delta y, \Delta z)$. In this special case of a "double entry" where F serves the double purpose of representing the manifold *and* defining a vector field on this manifold we get the well known equation:

$$\Delta F(u, v) = 2H(u, v)n(u, v) \tag{3}$$

where H is the mean curvature and n is the surface normal of the point $F(u, v)$. This form where the parameterization of the manifold and the argument of the operator are being identified is often used for mesh smoothing (see [18,45] for the discrete case). Eq. (3) follows directly from Eq. (2) and the Gauss equations (see [14]):

$$\partial_i \partial_j F = \sum_k \Gamma_{ij}^k \partial_k F + h_{ij} n$$

$$\Rightarrow h_{ij} n = \partial_i \partial_j F - \sum_k \Gamma_{ij}^k \partial_k F. \tag{4}$$

Here h_{ij} are the coefficients of the second fundamental form. It follows that

$$\begin{aligned} 2Hn &= \sum_{ij} h_{ij} g^{ij} n \quad \text{by definition of } H \\ &= \sum_{ij} g^{ij} \left(\partial_i \partial_j F - \sum_k \Gamma_{ij}^k \partial_k F \right) \quad \text{with (4)} \\ &= \Delta F \quad \text{with (2).} \end{aligned}$$

There is a long tradition in studying the eigenvalues of the Laplace operator. Formally we define an eigenvalue λ by:

$$\Delta f - \lambda f = 0. \tag{5}$$

The (multi-)set of possible solutions λ to this equation is defined to be the *Laplacian spectrum* of M. This spectrum has a variety of interesting properties some of which make it interesting for image and shape identification.

- The spectrum is an isometry invariant (see [8]), i.e. if one maps M to an isometric manifold M' the spectrum remains unchanged. This fulfills criterion [ISOMETRY]. For experimental results and a detailed discussion we refer to [38].
- Continuous changes of the manifold's geometry result in continuous changes of the spectrum. Furthermore, we will demonstrate that small changes of the geometry yield likewise small changes of the spectrum; see [12], p. 366. This corresponds to criterion [SIMILARITY]. It is important to notice that a topological change of the manifold can change its spectrum radically. I.e. given the full disc, removing an infinitesimal small open disc at the center transforms the surrounding disc into a topological annulus. One will observe a significant change of the spectrum without changing the visual appearance. Luckily this restriction does not apply to images, as we will see in Section 5.
- The multiset of eigenvalues form an infinite but countable growing sequence, i.e.

  $$0 \leq \lambda_1 \leq \lambda_2 \leq \cdots.$$

  Nevertheless a finite number of these eigenvalues is sufficient to distinguish shapes in a practical situation. This is because the smaller eigenvalues correspond to "raw" features of the geometry (like area and boundary length) whereas the higher eigenvalues are related to finer details of the geometry. In a practical setup the number of eigenvalues being sufficient to distinguish images is much smaller than the data needed to present the whole image. This corresponds to criterion [COMPRESSION].

- For a uniform scaling of the manifold by an factor $a > 0$ in every dimension, the spectrum is scaled by $\frac{1}{a^2}$. We will show how to use this knowledge to cancel this effect of scaling, thus making the spectrum scaling invariant. This fulfills our criterion [SCALING].

Summarizing the points above, we take a finite number of eigenvalues of the Laplace operator as a fingerprint of the given manifold. We will see in Section 5 how images are related to manifolds.

4.1. The Laplacian with a mass density function

In extension to Eq. (5) we can introduce a *mass density function* ρ. This function assigns a mass density to each point of the manifold that influences the vibration of the material. Formally we can reformulate the equation as

$$\Delta f - \lambda \rho f = 0 \tag{6}$$

with $\rho : M \to \mathbb{R}$. A mass density of 1 means the usual density; therefore setting $\rho = 1$ everywhere leads to the classical problem in Eq. (5). The new formulation yields some interesting properties:

- The solutions of the problem depend continuously on changes of the manifold and of the mass density function (see [12], pp. 304).
- Given two isometric manifolds M and M' with an isometry I and mass densities ρ and ρ' that are conformal to the isometry, that is

  $$I(M) = M' \quad \text{and} \quad \rho(m) = \rho'(I(m)) \quad \forall m \in M$$

  then the spectra are identical. In different words, for a fixed mass density function the spectrum is isometry invariant.
- The statement above still holds if I is not an isometry but an isospectral transplantation (see Section 6.3), i.e. a function that maps a manifold to an isospectral twin.
- Scaling the mass density function by a factor of $1/k$ results in a spectrum also scaled by $1/k$. Let $\rho' := k\rho$ and $\Delta f = \lambda \rho f$, then:

  $$\Delta f = \lambda \rho f = \lambda \frac{1}{k} \rho' f = \frac{1}{k} \lambda \rho' f.$$

  For each eigenvalue λ of the problem with density ρ an eigenvalue $\lambda' := \frac{1}{k}\lambda$ of the problem with density ρ' exists with identical eigenfunction f.
- When scaling the manifold by k while keeping the mass density function, one gets

  $$\Delta' f = \frac{1}{k^2} \Delta f \quad \text{and} \quad \rho' = \rho$$

  $$\Rightarrow \Delta' f = \frac{1}{k^2} \Delta f = \frac{1}{k^2} \lambda \rho f = \frac{1}{k^2} \lambda \rho' f.$$

  Thus the spectrum is scaled by $\frac{1}{k^2}$.

4.2. Numerical calculations

In order to solve the stated eigenvalue problems 5 and 6 using a computer we need to discretize them. This can be accomplished by transforming the respective eigenvalue problem into an equivalent variational problem for the manifold M (cf. [38] for the steps involved in the transformation):

$$\int_M \langle \nabla \phi, \nabla f \rangle \, \mathrm{d}\xi = \int_M \phi \Delta f \, \mathrm{d}\xi = \lambda \int_M \rho \phi f \, \mathrm{d}\xi.$$

Here ξ represents the area element of M, ϕ is an arbitrary function from the same space as f, and ρ is the mass density function. For the classical problem we set $\rho = 1$. Now we approximate f by

$$f = \sum_{k=1}^{m} c_k \phi_k$$

with $\{\phi_1, \ldots, \phi_m\}$ being an FEM base for the intended space of solutions. Sometimes we will refer to these functions as *form functions*. Furthermore, we choose $\phi \in \{\phi_1, \ldots, \phi_m\}$. This way the problem is transformed into a generalized matrix eigenvalue problem (again see [38] for details):

$$A\vec{c} = \lambda B \vec{c} \tag{7}$$

$$A = (a_{jk}) = \left(\int_G \langle \operatorname{grad} \phi_j, \operatorname{grad} \phi_k \rangle W \mathrm{d}u \mathrm{d}v \right) \tag{8}$$

$$B = (b_{jk}) = \left(\int_G \rho \phi_j \phi_k W \mathrm{d}u \mathrm{d}v \right). \tag{9}$$

This problem can be solved using standard numerical libraries.

The existence of boundary conditions can have some impact on the choice of form functions also. Given a Dirichlet boundary condition $f(x) = 0$ for all $x \in \Gamma$, with Γ being the boundary, given a point P_i on the boundary, and given that the intended form functions have local character, that is each is associated to a special point and is only defined on a compact surrounding of that point, then the boundary condition mostly implies that the corresponding form function ϕ_i is zero. This is the case for most sets of popular form functions like piecewise linear function ("hat functions"), polynomial bases, etc. In this case every product with ϕ_i and its derivative is cancelled also. This implies that the corresponding entries in the FEM matrices A and B become zero.

Given a mesh that covers each pixel of an image, the matrices A and B can be computed in a time proportional to $O(n)$ for a n-pixel image. In practice this leaves the complexity of the approach up to the method for calculating the eigenvalues of A and B. Since the matrices are sparse one can greatly benefit from using a suitable method, e.g. the Lanczos algorithm.

4.3. The Laplace–Kirchhoff operator

There is a different version of the Laplace operator in the discrete case. It is defined as follows:

Definition 3. Given a node weighted graph $G = (A, M)$ with $A \in \mathbb{R}^{n \times n}$ being the adjacency matrix of the graph. In the case of an (edge-)weighted graph the entry A_{ij} contains the weight of the edge connecting nodes i and j. For an unweighted graph the weights are simply 0 or 1. Most of the time we will use the unweighted case, but there are many applications for the weighted graph also (see e.g. [16,18,45]).

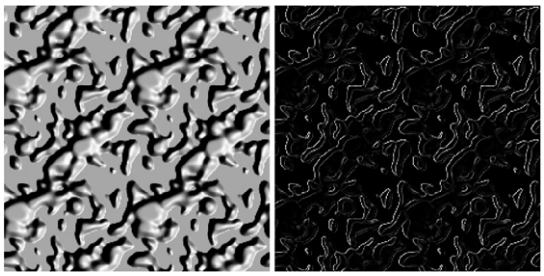

Fig. 1. Application of the Laplace–Kirchhoff operator to an image.

$M = \mathrm{diag}(m_1, \ldots, m_n) \in \mathbb{R}_+^{n\times n}$ is the diagonal matrix containing the weights of the nodes, if there are any. Let

$$D_{ii} = \sum_{j=1}^{n} A_{ij}$$

define the valence matrix of G, i.e. the diagonal matrix containing all degrees of the nodes of G. Then the discrete *Laplace operator* of G is given by

$$L = L(G) := M^{-1}(D - A).$$

The discrete Laplace operator is also known from the Matrix-Tree theorem by G. Kirchhoff and is therefore sometimes referred to as the Kirchoff operator. To distinguish the operator from the Laplacian in the continuous case we will refer to it as the *Laplace–Kirchhoff operator*.

