Computer Graphics, Vision and Mathematics

in co-operation with

EUROGRAPHICS

GraVisMa 2009

Abstract Proceedings

University of West Bohemia
Plzen
Czech Republic

September 2-5, 2009

Co-Chairs
Dietmar Hildenbrand, University of Darmstadt, Germany
Vaclav Skala, University of West Bohemia, Czech Republic

Edited by
Dietmar Hildenbrand
Vaclav Skala

GraVisMa 2009 Proceedings

Editor-in-Chief: prof.Ing.Vaclav Skala, CSc.
c/o University of West Bohemia, Univerzitni 8, Box 314
CZ 306 14 Plzen
Czech Republic
skala@kiv.zcu.cz

Managing Editor: Vaclav Skala

Author Service Department & Distribution:
Vaclav Skala - UNION Agency
Na Mazinách 9
CZ 322 00 Plzen
Czech Republic

Printed: University of West Bohemia, Plzeň, Czech Republic

Hardcopy: *ISBN 978-80-86943-92-3*

CONTENTS

Raytracing Point Clouds using Geometric Algebra

Crispin Deul[1] Michael Burger[1] Dietmar Hildenbrand[1] Andreas Koch[2]

[1] Interactive Graphics Systems Group
Computer Science Department
TU Darmstadt, Germany
dietmar.hildenbrand@gris.informatik.tu-darmstadt.de

[2] Embedded Systems and Applications
Computer Science Department
TU Darmstadt, Germany
koch@esa.informatik.tu-darmstadt.de

ABSTRACT

Geometric Algebra (GA) supports the geometrically intuitive development of an algorithm with its build-in geometric primitives such as points, lines, spheres or planes, but on the negative side GA has a huge computational footprint. In this paper we study how GA can compete with traditional methods from Linear Algebra (LA) in the field of raytracing. We examine the raytracing algorithm for both GA and LA on the basis of primitive operations. Furthermore we benchmark implementations of both algebras with a framework for rendering point clouds based on GA. Our results show that depending on the microprocessor architecture like CPUs, FPGAs or GPUs Geometric Algebra and Linear Algebra can raytrace with comparable speed.

Keywords: FPGA, Geometric Algebra, GPGPU, Point Cloud, Raytracing.

1 EXTENDED ABSTRACT

In this paper we compare the speed of the raytracing algorithm in Conformal Geometric Algebra (CGA) and Linear Algebra (LA). We use two approaches to get an answer whether raytracing with CGA can compete with LA. First the computations in CGA and LA are analyzed on the level of primitive operations with scalars like multiplications and additions. Important parts of the rendering are the intersection of the viewing ray with the objects of the scene, the calculation of the surface normal at the intersection point and reflections, which are needed for lighting and shadow computations. Furthermore we analyze how to parallelize the computations to achieve a speed-up on parallel hardware architectures. Raytracing in CGA on CPUs has already been investigated [1]. There are two starting points for our investigations. The first one is the property of the raytracing algorithms to compute the colour of each pixel independently from all the other ones in the actual frame. The second one is that objects in CGA consist of multivectors with 32 entries which can be computed independent of each other.

Using Gaalop [3] we found two possibilities of optimizing the calculations listed above. On the one hand, the number of operations is minimized and costly operations, such as square roots or divisions, are avoided whenever possible. On the other hand, we aim to decompose complex computations into simpler steps that can be executed in parallel or serially. Parallel calculations can be processed efficiently on SIMD-architectures like GPUs or appropriately configured FPGAs. An efficient implementation of serial steps is possible through using a pipeline architecture on a FPGA.

Beyond our work on the point cloud model we present a novel framework to raytrace point clouds using Conformal Geometric Algebra (CGA). The surface represented by a point cloud is locally approximated by splats consisting of a fitting and a bounding object. To compute the fitting object we fit planes or spheres into the k-neighbourhood of a chosen fitting point. Prior work uses a sphere fit into a point cloud to approximate the surface of the cloud. But instead of using the sphere only in an intermediate step [2] we use the sphere directly to represent the surface. Using the $k+1$ points of the neighbourhood we have to solve the eigenvector problem of a 5×5 matrix. Taking the eigenvector corresponding to the smallest positive eigenvalue we get the coefficients of our fitting object. The advantage of this approach compared to linear algebra is that, during the algorithm, we do not have to consider whether the outcome of the fitting process will be a sphere or a plane (all information is kept in the resulting eigenvector).

We use the framework to benchmark CGA and LA implementations on different hardware architectures like the CPU and GPU to prove our theoratical considerations on the primitive operations in a real world example.

Our Results show that although CGA needs more primitive operations than LA in general, CGA and LA can be comparable in speed depending on the microprocessor architecture. This is due to the parallel nature of the CGA multivector computations.

REFERENCES

[1] L. Dorst, D. Fontijne, and S. Mann. *Geometric Algebra for Computer Science, An Object-Oriented Approach to Geometry*. Morgan Kaufman, 2007.

[2] Gaël Guennebaud and Markus Gross. Algebraic point set surfaces. In *In Proceedings SIGGRAPH '07*, 2007.

[3] D. Hildenbrand and Joachim Pitt. The Gaalop home page. HTML document http://www.gaalop.de, 2008.

A Study of 3-D Surface Registration Using Distance Map and 3-D Radon Transform

Makoto Hasegawa
Faculty of Engineering, Kinki University
1 Takayaumenobe, Higashihiroshima, Hiroshima, 739-2116 Japan
hasegawa@hiro.kindai.ac.jp

ABSTRACT
A three-dimensional surface model registration method using distance map and three-dimensional Radon transform is proposed. Phase-only matched filter can be applied into our three-dimensional surface model registration just like conventional two-dimensional image registration. Our registration procedure is described and executes simulations.

Keywords
Surface registration, distance field map, Radon transform, R-transform, phase-only matched filter.

1. INTRODUCTION
Three-dimensional surface model registration is an important and fundamental topic in computer graphics. Registration of two surface models means matching between a template surface model and a target surface model described many polygons and vertices. Various methods are proposed. Chen and Medioni propose the iterative closest point method, which displace the surface model by the minimizing the difference of the corresponded vertices [Che 91a]. We discuss a new method using distance map, and three-dimensional Radon transform. Phase-only matched filter can be applied into our three-dimensional surface model registration just like conventional two-dimensional image registration.

2. REGISTRATION PROCEDURE
[Step 1] Create distance maps (Fig. 2) of the two surface models (Fig. 1). The distance map is defined as each grid in the 3-D space is attached a value which is inversely proportional to the distance the grid and the closest surface model vertex.

[Step 2] Convert the distance map using three-dimensional Radon transform. We obtain a Radon image $T_f(\rho,\phi,\theta)$ of the distance map f.

[Step 3] Convert $T_f(\rho,\phi,\theta)$ using R-transform as

$$R_f(\phi,\theta) = \int T^2{}_f(\rho,\phi,\theta)d\rho. \qquad (1)$$

The R-transform image is shown by Fig. 3 [Tab06a].

[Step 4] Rotate the surface models so that these two R-transform images may become equal.

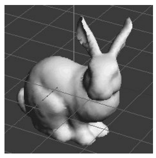


Fig. 1: Surface model

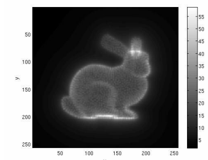


Fig. 2: Distance map (x-y projection)

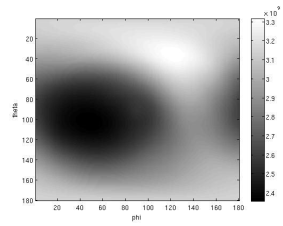


Fig. 3: Three-dimensional R-transform

[Step 5] Create x-y and y-z projection images from the distance maps. We can detect the translation position (x, y, z) between template and target surface model using phase-only matched filter.

ACKNOWLEDGEMENTS

This research was supported by Takahashi Industrial and Economic Research Foundation.

REFERENCES

[Che91a] Y. Chen and G. Medioni, “Object modeling by registration of multiple range images,” *IEEE Int. Conference on Robotics and Automation*, vol. 3, pp. 2724-2729, 1991.

[Tab06a] S. Tabbone, L. Wendling, and J. –P. Salmon, “A new shape descriptor defined the Radon transform,” *Computer Vision and Image understanding*, 102, pp.42-51, 2006.