The Laplace–Kirchhoff operator is a difference operator on G since it maps each function $f : G \to \mathbb{R}$ to a difference function $f' : G \to \mathbb{R}$. Here we identify f and f' with vectors from $\mathbb{R}^n$ by $f_i := f(i)$ and get $f' = L \cdot f$. It is then possible to interpret a gray value image as a node weighted graph by connecting each pixel with its four direct neighbours, and assigning each pixel its gray value as a weight. The gray value function then becomes a function on the graph where we can apply the Laplace–Kirchhoff operator to. Fig. 1 shows an image before and after applying its Laplace–Kirchhoff operator. The operator has no impact on an image corresponding to an eigenfunction. Its frequency response on the image can be characterized as a band pass convolution filter (see [35,19,22,23]).

The spectrum of the Laplace–Kirchhoff operator is given by the ordinary eigenvalue problem

$$Lx - \lambda x = 0 \tag{10}$$

with $x \in \mathbb{R}^n$. One can easily check that this is equivalent to

$$(D - A)x = \lambda M x \Leftrightarrow \det(D - A - \lambda M) = 0 \tag{11}$$

which is a generalized eigenvalue problem. The advantage of (11) over the formulation in (10) is, that all matrices are given in symmetric form, whereas L might be non-symmetric in general. This can be easily reformulated to the ordinary eigenvalue problem:

$$M^{-1/2}(D - A)M^{-1/2}x = \lambda x. \tag{12}$$

For symmetric eigenvalue problems more accurate numerical solutions are available.

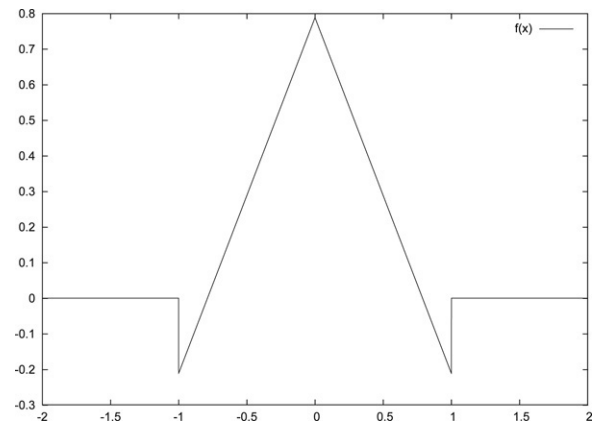

Fig. 2. Orthogonal FEM base function.

Note that again the main part of the calculations is finding the eigenvalues. Therefore the complexity of this approach is given by the chosen method for calculating the eigenvalues, e.g. the Lanczos algorithm.

The Laplace–Kirchhoff operator can be seen as a special discrete formulation of the Laplace–Beltrami operator. If we choose the FEM base from Section 4.2 to be the set $\{\phi_{ij} : (i, j) \in \mathbb{Z}^2\}$ we can define a FEM base for the space of functions defined on $\mathbb{R}^2$ with

$$\phi_{ij}(x, y) := \phi_i(x)\phi_j(y)$$
$$\phi_i(x) := \phi(x - i)$$
$$\phi(x) := \begin{cases} x + C & \text{for } -1 \le x < 0 \\ -x + C & \text{for } 0 \le x < 1 \\ 0 & \text{else} \end{cases}$$

using $C = \frac{3+\sqrt{3}}{6}$ (see Fig. 2). This yields the following matrices (by evaluating Eqs. (8) and (9); note that we choose pairs of numbers as indices here):

$$A = (a_{ij,kl}) = \begin{cases} 4 & \text{for } (i, j) = (k, l) \\ -1 & \text{for } |(i, j) - (k, l)| = 1 \\ 0 & \text{else} \end{cases}$$

$$B = (b_{ij,kl}) = \begin{cases} \dfrac{\sqrt{3} - 1}{3} & \text{for } (i, j) = (k, l) \\ 0 & \text{else} \end{cases}$$

after dividing by a factor of $\frac{\sqrt{3}-1}{3}$ on both sides of the equation. This conforms to the Laplace–Kirchhoff operator of a regular grid, except for a factor of $\frac{\sqrt{3}-1}{3}$ for the B matrix. This factor is of no practical importance, since it only leads to a scaling of the spectrum and can be divided from the spectrum after calculation. Now let there be a mass density function given by

$$\rho(x, y) := K_{[x][y]}$$

where $B \in \mathbb{N}^{n\times m}$ is a grey value image. For each square in $\mathbb{R}^2$ this function ρ is constant, therefore we get (again after dividing by a factor):

$$A = (a_{ij,kl}) = \begin{cases} 4 & \text{for } (i, j) = (k, l) \\ -1 & \text{for } |(i, j) - (k, l)| = 1 \\ 0 & \text{else} \end{cases}$$

$$B = (b_{ij,kl}) = \begin{cases} K_{ij}\dfrac{\sqrt{3}-1}{3} & \text{for } (i,j) = (k,l) \\ 0 & \text{else.} \end{cases}$$

This conforms to the Laplace–Kirchhoff operator with mass density function ρ. So far we have constructed discrete operators for the case of a uniformly spaced grid with grid size 1. Now let the grid have a grid size of $1/w$. We define

$$\phi^{[w]}(x) := \phi(wx)$$
$$\phi_i^{[w]}(x) := \phi^{[w]}(x-i) = \phi(w(x-i)).$$

By applying the substitution rule for integration we get:

$$\int_{-\infty}^{\infty} \phi_i^{[w]}\phi_j^{[w]}\mathrm{d}x = \frac{1}{w}\int_{-\infty}^{\infty} \phi_i\phi_j\mathrm{d}x$$
$$\int_{-\infty}^{\infty} \phi_i'^{[w]}\phi_j'^{[w]}\mathrm{d}x = \frac{1}{w}\int_{-\infty}^{\infty} w\phi_i'\phi_j'\mathrm{d}x = \int_{-\infty}^{\infty} \phi_i'\phi_j'\mathrm{d}x.$$

This shows that the mass matrix B is scaled by an additional factor of $1/w$ in the one-dimensional (1D) case; for the two-dimensional (2D) case B is scaled by $1/w^2$ and A is scaled by $1/w$. Again we can divide by $1/w$.

Summarizing the above results, the Laplace–Kirchhoff operator can be seen as a discrete approximation of the Laplace–Beltrami operator with mass density operator on a regular grid. Given that the mass matrix B is scaled by $w\frac{\sqrt{3}-1}{3}$, the resulting spectrum is an approximation of the spectrum of the Laplace–Beltrami operator. Note that according to Section 4.2 one needs to zero out the matrix entries corresponding to a boundary point to zero if there is a Dirichlet boundary condition.

Instead of interpreting the Laplace–Kirchhoff operator as a special case of the Laplacian resulting from FEM calculations, we could also view it as an approximation: Let B be a small geodesic disc with boundary ∂B and center point p, $\phi, \psi : B \to \mathbb{R}$ with $\psi \equiv 1$ on B and ϕ being C^2-smooth. Then we obtain (cf. [5])

$$\int_B \psi\Delta\phi\mathrm{d}\xi = -\int_{\partial B}\psi\frac{\partial}{\partial n}\phi\mathrm{d}s - \int_B \langle\nabla\phi, \nabla\psi\rangle\mathrm{d}\xi$$
$$\Rightarrow \int_B \Delta\phi\mathrm{d}\xi = -\int_{\partial B}\frac{\partial}{\partial n}\phi\mathrm{d}s - 0$$

as $\langle\nabla\phi, \nabla\psi\rangle = \sum_{ij} g^{ij}\partial_i\phi\partial_j\psi = 0$ since $\psi \equiv 1$. This gives:

$$\int_B \Delta\phi\mathrm{d}\xi = \int_{\partial B}\frac{\partial}{\partial n}\phi\mathrm{d}s. \tag{13}$$

It is well known that for the arclength l and area F of a geodesic circle the following approximations hold (with $K(p)$ being the Gaussian curvature at center p, cf. [5], p. 204):

$$\lim_{r\to 0}\frac{2\pi r - l}{r^3} = \frac{\pi}{3}K(p)$$
$$\Rightarrow l \approx 2\pi\left(r - \frac{r^3}{6}K(p)\right)$$
$$\lim_{r\to 0}\frac{\pi r^2 - F}{r^4} = \frac{\pi}{12}K(p)$$
$$\Rightarrow F \approx \pi\left(r^2 - \frac{r^4}{12}K(p)\right).$$

For a sufficiently small radius of B, $\Delta\phi$ is approximately constant; thus we have in geodesic polar coordinates (r, α):

$$\begin{aligned}
\int_B \Delta\phi\mathrm{d}\xi &= (\Delta\phi + o(r))\int_B 1\mathrm{d}\xi \\
&= \underbrace{\Delta\phi\left(\pi r^2\left(1 - \frac{r^2}{12}K(p)\right) + o(r^4)\right) + o(r)}_{\text{Area of the geodesic disc}} \\
&= \int_{\partial B}\frac{\partial}{\partial n}\phi\mathrm{d}s \qquad \text{using (13)} \\
&= \int_0^{2\pi}\frac{\partial}{\partial n}\phi\ \underbrace{\sqrt{G(r,\alpha)}}_{\text{Arclength element}}\ \mathrm{d}\alpha \\
&= \int_0^{2\pi}\frac{\partial\phi}{\partial n}\underbrace{\left(\left(r - \frac{r^3}{6}K(p)\right) + o(r^4)\right)}_{\text{Arclength element}}\mathrm{d}\alpha \\
&= \int_0^{2\pi}\frac{\partial\phi}{\partial n}r + \gamma r^3 K(p)\mathrm{d}\alpha \quad \gamma \in [-1, 1].
\end{aligned}$$