Virtual Reconstruction of 3D Scenes Based on Range Cameras

Alexander Sabov
Fraunhofer IPK
Pascalstrasse 8-9
10587 Berlin, Germany
alexander.sabov@ipk.fraunhofer.de

Jörg Krüger
Fraunhofer IPK
Pascalstrasse 8-9
10587 Berlin, Germany
joerg.krueger@ipk.fraunhofer.de

Keywords
Virtual reconstruction, range imaging, image registration, time-of-flight

EXTENDED ABSTRACT

1. INTRODUCTION

Virtual reconstruction of existing objects and environments is needed in many different areas such as the planning of building extensions, quality assurance for production processes and city reconstructions for geographic information systems. Different scanner systems have been established which meet the demands on the application. For applications such as computer aided facility management a short scanning time is the most important factor instead of high precision. Range cameras are a new generation of 3D scanners which can measure distances to their environment directly with a high frame rate. In this paper a reconstruction method is presented which is specific to range camera data.

2. RECONSTRUCTION

Features are identified in the intensity images of the camera with the Sobel edge detector [Xia08a]. 3D coordinates of corresponding surface elements (surfels) are calculated from the depth map. The feature tracking and registration is based on the iterative closest point method [Rus01a]. The nearest neighbour surfel between two frames is the first assumption for the new surfel position. An iterative process calculates the corresponding transformation matrix and determines the new neighbours for the transformed data. During the registration, a feature map and a surfel map are continuously constructed and updated. For performance improvement, a registration between two frames is done first and provides an initial position to register the features and surfels in the map. Fig.1 shows the surfel map for a room in an office building.

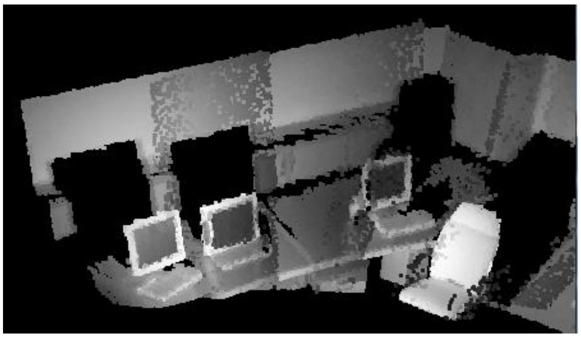


Figure 1. Reconstructed office

3. EVALUATION AND CONCLUSION

For the evaluation of the reconstruction method the wall thickness of the reconstructed office is measured. Ideally, all wall surfels are a thin plane, but due to imprecise distance data they expand in depth. The wall thickness is 6-10 cm in a distance of 4-5 meters. This value is high compared to other sensors like 3D laser scanners, but a scanning time of 10 seconds for a panorama compared to 15 minutes using a 3D laser scanner makes the range scanner and the reconstruction method suitable for applications where a short scanning time is of highest priority.

4. REFERENCES

[Xia08a] Xi'ang, Liu, et al. Cocoon edge detection based on self-adaptive Canny operator. Computer Science and Software Engineering, 2008.

[Rus01a] Rusinkiewicz, S., and Levoy, M. Efficient variants of the ICP algorithm. 3D Digital Imaging and Modeling, 2001.

Using Geometric Algebra for Algorithms Calculating Convex Hull and Object Representation in 3D

Florian Wörsdörfer
TU Darmstadt (Germany), Computer Science Department
fw1001@rbg.informatik.tu-darmstadt.de

Dietmar Hildenbrand
TU Darmstadt (Germany), Computer Science Department
dhilden@gris.tu-darmstadt.de

Eduardo Bayro-Corrochano
CINVESTAV Guadalajara (Mexico)
edb@gdl.cinvestav.mx

Andreas Koch
TU Darmstadt (Germany), Computer Science Department
koch@esa.informatik.tu-darmstadt.de

ABSTRACT

In this paper we present the use of geometric algebra for computations of the complex hull and the object representation of unstructured points in 3D. We especially examine the performance of different implementations and the advantages of symbolic optimizations by Gaalop for modern parallel platforms using GPGPU. The first algorithm described here calculates the convex hull of a set of unstructured points. The second one calculates its alpha shape using Delaunay triangulation. Because of their very geometric nature both algorithms take advantage of a description in geometric algebra. Not only are they easier to understand this way but also they can be optimized for the use with parallel platforms automatically.

Keywords: Geometric Algebra, Convex Hull, Object Representation, CUDA.

1 EXTENDED ABSTRACT

The problem of reconstructing the three-dimensional shape of an object has been discussed using a variety of different approaches and heuristics. Our goal is to show the benefits this topic can gain from using conformal geometric algebra . Therefor we use a combination of well known techniques to compute the desired results and concentrate our work on using conformal geometric algebra where it is appropriate. We examine the advantages of geometric algebra over the more common vector algebra or linear algebra. One of these advantages is the ability to formulate our algorithms in a crisp and elegant way because most of our operations require manipulations of basic geometric entities represented by simple algebraic objects. Another advantage is the ability to generate high efficient source code for parallel platforms using the software Gaalop [4] and its capability of performing symbolic optimizations.

The first algorithm described here computes the convex hull from a set of unstructured points and consists of three parts. First redundant points are eliminated by dividing the *xy* plane into a regular grid and discarding unnecessary points while walking that grid. Then the remaining points are clustered using the k-means++ algorithm [1] and finally the complex hull is generated from these clusters by the quickhull algorithm [2].

The second algorithm computes the object representation from a set of unstructured points by performing two different steps. First the Delaunay triangulation [6] of the points is computed. We start with an initial tetrahedron and incrementally add new points by building more tetrahedrons. Each one of them is checked for the empty sphere criterion. If this criterion is violated the corresponding tetrahedrons are changed by doing either a triangle flip or an edge flip. Then in the second step from that Delaunay triangulation the alpha shape [3] is calculated as the final object representation.

After developing the above algorithms in the framework of conformal geometric algebra using CLUCalc [7] as an rapid prototyping tool we perform symbolic optimizations with Gaalop to generate efficient C code automatically. From this C sources a CUDA implementation is derived. Additionally we will give a C implementation of the algorithms using standard vector algebra. Also we will refer to Gaigen 2 implementations given in [5]. These different implementations are compared with respect to the achievable improvements in performance.

REFERENCES

[1] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[2] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22:469–483, 1995.

[3] Herbert Edelsbrunner and Ernst P. Mücke. Three-dimensional alpha shapes. *ACM Transactions on Graphics*, 13(1):43–72, 1994.

[4] D. Hildenbrand and Joachim Pitt. The Gaalop home page. HTML document http://www.gaalop.de, 2008.

[5] L. Dorst and D. Fontijne and S. Mann. *Geometric Algebra for Computer Science, An Object-Oriented Approach to Geometry*. Morgan Kaufman, 2007.

[6] E. P. Mücke. A robust implementation for three-dimensional delaunay triangulations. In *Proceedings of the 1st International Computational Geometry Software Workshop*, pages 70–73, 1995. To appear in the *International Journal of Computational Geometry & Applications*.

[7] C. Perwass. The CLU home page. HTML document http://www.clucalc.info, 2009.

Conformal Geometric Algebra by Extended Vahlen Matrices

Leo Dorst

University of Amsterdam, The Netherlands, L.Dorst@uva.nl

ABSTRACT

The conformal transformations in $\mathbb{R}^n$ (those that preserve angles) can be modeled conveniently as orthogonal transformations of the Minkowski space $\mathbb{R}^{n+1,1}$ [1], and these in turn are representable as versors in geometric algebra (i.e., as geometric products of invertible vectors) [3]. Using anti-symmetric combinations of geometric products, an outer product can be introduced as the foundation for the Grassmann algebra of $\mathbb{R}^{n+1,1}$. The outer product of (not necessarily invertible) vectors spans subspaces of $\mathbb{R}^{n+1,1}$; these 'blades' can be identified with circles, spheres, planes and tangents (and more) in the space $\mathbb{R}^n$. The versors act on them in a structure-preserving manner.

Thus we get universal conformal operators, simplifying software, and as a consequence this *conformal model* (a.k.a. CGA, conformal geometric algebra [4]) is beginning to be useful in computer science fields like graphics, vision and robotics. In such fields, the main interest is often in Euclidean similarities or rigid body motions, rather than the general conformal mappings of which they are a special case.

On the other hand, conformal transformations have been studied in mathematics using Vahlen matrices, as a homogeneous representation of Möbius transformations. Though originally defined for the complex plane, the Vahlen matrices have been extended to allow coefficients from a Clifford algebra to permit the study of the group of n-dimensional conformal transformations [5].

In this paper, the two ideas meet, to lay the foundation for a fruitful interaction. To embed the full conformal model we need to define the matrices that represent the blades in geometric algebra – this gives us an extension of the matrix representation from representing only transformations to being interpretable as geometric primitives. This is done by translating the outer product and contraction product constructions of such geometrical elements into matrix operations, through the intermediate step of writing them as linear combinations of geometric products. Moreover, the vector differentiation operation can be cast as an operator matrix. We provide a useful dictionary for this translation between the elements of the conformal model and their matrix representation.

Using the dictionary, we show some examples of how the two representations can interact to study properties of conformal transformations, with some emphasis on Euclidean similarities and motions.

- We derive *Chasles' theorem* in a compact manner: first we focus on the general motion that preserves the point at infinity; using the explicit representation of a line in matrix form from the dictionary, we determine its *eigenlines*. We then factor the original motion quantitatively in two terms, preserving the eigenline point-wise (a rotation around the line) and as a set (a translation along the line).
- We use Chasles' theorem to give a closed form *principal logarithm of a Euclidean motion*, in its Vahlen matrix representation, using the commutation properties of the above factorization.
- We compute a table of *commutation laws* between the four fundamental conformal motions: translation, rotation, uniform scaling and transversion and their associated parameter changes. Especially the commutation between translation and transversion is of interest (and may even be new).
- Using the basics presented here, we have been able to give a closed-form solution to the determination of a conformal transformation, based on exact transformation data of points and frames go [2].

We are beginning to find these new tools indispensable in solving such advanced problems. Although they are in principle fully equivalent to writing things out algebraically, computations are more automatic in the matrix structure – which takes care of the additional e_o, e_∞ and $e_o \wedge e_\infty$ bookkeeping at the heart of the conformal model, and leaves only straightforward n-dimensional Euclidean geometric algebra computations.

Keywords: conformal transformations, rigid body motions, geometric algebra, Vahlen matrix, Chasles' theorem, logarithm, commutation, transversion.

REFERENCES

[1] P. Anglès, *Conformal Groups in Geometry and Spin Structures*, PMP 50, Birkhäuser, 2008.

[2] C. Cibura, L. Dorst, *From Exact Correspondence Data to Conformal Transformations in Closed Form Using Vahlen Matrices*, GRAVISMA 2009.

[3] L. Dorst, D. Fontijne, S. Mann, *Geometric Algebra for Computer Science*, Morgan Kaufmann, 2007.

[4] D. Hestenes, A. Rockwood, H. Li, U.S. Patent 6,853,964, granted February 8, 2005

[5] P. Lounesto, *Clifford Algebras and Spinors*, LMSLNS 286, Cambridge University Press, 2001.

Motion Tracking with Geometric Algebra-valued Particle Filter

Kanta Tachibana

Faculty of Informatics,
Kogakuin University
1-24-2 Nishi-Shinjuku,
163-8677, Shinjuku-ku, Tokyo

kanta@cc.kogakuin.ac.jp

ABSTRACT

Choice of latent variables and likelihood function are important for three-dimensional time-series tracking. I propose motion tracking with high order geometric algebra-valued particle filter. The proposed method shows significant improvement compared to tracking with conventional Euler angles.

Keywords

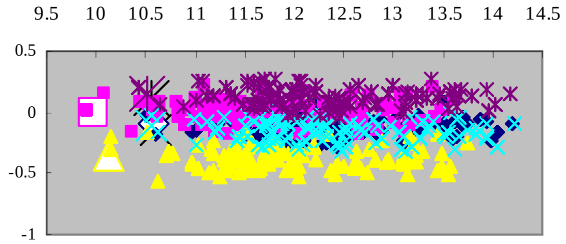
Geometric Algebra, Motion Tracking, Computer Vision

1. INTRODUCTION

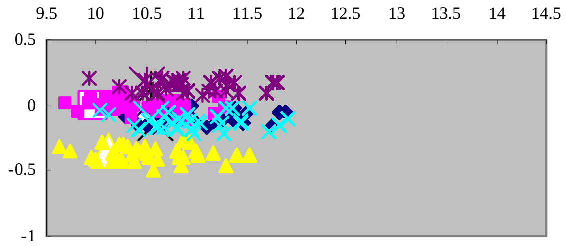
Inference of three-dimensional motion is an important task for computer vision. In this study, I discuss inference of rigid body motion when some feature points are observed by a monocular camera. I assume that feature points are observed at every timestep without occlusion but with measurement noise. For this task, I utilize particle filter (PF), a sequential Bayesian method. For PF to be applied to geometric problems, it is important to use appropriate latent variables. Some studies [e.g. Mar01] utilize quaternion instead of conventional Euler angels as latent variables for time-series three-dimensional inference. However, quaternion's effect has not been clarified quantitatively yet. I furthermore introduce a new resampling step, in which hypotheses are evaluated, for particle filter. In the new resampling step, likelihood of hypothesis is calculated using not only single points but also circles each of which is a combination of three points. The aim of this study is to clarify effects of 1) representations of rotation with rotor components to such geometric inference and of 2) introducing high-order entities to resampling step.

2. NUMERICAL EXPERIMENT

Five feature points fixed at a transparent rigid body, which moves smoothly in 3D space, are assumed to be observed through 100 timesteps. PFs track these feature points. One PF uses Euler angles as its latent variables and the other uses rotor components. The latter evaluates likelihood of each hypothesis under distance from hypothetical circles as well as hypothetical points. Figure 1 shows result of Euler angle PF (a) and that of my proposal (b). Inference of depth (shown as horizontal axis) improved. Average errors were (a) 0.89 (+/- 0.16) vs. (b) 0.59 (+/- 0.05).



(a) Euler angles



(b) Rotor components

Figure 1. True (big marks) and inferred positions (small marks) projected orthogonally into Depth-Height plane.

3. REFERENCES

[Mar01] Marins, J. L. et al. An extended Kalman filter for quaternion-based orientation estimation using MARG sensors. In Proc. IEEE-RSJ Intl Conf. on Intelligent Robots and Syst., pp.2003-2011, 2001

Pose estimation based on Geometric Algebra

Yan Cui
DFKI
Trippstadter Strasse. 122
67663 Kaiserslautern Germany
cuiyan119119@gmail.com

Dr. Dietmar Hildenbrand
TU Darmstadt
Hochschulstrasse. 10
64283 Darmstadt Germany
Hildenbrand@gris.informatik.tu-darmstadt.de

ABSTRACT

2D-3D pose estimation is an important task for computer vision, ranging from robot navigation to medical intervention. In such applications as robot guidance, the estimation procedure should be fast and automatic, but in industrial metrology applications, the precision is typically a more important factor. In this paper, a new 3D approach for infrared data visualization precisely with the help of 2D-3D pose estimation is proposed. The approach provides a user friendly interface, a flexible structure and a precise result, which can be adjusted to almost all the geometrically complex objects.

Keywords: Geometric algebra, 2D-3D pose estimation, ICP algorithm.

EXTENDED ABSTRACT

The main goal of pose estimation is to estimate the relative position and orientation of a known 3D object with respect to a reference camera system. In other words, we search for a transformation (i.e. the pose) of the 3D object such that the transformed object corresponds to 2D image data. Pose estimation is a subclass of the more general problem of registration which is one of the key problems in computer vision. In our scenario, there is a 3D object, which can contain different entities like 3D points, 3D lines, 3D spheres, 3D circles, kinematic chain segments, boundary contours or contour parts. The 3D object in an image of a calibrated camera is observed. We can estimate the relative position and orientation of the 3D object to the reference camera system with 2D-3D pose estimation method based on Geometric Algebra [1].