Let $\vec{r}_\alpha$ be the Cartesian vector of the point with coordinates (r, α). Then we have, using a linear approximation:

$$\begin{aligned}
&\frac{\partial\phi(r,\alpha)}{\partial n}r \\
&= \nabla\phi(r,\alpha)\cdot\vec{r}_\alpha \quad \text{(directional deriv. of } \phi \text{ along } r_\alpha) \\
&= \nabla\phi(0,0)\vec{r}_\alpha + \vec{r}_\alpha^{\,t}\begin{pmatrix}\phi_{xx}(0,0) & \phi_{xy}(0,0) \\ \phi_{xy}(0,0) & \phi_{yy}(0,0)\end{pmatrix}\vec{r}_\alpha + o(|\vec{r}_\alpha|^2) \\
&= \phi(r,\alpha) - \phi(0,0) + o(|\vec{r}_\alpha|^2).
\end{aligned}$$

Note that here (x, y) denote Riemannian normal coordinates with center p, implying that $\Delta\phi(p) = \phi_{xx}(0,0) + \phi_{yy}(0,0)$. Since $\int_0^{2\pi}\nabla\phi(0,0)\vec{r}_\alpha\,\mathrm{d}\alpha = 0$, and the same is true for the mixed partial derivatives, we get

$$\begin{aligned}
&\int_0^{2\pi}\frac{\partial\phi(r,\alpha)}{\partial n}r\mathrm{d}\alpha \\
&= \int_0^{2\pi}\vec{r}_\alpha^{\,t}\begin{pmatrix}\phi_{xx}(0,0) & 0 \\ 0 & \phi_{yy}(0,0)\end{pmatrix}\vec{r}_\alpha + o(|\vec{r}_\alpha|^2)\,\mathrm{d}x \\
&= \int_0^{2\pi}\phi(r,\alpha) - \phi(0,0) + o(|\vec{r}_\alpha|^2)\mathrm{d}\alpha.
\end{aligned}$$

Choosing a symmetric discretization of the geodesic disc B with angle steps $\mathcal{D} = \pi/n$ we obtain, with $\alpha_i := i\mathcal{D}$,

$$\begin{aligned}
&\left|\int_0^{2\pi}\vec{r}^{\,t}\begin{pmatrix}\phi_{xx}(0,0) & 0 \\ 0 & \phi_{yy}(0,0)\end{pmatrix}\vec{r}\mathrm{d}x\right. \\
&\qquad \left. - \sum_{i=1}^{2n}\mathcal{D}\vec{r}_{\alpha_i}^{\,t}\begin{pmatrix}\phi_{xx}(0,0) & 0 \\ 0 & \phi_{yy}(0,0)\end{pmatrix}\vec{r}_{\alpha_i}\right| \\
&\le r^2\frac{2\pi}{n}(|\phi_{xx}| + |\phi_{yy}|).
\end{aligned}$$

Using the above equations we get:

$$\int_0^{2\pi} \frac{\partial \phi}{\partial n} r + \gamma r^3 K(p) \mathrm{d}\alpha \quad \gamma \in [-1, 1]$$
$$= \sum_{i=1}^{2n} \mathcal{D}\left(\vec{r}_{\alpha_i}^t \begin{pmatrix} \phi_{xx}(0,0) & 0 \\ 0 & \phi_{yy}(0,0) \end{pmatrix} \vec{r}_{\alpha_i}\right.$$
$$\left. + \beta\left(r^2 \frac{2\pi}{n}(|\phi_{xx}| + |\phi_{yy}|)\right) + o(|\vec{r}_\alpha|^3)\right) \quad \beta \in [-1, 1]$$
$$= \sum_{i=1}^{2n} \mathcal{D}(\phi(i\mathcal{D}, r) - \phi(0,0) + o(r^2)) + O\left(\frac{1}{n}\right) r^2.$$

Therefore we get:

$$\Delta\phi(p) = \lim_{r\to 0}\left(\frac{\sum_{i=1}^{2n} \phi(i\mathcal{D}, r) - \phi(0,0) + O\left(\frac{1}{n}\right) r^2 + o(r^2)}{\pi r^2}\right).$$

Hence

$$\Delta\phi(p) = \lim_{r\to 0}\left(\frac{\sum_{i=1}^{2n} \phi(i\mathcal{D}, r) - \phi(0,0)}{\pi r^2}\right) + O\left(\frac{1}{n}\right).$$

This corresponds to the well known non-matrix formulation of the Laplace–Kirchhoff operator, i.e. for a function $f = (f_1, \ldots, f_n)$ defined on a graph with each f_i located at a node p_i we get the respective discrete form $\Delta f_i = \sum_j (f_j - f_i) w_{ij}$ for edge weights w_{ij}. In matrix formulation this is $\Delta f = -Lf$ with L as defined in Definition 3.

5. Images

In this section we will explain how to represent images in order to compute fingerprints. First we will restrict ourselves to gray value images to keep the description simple, but we will also explain how to extend the technique developed to images given in arbitrary color spaces.

Definition 4. Let $m, n, k, \in \mathbb{N}$, $G = \{0, \ldots, k-1\}$ and $B \in G^{m\times n}$. Then we call B a *discrete gray value image* with k steps.

We define a gray value image to be a matrix, but as one can easily see, this is the same as defining B to be a discrete height map $B : \{1, \ldots, m\} \times \{1, \ldots, n\} \to G$. We will use both definitions interchangeably. See Fig. 3 for an example. We will define a gray value of 0 to represent "black" and $k-1$ to represent "white", with the values in between defining shades of gray. This specification is arbitrary and – as we will see later – without effect on the actual calculations.

5.1. Fingerprints and the Laplace–Beltrami operator

One way to calculate a fingerprint for an image is to take the image as a discrete height map and transform it into a continuous manifold. Then the eigenvalues of its associated Laplace–Beltrami operator can be used as a fingerprint. At first glance it is not clear how to transform the image into a continuous form. An ad hoc technique could be to interpolate the manifold with any kind of 2D spline, e.g. a tensor product NURBS. However, the interpolation would take some time and there would also be calculation time needed for evaluating the resulting rather large spline representation of the surface.

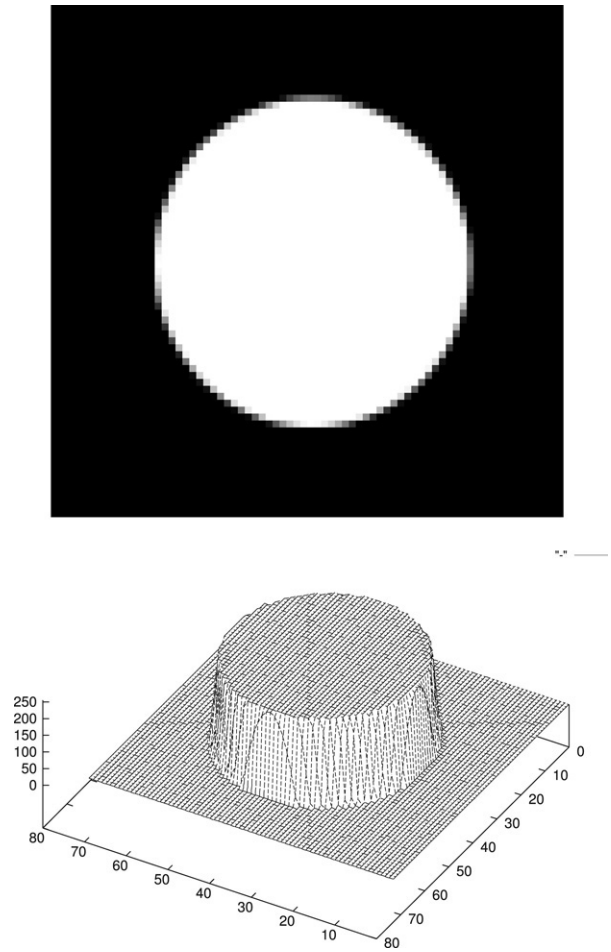

Fig. 3. An image and its height map.

An obvious solution to calculate the needed coefficients of the first fundamental form is to create a bilinear surface and then use the partial derivatives of that surface. For given pixel values z_1, z_2, z_3, z_4 the associated patch is:

$$f(h, v) := (1-v)((1-h)z_1 + hz_2) + v((1-h)z_3 + hz_4)$$

with local coordinates $h, v \in [0, 1]$ (see Fig. 4). We get the partial derivatives

$$f_h = v(z_4 - z_3) + (1-v)(z_2 - z_1)$$
$$f_v = h(z_4 - z_2) + (1-h)(z_3 - z_1)$$

and therefore the coefficients

$$\begin{aligned} g_{11} &= 1 + f_h^2 \\ &= 1 + (v(z_4 - z_3) + (1-v)(z_2 - z_1))^2 \\ g_{12} &= f_h f_v \\ &= (v(z_4 - z_3) + (1-v) \\ &\quad (z_2 - z_1))(h(z_4 - z_2) + (1-h)(z_3 - z_1)) \end{aligned}$$

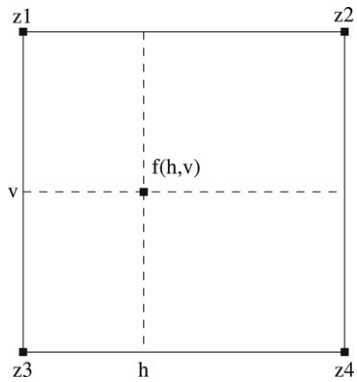

Fig. 4. Bilinear interpolation of partial derivatives.

$$\begin{aligned} g_{22} &= 1 + f_v^2 \\ &= 1 + (h(z_4 - z_2) + (1-h)(z_3 - z_1))^2. \end{aligned}$$

Given this Riemannian metric of an image one can use the finite element method to obtain the eigenvalues. See Section 4.2 and [37,38] for the details of this calculation.