In this paper we give two methods to solve the pose estimation by using features of the object (i.e. corners, edges) and by using free-form contour model. In the first method, the user should select corresponding features (i.e. corners, edges) between 3D object and 2D image manually. Then with the equation $\lambda\left(\left(M\underline{X}\widetilde{M}\right)\underline{\times} e \wedge (O \wedge x)\right) \cdot e_+$ in Conformal Geometric Algebra define the distance between 3D object point and the ray from the camera position to the 2D image point, where $\underline{X}$ is the 3D object point in conformal space, M stands for the rigid motion of the 3D object, O is the camera position, x is the 2D image point. Construct a linear system to minimize the energy function that sum of the distances, afterward, we can get the rigid body motion parameter M. The final result will be affected by the features that user selected. We will improve the unstable result by method two. Firstly user can extract the silhouette of the 3D object in the image with Chan and Vese model [2]. In the same time the algorithm can find the free-form contour automatically by Fourier descriptors, which enable a low-pass description of the 3D object contour, and is advantageously applied to the pose problem. In order to relate 2D image information to 3D entities, we interpret a point on the 2D silhouette as a projection ray from the camera position in 3D space, construct a similar cone shape. The aim is to find a best fit with the cone shape. We use the Iterative Closest Point (ICP) algorithm [3] to solve the 2D to 3D registration problem. In order to avoid the local minimum, we try to improve the ICP algorithm, define the error a sum of the distances between the posed object points and the nearest rays. If the error is bigger than a suitable threshold, we should rotate the 2D image some degree around the camera view direction, until the error is smaller than the threshold. After finding the right posed position of the 3D object, we try to map the texture of the 2D image to the 3D mode, test the triangles of the 3D object are visible or not with ray-tracing algorithm. In this paper we try to implement the ray-tracing method based on the Geometric Algebra [4].

With the framework above, the developed approach can deal with image time sequences from different view points captured by the camera. The final visual results are high-precision and satisfied the requires of industrial metrology application.

REFERENCES

[1] *Rosenhahn B. and Sommer G. Pose Estimation in Conformal Geometric Algebra Part I: The stratification of mathematical spaces. Part II: Real-Time pose estimation using extended feature concepts. Technical Report 0206, University Kiel, 2002.*

[2] *T. Chan and L. Vese. Active contours without edges. IEEE Transations on Image Processing,10(2):266-277, 2001.*

[3] *Rusinkiewicz S. and Levoy M. Efficient variants of the ICP algorithm. Available at http://www.cs.princeton.edu/smr/papers/fasticp/. Presented at Third International Conference on 3D Digital Imaging and Modeling (3DIM), 2001.*

[4] *Leo Dorst. Geometric Algebra for Computer Science An Object -Oriented Approach to Geometry, Chapter 23.*

Spherical Offset

Seok Won Lee*, Andreas Nestler

Institute of Forming and Cutting Manufacturing Technology
Dresden University of Technology, D-01062, Dresden, Germany
{swlee,nestler}@mciron.mw.tu-dresden.de

Keywords

offset, geodesic offset, spherical geometry, visibility, visibility cone, tool path generation, multi axis milling, collision-free tool movement.

EXTENDED ABSTRACT

1. INTRODUCTION

Numerically controlled (NC) milling is one of the most prevailing manufacturing processes. During milling the blank material is removed by geometry-defined cutters moving along given tool paths, so that the net shape of the target part is achieved in the end. In this paper, spherical offset algorithm is proposed which is applicable to generate tool path and to avoid collision between the cutting tool and the material.

2. SPHERICAL VISIBILITY MAP

By definition, visibility is a synonym of a determination of the visible parts/polygons/lines of the scene from the viewpoint P. It is, however, diversely interpreted in various contexts such as shadow computation and hidden surface removal in computer graphics, finding accessible direction of a probe in metrology, scanning strategy in reverse engineering, motion planning in robotics, searching parting line in molds and dies and so on. Especially in CAD/CAM field visibility problem arises either in finding an accessible area of the tool with respect to a given part contour or in generating interference-free tool paths for 5 axis machining.

Spherical visibility map projects a visible scene from the viewpoint in Euclidean space into the sphere S^2, of center which is imaginarily located on the viewpoint. An intersection of the ray starting from the center with the scene is mapped onto S^2. Then the boundary curve builds the visibility cone.

Since a tool has a non-zero radius the visibility cone is needed to be offset inwards with a constant angular radius ρ on S^2 to avoid gouges while a tool approaches the position P. The deepest point Q of the spherical offset is regarded as the most reliable tool access direction. To confirm that Q is the safe approach direction, the collision is checked between the tool and the local area around P finally.

3. SPHERICAL OFFSET AND ITS APPLICATION

By definition, the offset is the locus of the points which are displaced with constant distance (or offset radius) from the progenitor. Depending on the surface where the progenitor lies it is classified into plane offset and geodesic offset. The spherical offset belongs to the geodesic offset of which distance is measured along great circle on spherical surface S^2.

At the beginning we define the spherical geometry including point, circle, segment and polygon etc on S^2 and its elementary geometric operations which are, for example, the intersection test, orientation test, tangency test etc.

Next we introduce the algorithm to calculate the offset on S^2 using piecewise spherical segment (PSS) method. The polygonal mesh in Euclidean space, which is composed of line segments, is projected into S^2 and each line segment is mapped as a segment of great circle on S^2. From the soup of segments the visibility cone is calculated while the intersection points between segments are found by adapted sweep line algorithm for the sphere and invalid segments are rejected. Then the irrelevant spherical segments to the offset building are discarded prior to true offset calculation so that the most numerical inconsistency could be prevented previously.

Finally we take several implementation examples in CAD/CAM applications taking advantage of the spherical offset. First, the spherical offset could be used to calculate a collision-free approach direction of the tool taking the workpiece contour into consideration. The visibility cone of the workpiece is calculated from a viewpoint. The offset from visibility cone is built by the proposed spherical offset algorithm. The deepest point of the offset, say, the offset point with the largest angular offset radius ρ_{max}, is regarded as the most reliable tool access direction. Another example using spherical offset could be found in the tool path generation to groove characters or curves on the spherical surface. The cutting tool follows along the tool paths which are comprised of the path of spherical offset itself and some linking paths so as that the contour of the world map, for instance, could be grooved on the spherical workpiece.

Inertial Navigation using Geometric Algebra

Liam Candy
Department of Engineering
University of Cambridge, CB2 1PZ, UK
and CSIR, Pretoria, South Africa
lpc28@cam.ac.uk

Joan Lasenby
Department of Engineering
University of Cambridge, CB2 1PZ, UK
jl221@cam.ac.uk

Keywords: Inertial Navigation, Bortz Equation, Geometric Algebra, Conformal Algebra.

EXTENDED ABSTRACT

1 INTRODUCTION

In strapdown inertial navigation systems (SDINS) the angular velocity measured in a body frame is used to update a rotation generator that relates the orientation of the body frame to some reference frame. Because the angular velocity measurements are made in the body frame, which changes its orientation relative to the reference frame, direct integration of the conventional direction cosine differential equation is an unsuitable method for tracking attitude. Bortz [Bor71] derived a method of accounting for this effect by representing the actual finite rotation by an orientation vector $\phi(t)$ and obtaining an expression for $\dot{\phi}(t)$. Bortz showed that $\dot{\phi}$ could be split into two parts, $\omega(t)$ and $\dot{\sigma}(t)$. $\omega(t)$ is the angular rate vector and $\dot{\sigma}(t)$ is a term arising due to the change in the body coordinate system.

$\dot{\phi}$ in terms of ϕ and ω is known as the *Bortz Equation* – updating the quantity ϕ given the measurements and then extracting the attitude, is known to be more accurate than direct integration of the dynamical equations. In this paper we will use *geometric algebra (GA)* to show how the Bortz equation arises almost trivially from the dynamical equations.

Recent contributions in the SDINS literature have used dual quaternions, a concise representation for both rotations and translations, to extend the Bortz equation to the more general case of simultaneously updating both attitude and position [YWL05]. The claim is that there is an equivalent *Bortz* equation which allows more accurate updating. Here we use the *conformal geometric algebra* (CGA) [HS84] to investigate what a *conformal Bortz equation* would look like and whether the claims made for dual quaternions are valid.

The 3D and 5D Bortz equations

If the angular velocity of the body wrt the reference frame is written as a bivector in the body frame, Ω^b, and the rotor which takes the reference frame to the body frame at time t as $R(t)$, the dynamical equation is $\dot{R} = -\frac{1}{2}R\Omega^b$

We define Φ as $\Phi = \alpha B$, where B is a unit bivector and $R = \mathrm{e}^{-\Phi/2}$, ie a rotation of α in the plane of B. Since $\dot{\Phi} = \alpha\dot{B} + \dot{\alpha}B$, and using expressions for $\dot{\alpha}$ and $\dot{B}$ obtained by equating scalar and bivector parts of the dynamical equation, we are able to write $\dot{\Phi}$ as a bivector equation (where Ω^b is written as Ω for conciseness):

$$\dot{\Phi} = \Omega + \left(\frac{|\Phi|}{2}\cot\frac{|\Phi|}{2} - 1\right)\left[\Omega + \frac{(\Omega\cdot\Phi)\Phi}{|\Phi|^2}\right] - \frac{\langle\Phi\Omega\rangle_2}{2} \tag{1}$$

This is effectively the dual of the conventional Bortz equation.

Using the CGA framework we know that we can express both rotations and translations as rotors – thus we are able to define a new Φ in our 5D space as $\Phi = \alpha B + tn$, where αB gives the spatial rotation as before and t is related to the spatial translation. It is not hard to show that $R = \mathrm{e}^{-\Phi/2}$ represents a rotation of α in the plane of B followed by a translation of s which is a function of both t and the spatial rotation. We then also show that the equivalent 5D angular velocity bivector, Ω is given by $\Omega = \Omega_3 + n\tilde{R}_\alpha \dot{s} R_\alpha$, where Ω_3 is the 3D angular velocity bivector.

The 5D dynamical equation is again $\dot{R} = -\frac{1}{2}R\Omega$. Equating scalar and spatial bivector parts gives expressions for $\dot{\alpha}$ and $\dot{B}$ (indeed those obtained in the 3D case). Equating non-spatial bivector and 4-vector parts simply gives identities, thus confirming consistency.

We then form $\dot{\Phi} = \dot{\alpha}B + \alpha\dot{B} + \dot{t}n$ in terms of Φ and Ω. However, the resulting equation is not as simple in form as the 3D equation, ie it is not simply a matter of replacing ϕ and ω with their 5D counterparts. This therefore calls into question the process of substituting a dual quaternion generator into the Bortz equation form and updating the dual quaternion components. The paper will discuss the differences and investigate the discrepancies via simulations.

REFERENCES

[Bor71] J.E Bortz. A new mathematical formulation for strapdown inertial navigation. *IEEE Transactions on Aerospace and Electronic Systems*, AES-7, no. 1:61–66, 1971.

[HS84] D Hestenes and G Sobczyk. *Clifford Algebra to Geometric Calculus: A unified language for mathematics and physics*. 1984.

[YWL05] D. Hu T. Li Y. Wu, X. Hu and J. Lian. Strapdown inertial navigation system algorithms based on dual quaternions. *IEEE Transactions on Aerospace and Electronic Systems*, 41-4:110–132, 2005.

Duality and Computation in Projective Space

Vaclav Skala

Department of Computer Science and Engineering
Faculty of Applied Sciences, University of West Bohemia
Univerzitni 8, CZ 306 14 Plzen, Czech Republic
http://Graphics.zcu.cz

ABSTRACT

This paper presents solutions of some selected problems that can be easily solved in the projective space. If the principle of duality is used, quite surprising solutions can be found and new useful theorems can be generated as well. Principle of duality can be used to derive an equation of a parametric line in E3 as an intersection of two planes. This new formulation avoids division operations and increases the robustness of computation.

Keywords

Projective space, homogeneous coordinated, duality principle, intersection computation

1. PROJECTIVE GEOMETRY

The homogeneous coordinates are mostly introduced with geometric transformations concepts and used for the projective space representation. Nevertheless, geometrical interpretation is missing in nearly all publications. Conversion from the homogeneous coordinates to the Euclidean coordinates is defined for E^2 case as:

$$X = x / w \qquad Y = y / w \tag{1}$$

where: $w \neq 0$, point $\boldsymbol{x} = [x, y, w]^T$ and $\boldsymbol{x} \in P^2$, $\boldsymbol{X} = [X, Y]^T$ and $\boldsymbol{X} \in E^2$, if $w = 0$ then $\boldsymbol{x}$ represents "an ideal point", that is a point in infinity.

The cross-product of two vectors $\boldsymbol{x}_1$, $\boldsymbol{x}_2 \in E^2$, if given in the homogeneous coordinates, is defined as

$$\mathbf{x}_1 \times \mathbf{x}_2 = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ x_1 & y_1 & w_1 \\ x_2 & y_2 & w_2 \end{vmatrix} \tag{2}$$

The cross-product of three vectors $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$ is defined in the homogeneous coordinates as:

$$\boldsymbol{x}_1 \times \boldsymbol{x}_2 \times \boldsymbol{x}_3 = \begin{vmatrix} \boldsymbol{i} & \boldsymbol{j} & \boldsymbol{k} & \boldsymbol{l} \\ x_1 & y_1 & z_1 & w_1 \\ x_2 & y_2 & z_2 & w_2 \\ x_3 & y_3 & z_3 & w_3 \end{vmatrix} \tag{3}$$

Let three points $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{x}_3$ be given in the projective space. Then a plane $\boldsymbol{\rho} \in E^3$ defined by those three points is determined as:

$$\boldsymbol{\rho} = \boldsymbol{x}_1 \times \boldsymbol{x}_2 \times \boldsymbol{x}_3 \tag{4}$$

As a point is dual to a plane, a plane is dual to a point in E^3 we can use the principle of duality directly, now.

Let three planes $\boldsymbol{\rho}_1$, $\boldsymbol{\rho}_2$ and $\boldsymbol{\rho}_3$ be given in the projective space. Then a point $\boldsymbol{x}$, which is defined as the intersection point of those three planes, is determined as:

$$\boldsymbol{x} = \boldsymbol{\rho}_1 \times \boldsymbol{\rho}_2 \times \boldsymbol{\rho}_3 \tag{5}$$

The computations can be extended to the projective space as follows:

$$w_1 w_2 (\mathbf{X}_1 \times \mathbf{X}_2) = \mathbf{x}_1 \times \mathbf{x}_2 = w_1 w_2 \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ X_1 & Y_1 & 1 \\ X_2 & Y_2 & 1 \end{vmatrix} \tag{6}$$

Let $\boldsymbol{X}_2 - \boldsymbol{X}_1 = \boldsymbol{\omega}$ and $\boldsymbol{X}_1 \times \boldsymbol{X}_2 = \boldsymbol{v}$. A point on the line $\boldsymbol{q}(t) = \boldsymbol{q}_1 + \boldsymbol{\omega}\, t$ is defined as:

$$\boldsymbol{q}(t) = \frac{\boldsymbol{v} \times \boldsymbol{\omega}}{\|\boldsymbol{\omega}\|^2} + \boldsymbol{\omega} t \qquad \hat{\boldsymbol{q}}(t) = \begin{bmatrix} \boldsymbol{v} \times \boldsymbol{\omega} + t\boldsymbol{\omega}\|\boldsymbol{\omega}\|^2 \\ \|\boldsymbol{\omega}\|^2 \end{bmatrix} \tag{7), (8}$$

The Eq.7 defines a line $\boldsymbol{q}(t)$ in the E^3 by two points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ given in the homogeneous coordinates. Of course, we can avoid the division operation easily using homogeneous notation for a scalar value $\hat{\boldsymbol{q}}(t)$, Eq.8 and the resulting line is defined directly in the projective space P^3.

Let us imagine that we have to solve the second problem, i.e. a line defined as an intersection of two given planes $\boldsymbol{\rho}_1$ and $\boldsymbol{\rho}_2$ in the Euclidean space:

$$\boldsymbol{\rho}_1 = [a_1, b_1, c_1, d_1]^T \qquad \boldsymbol{\rho}_2 = [a_2, b_2, c_2, d_2]^T \tag{9}$$

Now we can apply the principle of duality as we can interchange the terms "point" and "plane" and exchange $\boldsymbol{v}$ and $\boldsymbol{\omega}$ in the Eq.7.8 and we get:

$$\boldsymbol{q}(t) = \frac{\boldsymbol{\omega} \times \boldsymbol{v}}{\|\boldsymbol{v}\|^2} + \boldsymbol{v} t \qquad \hat{\boldsymbol{q}}(t) = \begin{bmatrix} \boldsymbol{\omega} \times \boldsymbol{v} + t\boldsymbol{v}\|\boldsymbol{v}\|^2 \\ \|\boldsymbol{v}\|^2 \end{bmatrix} \tag{10), (11}$$

If $\|\boldsymbol{v}\| = 0$ then the given planes are parallel. This means that using a duality principle in projective space we get robust computation. Similar approach can be applied in the Geometry Algebra case as well.