In Section 4 we have noted that a change of the topology of a given object changes its spectrum fundamentally without affecting the point set of the object significantly. For images this is of no relevance since we take each image to be a height function defined on a rectangular area, i.e. the resulting manifold is always homeomorphic to the full disc. Therefore changes in topology cannot occur and we only have to deal with changes in geometry. As noted in Section 4, these changes take place continuously, which is the desired behaviour.

These small changes of the geometry could also include noise, compression artifacts and transmission errors on the images, given that these errors are kept within a certain range resulting in a similar geometry. Therefore the approach is robust against noise up to a certain degree.

Furthermore, we have noted that uniform scaling of a manifold by a modifies the associated Laplace–Beltrami spectrum by $1/a^2$. This might be problematic with images since images are usually either scaled along their x and y directions (which represents a uniform scaling of an *image*) or along their z direction only (which corresponds to a change of contrast). As a result of this, changing e.g. the contrast of an image modifies its spectrum in a rather unpredictable manner. To make the spectrum invariant against contrast changes we need to scale the gray values to the range $[0, \ldots, s_{\max}]$, where $s_{\max}$ represents the larger value of image width and image height. This way the effect of contrast changes are canceled and a scaling of the image in x and y directions becomes a uniform scaling along all axes. To make this spectrum scaling invariant we could simply divide it by its first non-zero eigenvalue. This way scaled spectra become identical. Note that this effect can be achieved also using a different method for computing the similarity of two spectra. For example, if we use the correlation coefficient instead of a simple Euclidean distance, scaling effects are canceled. See [38] for more detailed insights on this.

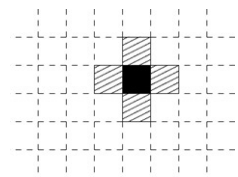

Fig. 5. The 4-neighborhood of a pixel.

5.2. Fingerprints from the Laplace–Kirchhoff operator

Another possibility is to choose the eigenvalues of the Laplace–Kirchhoff operator as a fingerprint. To accomplish this, the image is transformed into a node weighted graph. Each pixel is interpreted as a node whose weight corresponds to the given gray value. Then two nodes are connected iff they are adjacent in the image. One can choose different models of neighborhoods here; for our experiments we implemented the 4-neighborhood (see Fig. 5) for the sake of simplicity. Note that for arbitrary neighborhoods the spectrum of the Laplace–Kirchhoff operator should also converge against the underlying spectrum of the Laplace–Beltrami operator for finer resolutions of the image. For the resulting graph the Laplace–Kirchhoff operator according to Definition 3 is calculated and its eigenvalues are computed numerically. Note that one needs only the n smallest eigenvalues, so efficient techniques like the Lanczos algorithm (see [13]) can be applied. These eigenvalues, possibly normalized to avoid scaling effects, are then taken as a fingerprint.

5.3. Color images

Usually color images are given in some color space representation. The most commonly used color space is the RGB space, i.e. each color pixel is given by a triple (r, g, b) with r representing the red value, g the green value and b the blue value of the pixel. Most color spaces are three dimensional, although there are some four-dimensional (4D) spaces also, e.g. the CMYK space used in printing. We will outline a technique for three-dimensional (3D) spaces here; the extension to four dimensions is straightforward.

Definition 5. Let $m, n, k \in \mathbb{N}$, $G = \{0, \ldots, k-1\}$ and $B \in (G \times G \times G)^{m \times n}$. Then we call B *a discrete RGB value image with k steps*.

Just like in the monochrome case, we can interpret B as a map $B : \{1, \ldots, m\} \times \{1, \ldots, n\} \to G \times G \times G$. This is some kind of generalized height function. Like the monochrome image can be understood as a two-manifold embedded in 3D space, we can interpret the color image as a two-manifold embedded in 5D space. The manifold is parameterized by:

$$(u, v) \mapsto (u, v, r(u, v), g(u, v), b(u, v)).$$

Thus we get the following components for the first fundamental form:

$$\begin{aligned} g_{11} &= 1 + r_u^2 + g_u^2 + b_u^2 \\ g_{22} &= 1 + r_v^2 + g_v^2 + b_v^2 \\ g_{12} &= g_{21} = r_u r_v + g_u g_v + b_u b_v \end{aligned}$$

with r_u and r_v denoting the partial derivatives of $r(u, v)$ and similar definitions for g and b. We can then apply exactly the same interpolation techniques as in the monochrome case in Section 5.1 and then use the Laplace Beltrami operator to gain a fingerprint for the color image. See Section 6.2 for a discussion of advantages and disadvantages that may arise from this approach.

Interestingly this approach is compatible with the method developed for gray value images in Section 5.1. In RGB colour space a gray value image can be represented by a colour image with equal channels $f := r = g = b$. Thus we get:

$$g_{11} = 1 + 3f_u^2$$
$$g_{22} = 1 + 3f_v^2$$
$$g_{12} = g_{21} = 3f_u f_v.$$

This can be seen as the calculation done for a gray value image with height function $\sqrt{3}f$. This means that we get the same spectrum as with an ordinary height function f, only scaled by 1/3.

6. Isometry and isospectrality

In Section 5 we have presented two possibilities for constructing fingerprints of an image applying variants of the Laplace operator. The question we are dealing with now is to what extent a fingerprint is unique for a given image.

6.1. Isometry

Let us first discuss the spectrum of the Laplace–Beltrami operator. From Section 4, we know that isometric manifolds share identical spectra. So we would have to find out in which cases two images can be considered isometric. Furthermore, there are some rare cases where manifolds are isospectral (i.e. they share the same spectrum) but are non-isometric. We will discuss them in Section 6.3. In particular, two manifolds generated by the process described in Section 5.1 are isometric if their Riemannian metrics (g_{ij}) are identical:

Theorem 1. *Let the manifolds be defined by height functions $f : [0, m-1]\times[0, n-1] \to \mathbb{R}$ and $g : [0, m-1]\times[0, n-1] \to \mathbb{R}$ with $g^f_{ij} = g^g_{ij} =: g_{ij}$. Then:*

$$f = g + \alpha \quad \text{or} \quad f = -g + \alpha$$

for any $\alpha \in \mathbb{R}$.

Proof. We first show that the partial derivatives of f and g are identical up to their sign:

$$g_{11} = 1 + f_x^2 \wedge g_{11} = 1 + g_x^2$$
$$g_{22} = 1 + f_y^2 \wedge g_{22} = 1 + g_y^2$$
$$g_{12} = g_{21} = f_x f_y \wedge g_{12} = g_{21} = g_x g_y$$
$$\Rightarrow (1) \quad f_x^2 = g_x^2$$
$$(2) \quad f_y^2 = g_y^2$$
$$(3) \quad f_x f_y = g_x g_y$$
$$\Rightarrow \text{(from (1))} \quad |f_x| = |g_x|$$
$$\text{(from (2))} \quad |f_y| = |g_y|$$
$$\Rightarrow \text{(with (3))} \quad (f_x = g_x \wedge f_y = g_y) \vee (f_x = -g_x \wedge f_y = -g_y).$$

Now let $f_x = g_x$ and $f_y = g_y$. Then we have:

$$(1) \quad f = \int_{x_1}^{x_2} f_x \mathrm{d}x + C(y)$$
$$(2) \quad f = \int_{y_1}^{y_2} f_y \mathrm{d}y + D(x)$$
$$(3) \quad g = \int_{x_1}^{x_2} f_x \mathrm{d}x + E(y)$$
$$(4) \quad g = \int_{y_1}^{y_2} f_y \mathrm{d}y + F(x)$$

where $C(y)$ and $E(y)$ are functions depending solely on y, $D(x)$ and $F(x)$ solely on x. We get:

$$(5) \quad 0 = \int_{x_1}^{x_2} f_x \mathrm{d}x - \int_{y_1}^{y_2} f_y \mathrm{d}y + C(y) - D(x)$$
$$(6) \quad 0 = \int_{x_1}^{x_2} f_x \mathrm{d}x - \int_{y_1}^{y_2} f_y \mathrm{d}y + E(y) - F(x)$$

and finally:

$$(7) \quad 0 = C(y) - D(x) - E(y) + F(x).$$

This means that there has to be some constant α fulfilling:

$$D(x) - F(x) = \alpha = C(y) - E(y).$$

By substitution in (1) and (3), respectively (2) and (4) we get:

$$f = g + \alpha.$$

Now let $f_x = -g_x \wedge f_y = -g_y$. From an analogous argument we get:

$$f = -g + \alpha.$$

There is an alternative proof of Theorem 1 using knowledge about Riemannian manifolds: Given two surfaces defined by height functions f and g we can define a *solid height function* for f by:

$$S_f(x, y; z) := \begin{pmatrix} x \\ y \\ f(x, y) + z \end{pmatrix} \quad \text{with } z \in \mathbb{R}^n$$

and the same for g. One can easily check that its Riemannian (volume) metric is given by:

$$(g^S_{ij}) = \begin{pmatrix} g^f_{11} & g^f_{12} & f_x \\ g^f_{21} & g^f_{22} & f_y \\ f_x & f_y & 1 \end{pmatrix} = \begin{pmatrix} 1 + f_x^2 & f_x f_v & f_x \\ f_x f_v & 1 + f_v^2 & f_y \\ f_x & f_y & 1 \end{pmatrix}.$$