Histogram smoothing for bilateral filter

Michal Seeman
Faculty of Information Technology
Brno University of Technology
Bozetechova 2
CZ 612 66, Brno, Czech Rep.
seeman@fit.vutbr.cz

Pavel Zemcik
Faculty of Information Technology
Brno University of Technology
Bozetechova 2
CZ 612 66, Brno, Czech Rep.
zemcik@fit.vutbr.cz

Keywords
bilateral filtering, histogram, signal filtering, resampling

EXTENDED ABSTRACT

1. INTRODUCTION

Bilateral filtering described and named by Tomasi and Manduchi [Tom98] is the core of several graphics and image processing algorithms. For example, several HDR processing mechanisms do exploit bilateral filtering.

2. HISTOGRAM CONVOLVED WITH GAUSSIAN

One of the possible methods to speed-up the bilateral filtering is through sampling the features of local neighborhoods in the image. The features of the neighborhoods can then be described by histograms. The histograms must be gathered and then convolved with a Gaussian. As the Gaussian has limited spectrum (works, in fact, as a low-pass filter), a space exists to subsample the representation of the histogram in the computer without a loss of precision of the final output.

The Gaussian could be approximated as a linear combination of exponential curves. Convolution with an exponential function could be computed in linear time (but histogram size doesn't grow with image size), because each convolved item could be retrieved from the previous item as follows:

$$\sum_{n=0}^{\infty} a^n f_n \Bigg/ \sum_{n=0}^{\infty} a^n = (1-a) f_n + a[a_{n+1} f^{n+1}]$$

While simulating exact Gaussian is not necessary (rough filter approximation use even rectangular core instead), the result has to be convolution by the same core for any input data. The reason is that different parts of bilateral-filter equation are convolved with the same Gaussian. The exponential curves linear combination fulfills this condition, and therefore ideally fits histogram smoothing needs.

3. HISTOGRAM RESAMPLING

Before applying the convolution, the histogram can be subsampled. For accelerating the computation, already subsampled histogram is gathered. Splitting histogram contribution into several neighboring bins can perform as precise resampling, however high frequency artifacts caused by cheap (e.g. linear resampling) are damped by Gaussian convolution, which works as a low-pass filter.

Retrieval of the values from the subsampled histogram could be done by done by some precise resampling filter (e.g. Lanczos) or again by fast bilinear interpolation (bilateral filter computation needs random access to the histogram, so complex interpolation method accelerating is limited.

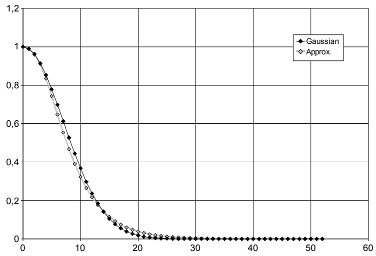


Figure 1. Gaussian σ=10 and simulation using three exponential curves

4. CONCLUSIONS

The bilateral filter performance was measured as PSNR against the brute-force computation. For 29 different images and Gaussian range σ = 4dB (same width as used by Durand and Dorsey [Dur02]) the PSNR was between 43 and 55dB. 40dB is considered as the human eye recognition mark.

5. ACKNOWLEDGMENTS

The work was supported by the MSMT 2B06052 project.

6. REFERENCES

[Tom98] C. Tomasi, T, Manduchi, R. Bilateral Filtering for Gray and Color Images. In Proc. of the International Conference on Computer Vision, pages 839–846. IEEE, 1998.

[Dur02] Durand, F., Dorsey, J. Fast Bilateral Filtering for the Display of High-Dynamic-Range Images. ACM Trans. on Graphics 21 (2002) Proc. of SIGGRAPH conference

Particle systems for riverbed modeling over multilayer height-field data with hardness

Korneliusz Warszawski
University of Zielona Gora
ul. Podgorna 50,
65-246 Zielona Gora, Poland
k.warszawski@weit.uz.zgora.pl

Slawomir Nikiel
University of Zielona Gora
ul. Podgorna 50
65-246 Zielona Gora, Poland
s.nikiel@issi.uz.zgora.pl

Pawel Filipczuk
University of Zielona Gora
ul. Podgorna 50
65-246 Zielona Gora, Poland
p.filipczuk@issi.uz.zgora.pl

Keywords

Particle Systems, Terrain Surface Modeling, Riverbed Generation, Virtual Reality.

EXTENDED ABSTRACT

Striking forms of virtual landscapes are possible when developers, artists and virtual world builders can spend a lot of time manually deforming polygon-nets. Alternatively an acceptable visual level of geological models can be obtained much faster by automated techniques. Several applications of procedural methods have been used in simulation and visualization systems. Virtual terrain is 'a must' element of environment modeling in digital entertainment and game development.

Applying particle systems to terrain surface modeling constitute very efficient alternative for currently used methods. In addition, this technique makes it possible to generate very complex height-field data, similar to mountains or island-like forms and to deform pre-defined (imported from file or pre-generated) landscape structures and to enrich them with canyons or riverbeds. This paper proposes a method that applies particle systems to modify height-field based landscapes with generation of hydrological erosion caused by fast flowing spout. We present the construction of particle system terrain generator/ modifier, its main attributes and how they influence the final product of the modeling process. The key structure for modeled landscape is a typical height-field data, based on the two dimensional array. Each cell represents altitude at the coordination point defined by rows and columns of this array. Such kind of data structure enables fast and simple implementation of terrain deformation and can be relatively easily rendered by any of virtual reality systems. It can be a plain height-map or one imported from external file of real landscape data, e.g.: Digital Elevation Map (DEM). It can be also initially modeled by any automated method, e.g.: Midpoint Displacement, Fault Formation or modeled entirely by the Particle System. For our simulation, we assume that the base terrain consists of several layers of materials with different erosion vulnerability. Furthermore, each layer is also assembled with various materials of the kind. Similar to height-field data, information about hardness of materials, which given landscape is constructed of, can be obtained from geological data. Alternatively, for virtual terrain we can simulate this type of data, by one of previously mentioned height-field generation algorithms.

The proposed system behaves like in a classical particle system, it uses emitters as elements that control starting location, direction and quantity of particles in a given simulation environment. For our simulation process, we select the main attributes for the emitter as: its position and orientation in virtual space and the size of emission window. The emitter starting location should be set at a side and directed to the based landscape. Its emission window determines the area (plane or space) where new particles can be only injected into the simulation system environment. A single particle is defined by Reeves as a dynamic point in three-dimensional space. In our work, we organize particles as a collection of parameterized objects. We choose parameters for particles such as: the current position, directional angle, linear velocity, rotation angle, rotation velocity and the size which define its zone of influence for landscape modification processes. Each emitted particle is moving (rolling) over the surface of terrain structure making deformations at its current position. Scale of the modifications depends on particle parameters and landscape susceptibility for modifications process under a particle influence zone. In the paper we discuss basic problems of landscape hydrological erosion deformations and propose a fast alternative to algorithms based on Computational Fluids Dynamic (CFD) or rainfall erosion. Proposed method is not intended to simulate physically erosion process but focuses on its results for exploitation in virtual environments in real-time simulations and rapid prototyping of terrains.

Numerical Method for Accelerated Calculation of Point Light Source Optical Field

Pavel Zemcik
Faculty of Information Technology
Brno University of Technology
Bozetechova 2
CZ 612 66, Brno, Czech Republic
zemcik@fit.vutbr.cz

Ivo Hanak
Faculty of Information Technology
Brno University of Technology
Bozetechova 2
CZ 612 66, Brno, Czech Republic
hanak@fit.vutbr.cz

ABSTRACT

This contribution describes a method for calculation of a coherent point light source optical field. The optical field calculation is the key part of hologram synthesis methods. Hologram synthesis methods present are extremely computationally extensive and many of them use superposition of point light sources to construct a scene. In a previous work by the authors, an approximation has been introduced that uses fixed point type most of the time and does not impose distance limitations. Speedup gain with no loss of flexibility was achieved. In the presented contribution, it is shown that the approximation can be simplified even further and the resulting error does not impose any additional restrictions and can lead in an additional speedup. Furthermore, the numerical aspects of the new approach are shown and precision and data format considerations are discussed.

Keywords

Spherical wave, optical field, point light source, differential computational scheme

1. INTRODUCTION

Optical field synthesis is a crucial step in hologram synthesis. Even though a hologram can be calculated directly, the focus of the contribution is on optical field calculation because it can be used for generation of several varieties of holograms. This contribution focuses on geometrical aspects of the problem with an aim to calculate the columns of optical field of a point light source (PLS). More details of the application/background will be shown in the full text.

2. OPTICAL FIELD EQUATION

In the case concerned, an optical field is represented as a uniform grid of samples. Each sample represents a complex value. The scene consists of PLS, each of which contributes to the optical field. See Figure 1.

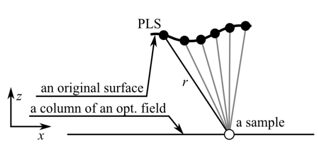


Figure 1: PLS and optical field samples.

A contribution of PLS to an optical field sample is

(1) $$\frac{a}{r}\exp\left[i2\pi\frac{r}{\lambda}+i\varphi\right],$$

where r is a distance depicted in Figure 1, a is an amplitude and φ is a phase of the coherent PLS.

The function (1) is non-linear and cannot be approximated directly. However, if the argument of a complex number and the amplitude are separated, it is possible to approximate them using a quadratic function on a small part of a column (hundreds of samples). The approximation can be performed using a differential scheme with fixed point arithmetic as shown in the previous work of the authors.

3. PROPOSED METHOD

In the proposed method, new approach to the differential scheme is proposed. The scheme is simpler than the previously developed one and uses smaller number of operations per sample; therefore, it leads in further speedup.

Interestingly, the novel method does not require more precise data than the previous one and in some cases it can even lead in reduction of data width. The precision considerations will be presented along with the evaluation of the reduction of operations.

Furthermore, it is shown that the new method leads in more efficient use of hardware resources if the algorithm is implemented in programmable hardware (FPGA).

4. CONSLUSIONS

The conclusion of the actual presentation will include qualitative and quantitative summary of the achieved results and features of the new method

Reusable performance driven facial animation system

Peter Drahoš
Faculty of Informatics and information Technologies,
Slovak University of Technology
Ilkovičova 3
842 16, Bratislava
drahos@fiit.stuba.sk

Keywords
facial modeling, facial expression generation, facial animation, GPU shaders

EXTENDED ABSTRACT

1. INTRODUCTION

Facial animation is and always has been one of the most interesting topics in computer graphics. Modern virtual worlds and games already take advantage of the high performance of modern GPUs and use limited facial animation systems to enrich the user experience. This paper describes modern approach to implementation of a practical reusable animation system for use in generic applications such as: e-learnig, language learning, communication and others. Importance is placed on reusability, ease of use and performance at the expense of the anatomical correctness of animation.

Realistic synthesis of human faces is one of the most interesting and certainly one of the most difficult problems in computer graphics. First publications on the problem can be seen back in early 70's [Par72]. Facial expressions can enhance believability, clarity and involvement of users in communication. Existing implementations and systems have to find optimal balance between realistic models and performance to achieve interactive response.

2. FACE MODEL

Ease of model setup and preparation for animation is important to us. Facial models must be easy to obtain and interchangeable in order to make the system reusable. To achieve real-time performance we use classic polygonal representation of the face, either scanned or synthetic, with a simple color texture for rendering. Mesh deformation and animation is achieved using customized virtual muscles [Wat87] implemented using vertex shaders. The user only needs to set up the facial muscle structure using a simple editor.

3. RENDERING

Visual quality and performance of the system is highly dependent on the rendering process. Modern GPUs are capable or rendering very high detail models at interactive frame rates and can deliver wide range of effects that improve realism [Gem07].

Subsurface light scattering and soft shadows are created using pixel shaders. Skin deformation and wrinkles are dynamically generated from virtual muscle contractions to improve results even further.

Additional details such as hair, eyes, teeth and tongue have to be rendered in real-time and provide additional performance challenges on the rendering process and affect overall performance and quality.

4. ANIMATION

Facial expressions are determined by sets of virtual muscle contraction parameters. Animation is achieved using dynamically generated sets of these muscle contraction parameters, which can be obtained from various sources including: video processing, speech synthesis and classic key frame animation or animation between expression sets.[1]

5. REFERENCES

[Gem07] d'Eon, E. and Luebke, D. GPU Gems 3, Addison-Wesley Professional, Chapter 14, August 2007.

[Par72] Parke, I.F. Computer Generated Animation of Faces, Proc. ACM, August 1972.

[Wat87] Waters, K. A Muscle Model for Animating Three-Dimensional Facial Expression, SIGGRAPH 87, vol. 21, pp. 17–24., July 1987

[1] This work was supported by VG 1/0822/08 grant: Intelligent embedded systems.

Context based controlled Virtual Tours using Viewpoint Entropy for Virtual Environments

RNDr. Ján Lacko
lacko@sccg.sk
Faculty of Mathematics, Physics and Informatics
Comenius University Bratislava
Mlynská dolina
842 48 Bratislava, Slovakia

Marian Maričák
marianmaricak@gmail.com
Faculty of Mathematics, Physics and Informatics
Comenius University Bratislava
Mlynská dolina
842 48 Bratislava, Slovakia

Keywords
Viewpoint entropy, virtual environment, virtual tour.

EXTENDED ABSTRACT

1. INTRODUCTION

In the field of computer graphics, one of the most important parts is creation of virtual environments. In the last years there is a big interest in creation of virtual cities. Man can visit a “real” city using various services (e.g. Google Earth, Microsoft Virtual Earth …). There is a question: How to create an automatic virtual tour with respecting virtual space of environment and best viewpoints for scene parts. In our method we combine viewpoint entropy for obtain the best views and context method for combine the fist *n* views into virtual tour.

2. VIEWPOINT ENTROPY

Viewpoint entropy is low level method based on Shannon entropy. Viewpoint entropy is ratio of relative sizes of visible triangles and surface of bounding sphere.

$$I(S,p) = -\sum_{i=0}^{N_f} \frac{A_i}{A_t} \cdot \log_2 \frac{A_i}{A_t}$$

Where S is scene, p is viewpoint, N_f is number of polygons in the scene and A_i/A_t is visibility of surface regarding to viewpoint p. In our method we use as bounding object hemisphere because in the virtual cities we are limited by ground.

3. CONTEXT METHOD

We can use the hemisphere viewpoint entropy for standalone objects, but if we want to use it in the context of 3D virtual city, we are strongly limited by position of the objects. It is important to find only that camera positions where:

a) The whole object is visible
b) The bounding box (hemisphere) is cut by another and the number of viewpoints at hemisphere is reduced

If both conditions are executed, we can find the first *n* best views around the objects. Than we find the path around the objects using the Voronoi diagram, because there is a condition, that path from one viewpoint in first object to another viewpoint in second object, must lie outside the objects of the whole scene. There are also other conditions for construction of camera path.

In our method there is possibility of adding some information to object (object weight) and for surfaces (surface weight) and use this information from e.g. reconstruction process to control the path of virtual tour through scene.

4. ACKNOWLEDGEMENTS

This paper was supported by VEGA grant

No. 1/0763/09

Shading of Bézier Patches

Jan Bocek
University of Ostrava
30 dubna 22
701 03 Ostrava, Czech Republic
jan.bocek@osu.cz

Alexej Kolcun
IG AS CR
Studentská 1768
702 00 Ostrava, Czech Republic
alexej.kolcun@ugn.cas.cz

Keywords
Beziér patch, shading, normal vector surface.

EXTENDED ABSTRACT

1. INTRODUCTION

There are well known shading algorithms (Gouraud, Phong) for the surfaces shading. The problem is that such algorithms just approximate the real shading.

Precise shading algorithms require the values of normal vectors at all points of rendered surface. Computation of normal vector for Cartesian and triangular parametric surfaces is presented in the paper.

2. CARTESIAN SURFACES

In [Yam] the normal vector surface for Cartesian Bézier surface of degree (*n,m*) is derived and it is expressed as a Bézier surface of degree (*2n-1,2m-1*). According to this rule normal vector surface for bilinear surface is again a bilinear surface.

In [Boc] it is shown, that for normal vector surface in Bézier form

$$N(s,t)=\sum_{i=0}^{n}\sum_{j=0}^{m}\binom{m}{j}\binom{n}{i}(1-s)^{n-i}s^{i}(1-t)^{m-j}t^{j}N_{ij}$$

the control normal vectors N_{ij} fulfil the equation

$$\sum_{i=0}^{n}\sum_{j=0}^{m}\binom{n}{i}\binom{m}{j}(-1)^{i+j}N_{ij}=0\,. \qquad (1)$$

From (1) we can prove that for bilinear surface the surface of the normal vectors is a plane. It means that the normal vectors of bilinear surface we can construct as a linear interpolation of normal vectors in control points. Conclusion of this fact is that

Phong's shading model is an exact shading model for bilinear surfaces.