We know from our elementary considerations at the beginning of this proof that $(f_x = g_x \wedge f_y = g_y) \vee (f_x = -g_x \wedge f_y = -g_y)$, hence for S_g we get

$$(g^S_{ij}) = \begin{pmatrix} g^g_{11} & g^g_{12} & g_x \\ g^g_{21} & g^g_{22} & g_y \\ g_x & g_y & 1 \end{pmatrix}.$$

This resolves to one of

$$\begin{pmatrix} 1+f_x^2 & f_x f_v & f_x \\ f_x f_v & 1+f_v^2 & f_y \\ f_x & f_y & 1 \end{pmatrix} \begin{pmatrix} 1+f_x^2 & f_x f_v & -f_x \\ f_x f_v & 1+f_v^2 & -f_y \\ -f_x & -f_y & 1 \end{pmatrix}.$$

Therefore S_f and S_g are isometric. Two solids in Euclidean three space are isometric if and only if they are congruent; that is, they are related by a series of Euclidean motions (see e.g. [27], p. 88). Hence all their faces including the original height functions must be identical up to Euclidean motions. □

Theorem 1 shows that if two images share the same Riemannian metric, one is either a brighter version of the other, its negative or a combination of both. However, in order to be isometric two images do not need to share the same Riemannian metric of their height functions. They could also be rotations or mirror images of each other. This still perfectly conforms to criterion [ISOMETRY]. Furthermore, there are some cases where isometries result from a change of parameterization. For example, let

$$f_1(x, y) := \sqrt{1-(x-1)^2}$$
$$f_2(x, y) := \frac{x}{2}\sqrt{\pi^2-4}.$$

One can easily check that these functions do not have identical Riemannian metrics of their height functions; however, their graphs on the parameter space $[0, 2] \times [0, 2]$ are both isometric to the Euclidean rectangle $[0, \pi] \times [0, 2]$. This way they are related by an isometry flattening the cylindrical surface to the rectangle. Fig. 6 shows the corresponding manifolds. Another possibility to construct isometric images is to determine the maximal and minimal partial derivative along one axis and then rotate the image at angles α and $-\alpha$ along the other axis (see Fig. 7). Since one needs to make sure that the rotated manifold remains describable by a height function, α can be calculated as

$$\alpha < 90 - \arctan m_{\max} \quad \text{and} \quad \alpha < 90 + \arctan m_{\min}$$

with $m_{\min}$ and $m_{\max}$ being the minimal respectively maximal derivative (see Fig. 8). Fig. 9 shows an image and the same image rotated by 5°. Note that the image needs to be rather blurred to allow even a small possible rotation angle of 5°. For a monochrome image $m_{\min}$ and $m_{\max}$ are determined by the maximal possible difference of two gray values, i.e. $g-1$:

$$m_{\min} = -g+1 \quad \text{and} \quad m_{\max} = g-1$$

(see Fig. 10). Typically $g-1 = 255$ so it follows:

$$\alpha < 90 - 89.775312 = 0.22468818.$$

This gives a maximal rotation angle of 0.22468818°, resulting for an original image width of w in a width of $w \cdot \cos\alpha$ after the rotation. In a typical situation with $g-1 = 255$ and $\alpha = 0.22468818$ this yields a width of 99.999231% of the original, that is for a total image width of 100 000 pixels less than 1 pixel difference! Effectively the rotation would be invisible.

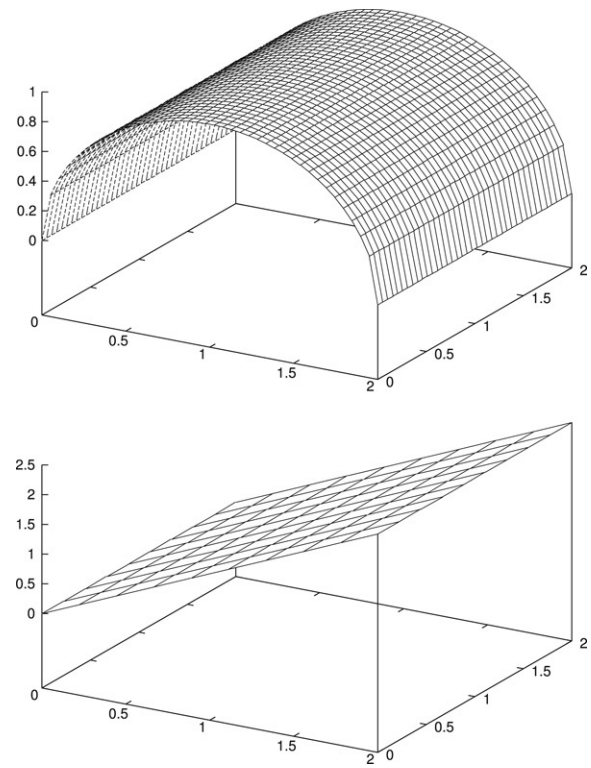

Fig. 6. Isometric height functions.

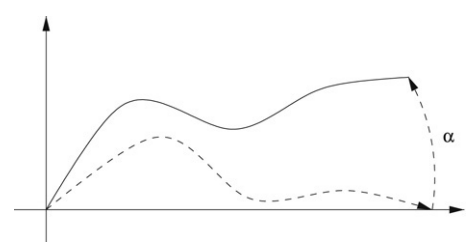

Fig. 7. Rotated height function.

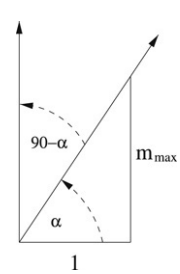

Fig. 8. Computation of the rotation angle.

Therefore in practice one rarely encounters images that meet the above conditions, so isometry usually means that two images are identical up to Euclidean motions and mirror operations.

6.2. Isometry and color images

Using the Laplace–Beltrami spectrum for color images introduces some differences from the monochrome case that can be both advantageous and disadvantageous. First of all,

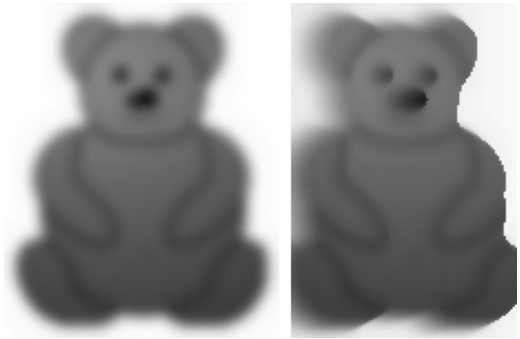

Fig. 9. Original image and by 5° rotated version.

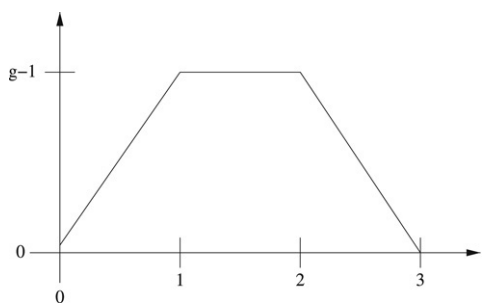

Fig. 10. Maximal gray value difference.

there are some well known cases of isometric but non-congruent two-manifolds in spaces with more than two dimensions. A popular example is the pair catenoide and helicoide (cf. [14]): These are isometric but not congruent. Since they are isometric we can construct a parameterization so that both share identical g_{ij} values. This way we have defined two color images (namely maps from two dimensions to three dimensions) that are isometric but look notably different. This can be stated more formally as follows. Let $f, g : \mathbb{R}^2 \to \mathbb{R}^3$ be maps from the parameter space to the color space defining the two images. Clearly most of the time the manifolds defined by f and g are degenerate (cf. Fig. 14). This is a result of images containing the same color in possibly adjacent places: the extreme case is a uniformly black image where the resulting gamut collapses to a single point. Nevertheless, the above examples show that there are non-degenerate cases also. Suppose the resulting manifolds in $\mathbb{R}^3$ are isometric. Then there exists some parameterization so that the g_{ij} values are identical for both manifolds. Suppose f, g are given by such a parameterization. Now define two two-manifolds in five dimensions by the maps $F(u, v) := (u, v, f(u, v))$ and $G(u, v) := (u, v, g(u, v))$. Then the first fundamental form for the manifold F are given by

$$g_{11}^F = 1 + g_{11}^f \cdot g_{11}^f \quad g_{22}^F = 1 + g_{22}^f \cdot g_{22}^f \quad g_{12}^F = g_{12}^f$$

and they are identical for G. Fig. 11 shows the resulting helicoide and catenoide images with marking lines between the areas where each prime color (red, green and blue) dominates to emphasize the difference. Fig. 12 shows the color gamuts of these images. Note that these gamuts approximate the geometry of helicoide and catenoide in RGB-space. Thus we have the disadvantage of non-similar images sharing identical spectra

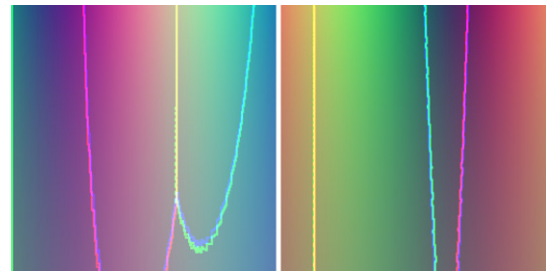

Fig. 11. Color images associated to helicoide and catenoide. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

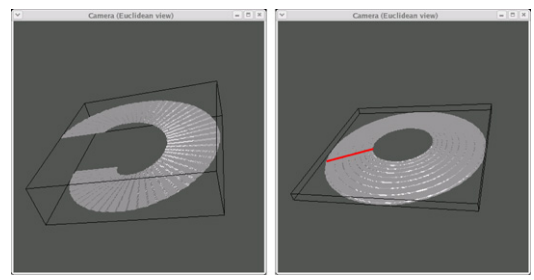

Fig. 12. Color gamuts of helicoide and catenoide. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

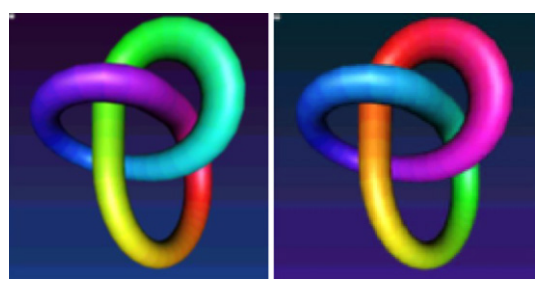

Fig. 13. Color rotation. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

here. Of course one could calculate separate spectra for each of the colors, but in that case we lose the invariance with respect to color rotations, an advantage of this approach which we will describe below.

Another possibility to create isometric manifolds associated with color images is to apply Euclidean motions in color space, most notably color rotations. In other words: the fingerprints are invariant with respect to color rotations. This can be a big advantage in some applications since color rotations rather represent a change in the look of an image but not in its intended content. Fig. 13 shows an example of a color rotation by 90°. Fig. 14 shows the two images from Fig. 13 in RGB-space (their color gamuts). Note that the resulting two-manifolds in three dimensions are congruent up to rotation. Also note that the gamuts almost fill the entire RGB cube. In such cases color rotation angles are naturally limited to multiples of 90°. Further Euclidean motions in color space correspond to translations. These are effectively linear changes of the dominant image color: once again the principal content of the image remains unchanged. Another advantage is that the fingerprints are

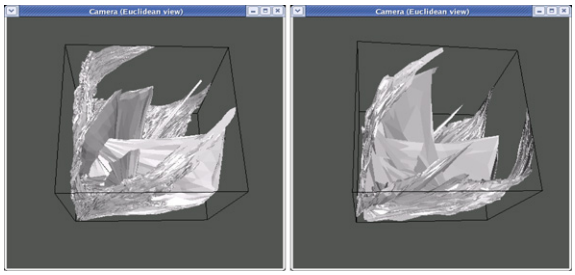

Fig. 14. Color gamuts. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

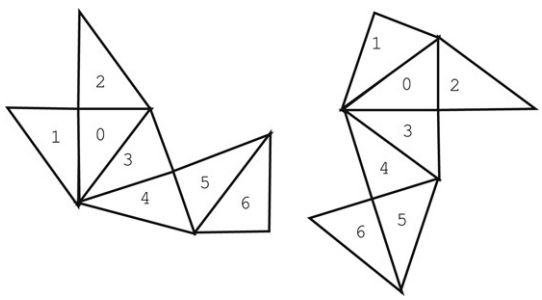

Fig. 15. Two isospectral domains.

invariant to changes of the color space, given that the color spaces are connected by an Euclidean motion. For example, one can obtain a representation in CMY space from an RGB representation by the transformation

$$\begin{pmatrix} c \\ m \\ y \end{pmatrix} = \begin{pmatrix} 1-r \\ 1-g \\ 1-b \end{pmatrix}.$$

Clearly this is an Euclidean motion (an inversion and a translation); thus the fingerprint does not change.

6.3. Isospectrality

On the other hand, there are cases where two manifolds share an identical spectrum while they are not isometric. We have seen how to represent monochrome images as two-manifolds; thus we have to investigate if it is possible for a pair of two-manifolds to be *isospectral*. Indeed, examples for non-isometric but isospectral two-manifolds have been found; see [11] for some planar domains. Fig. 15 shows a typical pair. However, none of those manifolds can be expressed by height functions over a convex domain, i.e. their parameter space must be a non-convex domain. Thus it is very unlikely that there exist isospectral monochrome images, since they are defined by height functions over rectangular (and therefore convex) domains. Fig. 16 shows a possible parameter domain for the manifolds from Fig. 15. From all we know so far isospectrality seems to be a rare phenomenon. Only pairs of isospectral domains have been found in two dimensions and all of them are non-convex. Therefore isospectrality should not be a serious problem and should not prevent the discrimination of images via their spectra.

When using the Laplace–Kirchhoff operator instead of the Laplace–Beltrami operator, one has to deal with isospectral graphs instead of isospectral manifolds. Just like in the continuous case, the phenomenon is not understood completely, but theories exist of how to construct such graphs; see e.g. [24]. Halbeisen and Hungerbühler describe a method for obtaining an isospectral "twin" for a given graph. An interesting property of their method is that the constructed graph is either not simple or has at least two more nodes than the original graph.

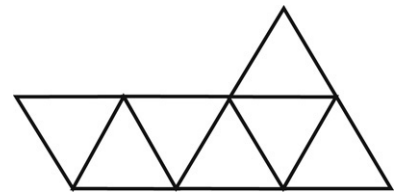

Fig. 16. Parameter domain for the isospectral domains.

7. Spectra and fingerprints

Formally we have defined the Laplacian spectra in Eqs. (5), (6) and in Eq. (10). Following the first definition a spectrum is an infinite countable multiset of real numbers, i.e. a sequence $0 \le \lambda_0 \le \lambda_1 \le \cdots \le \infty$. It is a multiset since one can have eigenvalues with multiplicities larger than 1. According to the second definition it is a finite multiset of real numbers with $\lambda_0 = 0$. The first eigenvalue is always zero if there is no boundary condition, since the sum of all rows of the matrix $D - A$ is zero; therefore its determinant is zero.

We define the fingerprint of an image to be a suitable finite subset of one of the possible spectra. We will see in Section 8 that for practical applications most of the time the first $n \le 10$ eigenvalues suffice.

To effectively compare two images one has to compare their associated fingerprints. This can be accomplished by defining a suitable metric on the space of the fingerprints. If we choose our fingerprints to be the first n eigenvalues we are dealing with the vector space $\mathbb{R}^n$, where we can choose between a large number of well known metrics. For our tests we have used different p-norms given by

$$d_p(u, v) := \left(\sum_{i=0}^{n-1} |u_i - v_i|^p \right)^{\frac{1}{p}}$$

with $u = (u_1, \ldots, u_n)$ and $v = (v_1, \ldots, v_n)$ being fingerprints. Furthermore, we have tested the Hausdorff distance and the Pearson correlation distance:

$$d_c(u, v) := 1 - \left| \frac{(n-1) \sum_{i=0}^{n-1} (u_i - \bar{u})(v_i - \bar{v})}{\sum_{i=0}^{n-1} (u_i - \bar{u})^2 \sum_{i=0}^{n-1} (v_i - \bar{v})^2} \right|$$

where $\bar{u}$ and $\bar{v}$ denote the arithmetic means of u and v. We observed that for most applications the Euclidean distance d_2 yields acceptable results (see Section 8) while being easy to implement.

Since we know that scaling an image can be transformed to an associated scaling of the fingerprint (see Sections 4 and 5),

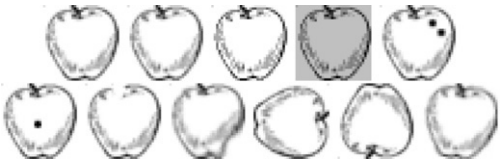

Fig. 17. Original test image and modified versions.

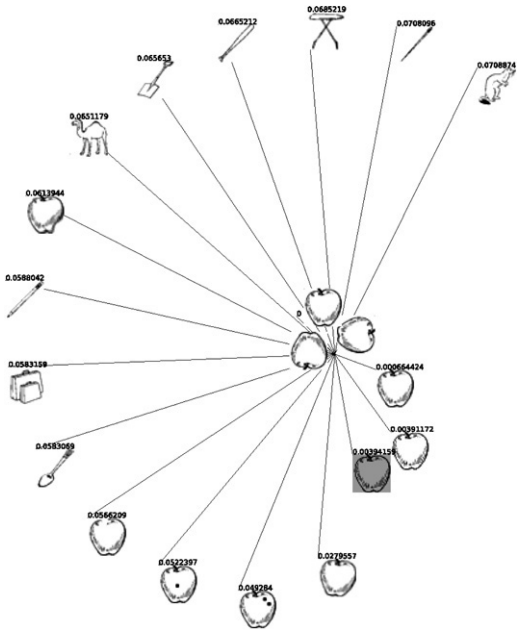

Fig. 18. Distances for the Laplace–Kirchhoff operator.

one could use a distance that is insensitive against linear scaling. For example, the correlation distance has this property by definition. Other possibilities include dividing the fingerprint by its first non-zero eigenvalue or normalizing the slope of the best fitting line of the sequence of eigenvalues to 1 before comparing. This way one makes sure that fingerprints being in fact scaled variations of each other are considered identical.

8. Examples

To test the implementation of the methods described above an image from the collection of Snodgrass [44] was modified (see Fig. 17). We calculated the eigenvalues of the Laplace–Kirchhoff operator with Dirichlet boundary condition. All images were scaled to 32 × 32 pixels to allow a fast computation. The fingerprints were divided by the second eigenvalue (since the first one is always zero), cropped to 10 eigenvalues and compared via Euclidean distance. Fig. 18 shows the result for the 15 best matches. The computed distances are shown in the figure; note that the distances plotted are not linearly scaled: the shorter distances are scaled up.

It can be observed that images that are not very similar in terms of human perception may be considered similar by our methods. This is a result of interpreting an image as a height function, i.e. taking the (white) background of an image as a part of the described shape (see also Fig. 23). If one does not wish to consider the background one could separate the content of the image from a possible background (see Section 9).

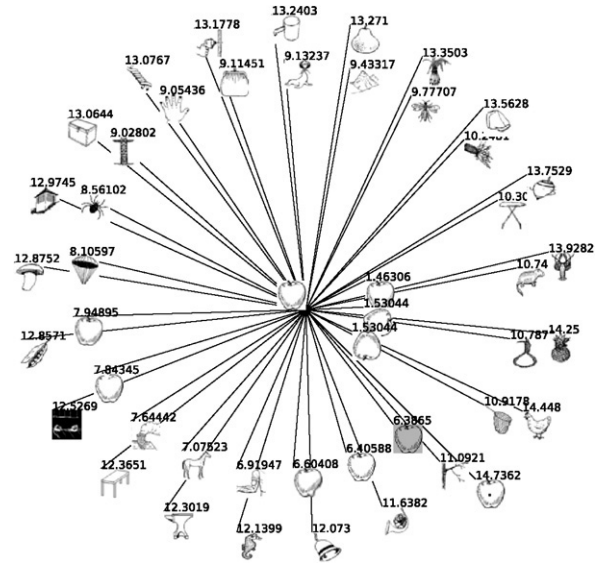

Fig. 19. Distances for the Laplace–Beltrami operator.

The experiment was repeated using the Laplace–Beltrami operator obtained from the height function. The fingerprints were computed using 338 elements and cubic form functions (cf. [38] for details of the FEM calculations used). They were divided by the first eigenvalue and compared via Euclidean distance. Fig. 19 shows the result.

One observes that both methods – using the Laplace–Kirchhoff operator and using the Laplace–Beltrami operator – perform similarly. In fact there is no notable difference in terms of run time. However, while the Laplace–Kirchhoff operator is more easily implemented, the Laplace–Beltrami variants open up the possibility to use a coarser mesh and thus save some time while computing eigenvalues while the size of the matrices for the Laplace–Kirchhoff operator is fixed for a given image size.

To test the robustness against scaling, each image from the collection of Rossion (see [40]) was scaled by a factor of 2 and added to the collection. The fingerprints were calculated using the Laplace–Beltrami operator obtained from the height function with 338 finite elements. They were compared using Euclidean distances with best fitting lines (see Section 7). For 511 of 532 the double sized images were the second best fits (behind the respective image itself); for the remaining 21 images the double sized versions were the third best fits. This corresponds to a reliability of about 96% for scaling.

The experiment was repeated with the images being changed in contrast instead of size. For each image a copy with 50% of the original contrast was added to the collection. In this case all of the 532 images could be matched to their low contrast counterparts. This corresponds to a reliability of 100% for change of contrast. Combining changes of scale with changes of contrast again gives the same results like changing scale only, i.e. 96% reliability. This makes the method especially useful in setups where combinations of such operations occur.

By simply averaging the results of the preceding experiments one could conclude that the method performs similarly to the combined Wavelet–Fourier approach of Sabharwal and Subramanya (see [41]). However, one should keep in mind that

Fig. 20. Original colour test image and modified versions.

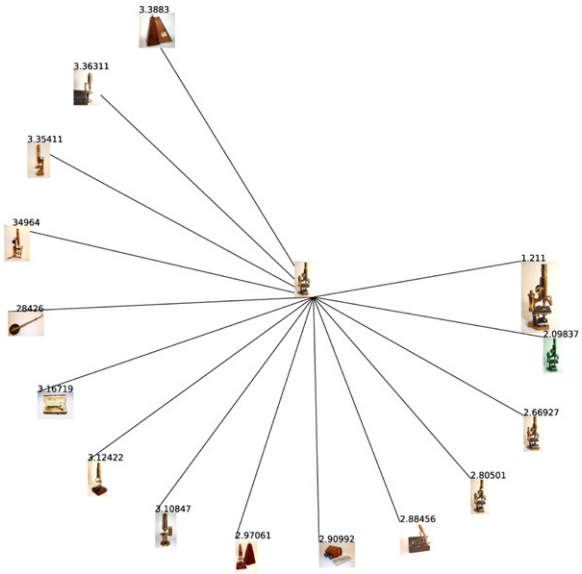

Fig. 21. Distances for the Laplace–Beltrami operator (colour images).

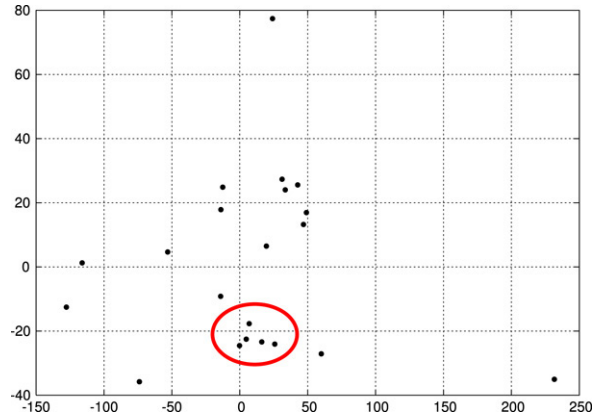

Fig. 22. Cluster for the colour images (Laplace–Beltrami).

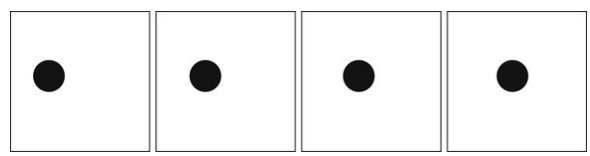

Fig. 23. One parameter family of images.

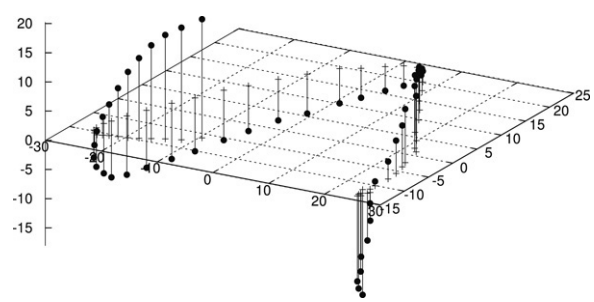

Fig. 24. One parameter curve of the fingerprints.

our test set was considerably larger (532 against only 78), and using less information: Sabharwal and Subramanya used 64 bytes of information while we compared the images using their first 10 eigenvalues amounting to only 40 bytes. For shorter set of images, e.g. 78 like the test set of Sabharwal and Subramanya, even fewer eigenvalues can be used, e.g. the first three ($3 \cdot 4 = 12$ bytes). This said, our method performs as good while using shorter representations which is a substantial advantage when working with large data bases where typical feature vectors should not be of dimension much greater than 10.

To test the methods developed for colour images an image from the collection of Kambeck [25] was modified in different ways (see Fig. 20). Thirty eigenvalues were calculated using the Laplace–Beltrami operator for generalized height functions with 338 finite elements and cubic form functions. They were compared using the Euclidean distance with best fitting lines. Fig. 21 shows the calculated distances for the best matches. Fig. 22 shows an MDS plot of the best matches depicted in Fig. 21, that means the resulting spectra projected to a 2D space using the standard MDS method. One can identify the cluster of the microscope images. Of course projecting the high dimensional spectra to a very low dimensional space means a massive loss of information, resulting in the formation of additional clusters, and thus cannot serve more than purposes of illustration. For practical applications one should work with more than two dimensions. Using the Pearson correlation distance (see Section 7) yielded similar good results in this case, whereas using the Hausdorff distance showed inferior results. This is because for the Hausdorff distance to become small it is sufficient that there are two eigenvalues at a relatively small distance, while for the other distances all eigenvalues need to be near their corresponding partners.

Finally we generated a family of images containing a moved full disc (see Fig. 23). Clearly, this is a one parameter family of different images, so one should predict the set of fingerprints to depend on one parameter as well. Note that we interpret these images to be different since we chose the height function of the image as a manifold for computing the spectrum. If one wishes to treat such images as principally the same image, more precisely as the same content, one has to cut the background from the represented content and use only this as the manifold. Fig. 24 shows an MDS plot of the calculated fingerprints. It forms a one parameter curve.

9. Conclusion and outlook

We have presented three methods for obtaining fingerprints from discrete monochrome or color images. Namely,

the Laplace–Beltrami operator with height functions, the Laplace–Beltrami operator with density functions, and its discrete counterpart the Laplace–Kirchhoff operator were used. We have shown an interlink between the discrete Laplace–Kirchhoff operator and the Laplace–Beltrami operator. Furthermore, we have introduced concepts from the theory of Riemannian geometry into the field of image fingerprints. We have demonstrated these techniques to work in a set of practical situations. The methods were shown to be especially useful in the presence of rotations or color rotations, changes of contrast and scale, and combinations of all these operations, since the calculations are invariant against such transformations. We have shown that our method uses substantially less information than established techniques for discriminating collections of images, while maintaining a high reliability. This is especially useful for data bases of images where high dimensional searches are very cost intensive (see [4,3]).

Future work will include investigations on how to apply the Laplace–Kirchhoff operator and the Laplace–Beltrami operator with density functions to color images.

As shown in Section 8 a drawback for some applications could be the representation of the image content together with its background as a single height function. A preprocess of separating the background from the content should be investigated.

Furthermore, methods would have to be provided to cancel the effects of special non-Euclidean transformations (e.g. most color space transformations) on color images. These problems could probably be solved by transforming images in a preprocess in such a way that the non-Euclidean transformations are mapped to Euclidean transformations, although this could mean that the former Euclidean transformations become non-Euclidean.

References

[1] Belkin M, Niyogi P. Laplacian eigenmaps for dimensionality reduction and data representation. Technical report TR 2002-01. Univ. Chicago, Dept. Comp. Sci. and Statistics; 2002.

[2] Bengio Y, Delalleau O, Roux NL, Paiement J-F, Vincent P, Ouimet M. Spectral dimensionality reduction. Technical report 2004s-27. CIRANO; 2004.

[3] Berchtold S, Böhm C, Braunmüller B, Keim DA, Kriegel H-P. Fast parallel similarity search in multimedia databases. In: SIGMOD '97: Proceedings of the 1997 ACM SIGMOD international conference on management of data. ACM Press; 1997. p. 1–12.

[4] Berchtold S, Keim DA, Kriegel H-P. The X-tree: An index structure for high-dimensional data. In: Vijayaraman TM, Buchmann AP, Mohan C, Sarda NL, editors. Proceedings of the twenty-second international conference on very large data bases. Los Altos (CA 94022, USA): Morgan Kaufmann Publishers; 1996. p. 28–39.

[5] Blaschke W, Leichtweiß K. Elementare Differentialgeometrie. 5th ed. Berlin: Springer; 1973.

[6] Bronstein AM, Bronstein MM, Kimmel R. Expression-invariant 3d face recognition. In: Kittler J, Nixon MS, editors. Proc. AVBPA. Lecture notes in comp. science, vol. 2688. Berlin, Heidelberg: Springer Verlag; 2003. p. 62–9.

[7] Bülow T. Spherical diffusion for 3d surface smoothing. IEEE Trans Pattern Anal Mach Intell 2004;26(12).

[8] Chavel I. Eigenvalues in Riemannian geometry. New York, San Francisco, London: Academic Press; 1984.

[9] Chung F. Spectral graph theory, vol. 92. Providence (RI): American Mathematical Society; 1997.

[10] Cohen S, Guibas L. Shape-based image retrieval using geometric hashing. In: Proc. of the ARPA image understanding workshop. 1997. p. 669–74.

[11] Conway J, Semmler K-D, Buser P, Doyle P. Some planar isospectral domains. Internat Math Res Notices 1994;9:391–400.

[12] Courant R, Hilbert D. Methoden der mathematischen physik. 4th ed. Berlin: Springer; 1993.

[13] Cullum JK, Willoughby RA. Lanczos algorithms for large symmetric eigenvalue computations. Boston: Birkhäuser; 1985.

[14] DoCarmo MP. Differential geometry of curves and surfaces. Prentice-Hall; 1976.

[15] Dong S, Bremer P-T, Garland M, Pascucci V, Hart JC. Quadrangulating a mesh using Laplacian eigenvectors. Technical report UIUCDCS-R-2005-2583TC. University of Illinois at Urbana-Champaign; 2005.

[16] Dong S, Bremer P-T, Garland M, Pascucci V, Hart JC. Spectral surface quadrangulation. ACM Trans Graph 2006;25(3):1057–66.

[17] Elad (Elbaz) A, Kimmel R. On bending invariant signatures for surfaces. IEEE Trans Pattern Anal Mach Intell 2003;25(10):1285–95.

[18] Field DA. Laplacian smoothing and Delaunay triangulations. Commun Appl Numer Methods 1988;4:709–12.

[19] Foley JD, van Dam A, Feiner SK, Hughes JF. Computer graphics (2nd ed. in C): Principles and practice. Boston (MA, USA): Addison-Wesley Longman Publishing Co., Inc.; 1996.

[20] Gaede V, Günther O. Multidimensional access methods. ACM Comput Surv 1998;30(2):170–231.

[21] Glassner AS. Principles of digital image synthesis, vol. 1. San Francisco (CA, USA): Morgan Kaufmann Publishers Inc.; 1994.

[22] Glassner AS. Principles of digital image synthesis, vol. 1. San Francisco (CA): Morgan-Kaufmann; 1995.

[23] Grady L, Schwartz EL. The graph analysis toolbox: Image processing on arbitrary graphs. Technical report CAS/CNS-TR-03-021. Boston (MA): Department of Cognitive and Neural Systems, Boston University. 2003.

[24] Halbeisen L, Hungerbühler N. Generation of isospectral graphs. J Graph Theory 1999;31:255–65.

[25] Kambeck BU. Antique microscopes and other scientific instruments and books. http://www.kambeck.de 2005.

[26] Karni Z, Gotsman C. Spectral compression of mesh geometry. In: Akeley K, editor. Siggraph 2000, computer graphics proceedings. ACM Press/Addison Wesley Longman; 2000. p. 279–86.

[27] Klingenberg WPA. Riemannian geometry. 2nd revised ed. Berlin, New York: Walter de Gruyter; 1995.

[28] Knuth DE. Sorting and searching. 2nd ed. The art of computer programming, vol. 3, Reading (MA): Addison-Wesley; 1998.

[29] Ko KH, Maekawa T, Patrikalakis NM, Masuda H, Wolter F-E. Shape intrinsic fingerprints for free-form object matching. In: Elber G, Shapiro V, editors. Proceedings of the eighth ACM symposium on solid modeling and applications. IEEE Computer Society; 2003. p. 196–207.

[30] Ko KH, Maekawa T, Patrikalakis NM, Masuda H, Wolter F-E. Shape intrinsic properties for free-form object matching. ASME J Comput Inform Sci Eng 2003;3(4):325–33.

[31] Loncaric S. A survey of shape analysis techniques. Pattern Recog 1998; 31(8):983–1001.

[32] Long F, Zhang H, Feng D. Fundamentals of content-based image retrieval. Berlin: Springer; 2003.

[33] Mohar B. Some applications of Laplace eigenvalues of graphs. NATO ASI series C, vol. 497. Dordrecht: Kluwer; 1997. p. 227–75.

[34] Müller H, Michoux N, Bandon D, Geissbuhler A. A review of content-based image retrieval systems in medical applications — clinical benefits and future directions. Int J Med Inform 2004;73(1):1–23.

[35] Peinecke N. Eigenwertspektren des Laplaceoperators in der Bilderkennung. Norderstedt, Germany: Books on demand GmbH; 2006.

[36] Pi MH, Li CH, Li L. Image indexing based on fractal feature. In: ICME. IEEE; 2004. p. 899–902.

[37] Reuter M, Wolter F-E, Peinecke N. Laplace-spectra as fingerprints for shape matching. In: SPM '05: Proceedings of the 2005 ACM symposium on solid and physical modeling. New York (NY, USA): ACM Press; 2005. p. 101–6.

[38] Reuter M, Wolter F-E, Peinecke N. Laplace–Beltrami spectra as "shape-dna" of surfaces and solids. Computer-Aided Design 2006;38(4):342–66.
[39] Rivest RL. The md5 message-digest algorithm. Technical report RFC 1321. Internet Activities Board; 1992.
[40] Rossion B, Pourtois G. Revisiting Snodgrass and Vanderwart's object set: The role of surface detail in basic-level object recognition. Perception 2004;33:217–36.
[41] Sabharwal CL, Subramanya SR. Indexing image databases using wavelet and discrete Fourier transform. In: SAC '01: Proceedings of the 2001 ACM symposium on applied computing. New York (NY, USA): ACM Press; 2001. p. 434–9.
[42] Sclaroff S, Pentland A. Modal matching for correspondence and recognition. IEEE Trans Pattern Anal Mach Intell 1995;17(6):545–61.
[43] Shilane P, Kazhdan M, Min P, Funkhouser T. The Princeton shape benchmark. In: Proc. shape modeling international. 2004.
[44] Snodgrass J, Vanderwart M. A standardised set of 260 pictures: Norms for name agreement, image agreement, familiarity and visual complexity. J Exp Psycho 1980;6(3):174–215.
[45] Taubin G. Geometric signal processing on polygonal meshes. In: State of the art report, EUROGRAPHICS. 2000. p. 81–96.
[46] Veltkamp R, Hagedoorn M. State-of-the-art in shape matching. Technical report UU-CS-1999-27. The Netherlands: Utrecht University; 1999.
[47] Wolter F-E, Friese K-I. Local & global geometric methods for analysis interrogation, reconstruction, modification & design of shape. In: Proceedings of computer graphics international, 2000. Geneva (Switzerland): IEEE Computer Society; 2000.