3. TRIANGULAR SURFACES

The idea of the presentation of the normal vector surface for Bézier triangular surface as a Bézier triangular surface is presented here. Degree of resulting normal vector surface is $2n-2$. So, from the point of view of effective visualization, quadratic Bézier triangles seems to be interesting, so as

- they are natural generalization of planar triangles,
- they are able to model smooth surfaces,
- normal vector surface is also quadratic.

As the quadratic Bézier triangle is the simplest generalization of (planar) triangle, this approach generalizes Phong's shading model.

The results of Phong's and our approach to shading are compared and shown in simple testing rectangular bilinear and triangular quadratic surfaces.

4. ACKNOWLEDGMENTS

The paper has been supported by grant projects AVOZ 30860518 and Internal Fund of Faculty of Science, University of Ostrava.

5. REFERENCES

[Boc] Bocek, J., Effective visualization of Bézier surfaces, MSc. Thesis, University of Ostrava, 2007. (in Czech)

[Jin] Jin, S., Lewis, R.,R., West, D.: A comparison of Algorithms for Vertex Normal Computation. The Visual Computer Vol. 21, No.1-2, 2005 pp.71-82.

[Yam] Yamaguchi, Y. Bézier normal vector surface and its applications, SMA '97, p.26, 1997.

Macromolecular structure interrogation new tool

Francisco Torrens
Institut Universitari de Ciència Molecular,
Universitat de València
Edifici d'Instituts de Paterna, P. O. Box 22085
Spain 46071, València
francisco.torrens@uv.es

Gloria Castellano
Insto. M. Ambiente y CC. Marinas, Universidad
Católica de Valencia *San Vicente Mártir*
Guillem de Castro-94
2nd line of address
Spain 46003, València
gloria.castellano@ucv.es

ABSTRACT

Our program BABELPDB allows browsing and interrogating the native and derived structural features of biological macromolecules using data obtained from the Protein Data Bank (PDB). Major features of BABELPDB are: (1) convert from PDB to other formats, (2) add or remove H atoms, (3) strip the crystallization water molecules and (4) separate the α-carbons (C^{α}). The coordinates obtained with BABELPDB permit characterizing the presence of hydrogen bonds (H-bond). The algorithm for detecting H-bonds is implemented in our program TOPO for the theoretical simulation of the molecular shape. An example is given to illustrate the capabilities of the software: the calculation of the fractal dimension of the lysozyme molecule with (1.908) and without (1.920) H atoms. The numbers compare well with reference calculations performed with our version of the GEPOL program and with the results from Pfeifer *et al*. For proteins, the C^{α} skeleton extracted with BABELPDB allows drawing the ribbon image, which determines the secondary structure of proteins.

Keywords

Information retrieval, Chemical structure, Secondary structure, Solvation water, α-Carbon skeleton.

1. INTRODUCTION

Our program BABELPDB includes subprograms that allows the following options to examine a particular PDB structure: (1) convert from PDB to other formats; (2) add or remove H atoms; (3) strip the water molecules of crystallization and (4) separate C^{α} atoms [Tor01]. BABELPDB would seem particularly suited to educational purposes and an example of how it might be used is given.

2. COMPUTATIONAL METHOD

Program BABELPDB allows browsing–interrogating native–derived structural features of biological macromolecules using data obtained from the Protein Data Bank (PDB). The coordinates obtained with BABELPDB allow characterizing the presence of hydrogen bonds (H-bonds). The algorithm for detecting H-bonds is implemented in program TOPO for the theoretical simulation of the molecular shape.

3. RESULTS AND DISCUSSION

An example illustrates the capabilities, *i.e.*, calculation of the fractal dimension of lysozyme with and /without H atoms. The numbers compare with reference calculations performed with program GEPOL. For proteins, C^{α} skeleton allows drawing ribbons image, which determines secondary structure.

Provisional conclusions follow.

1. Our program BABELPDB has been written for the search, retrieval, analysis and display of information from database PDB. Several options are allowed.

2. The fractal dimension of lysozyme has been calculated with and without H atoms.

3. For proteins, the C^{α} skeleton extracted with BABELPDB allows drawing the ribbon image, which determines the secondary structure of proteins.

4. It is still to be explored the methodological application of these computational programs enriching the present teaching techniques.

4. REFERENCE

[Tor01] Torrens, F., and Castellano, G., A new tool for interrogation of macromolecular structure in chemical education, in: Seijas, J.A., and Vázquez Tato, M.P. (eds.). Synthetic Organic Chemistry, MDPI, Basel (Switzerland), Vol. 12, pp. 1-17, 2008.

Geometric Algebra Computing on the CUDA Platform

Christian Schwinn
TU Darmstadt, Germany
Department of Computer Science
schwinn@rbg.informatik.tu-darmstadt.de

Andreas Görlitz
TU Darmstadt, Germany
Department of Computer Science
A.Goerlitz@stud.tu-darmstadt.de

Dietmar Hildenbrand
TU Darmstadt, Germany
Department of Computer Science
dhilden@gris.informatik.tu-darmstadt.de

ABSTRACT

Geometric Algebra (GA) is a mathematical framework that allows a compact, geometrically intuitive description of geometric relationships and algorithms. These algorithms require significant computational power because of the intrinsically high dimensionality of geometric algebras. Algorithms in an n-dimensional GA require 2^n elements to be computed for each multivector. GA is not restricted to a maximum of dimensions, so arbitrary geometric algebras can be constructed over a vector space $\mathcal{V}_n$. Since computations in GA can be highly parallelized, the benefits of a parallel computing architecture lead to a significant speed-up compared to standard CPU implementations, where elements of the algebra have to be calculated sequentially. An upcoming approach of coping with parallel computing is to use general-purpose computation on graphics processing units (GPGPU). In this paper, we focus on the Compute Unified Device Architecture (CUDA) from NVIDIA [2]. We present a code generator that takes as input the description of an arbitrary geometric algebra and produces an efficient implementation of geometric products for the underlying algebra on the CUDA platform.

Keywords: Geometric Algebra, Geometric Computing, GPU, CUDA.

1 INTRODUCTION

Geometric Algebra (GA) has become more and more popular in different fields of research. Due to its above mentioned properties it is possible to develop very compact algorithms while keeping them intuitive. One major drawback is the reduced performance when applying GA algorithms without further processing. But recent research has shown that it is possible to speed up GA algorithms drastically by means of static code optimization and switching to parallel computing architectures like field-programmable gate arrays (FPGA) or CUDA, which even leads to performance improvements compared to standard implementations [1].

Many applications of GA require a very large number of calculations to be processed, e.g. feature extraction algorithms [3]. In most cases it is necessary to define highly customized non-standard algebras in order to fit the problem statement. What these problems have in common is a remarkable amount of parallelization required to fulfill the constraints of reduction. In theory, this can lead to virtually infinite operations which have to be executed concurrently in order to decrease the order of time complexity.

In this paper, we investigate the potential of executing GA operations on parallel architectures. We focus on the implementation of arbitrary geometric products on the CUDA platform as a means for evaluating the performance of parallel computing in GA compared to standard implementations. We implement the calculation of the geometric product without any restrictions to the underlying algebra and associated metric and signature. We exploit the property that elements of the result multivector can be easily computed in parallel, e.g. each one in a separate (parallel) thread on a CUDA-enabled GPU. Therefore, we present a code generator that produces the CUDA code which calculates the geometric product of the related algebra and can be embedded in order to speed up algorithms.

Our approach takes as input the description of an n-dimensional algebra in terms of a metric or signature and calculates a data structure describing the elementary product of all possible combinations of basis blades. This can be seen as a lookup table that is first optimized according to GA simplifications and then used to generate expressions for the individual result multivector components that only depend on the coefficients of the input multivectors to be multiplied. Finally, these expressions are translated into parallel CUDA code that calculates the result multivector and can be used for efficient calculation of the geometric product.

As a result, we evaluate the performance of the parallelized geometric product to get an estimation on the impact of parallel computing on problems in GA.

REFERENCES

[1] H. Lange, F. Stock, D. Hildenbrand, and A. Koch. Acceleration and Energy Efficiency of a Geometric Algebra Computation using Reconfigurable Computers and GPUs. *FCCM*, 2009.

[2] NVIDIA. The CUDA home page. `http://www.nvidia.com/object/cuda\_home.html`, 2009.

[3] M. T. Pham, K. Tachibana, E. M. S. Hitzer, T. Yoshikawa, and T. Furuhashi. Classification and Clustering of Spatial Patterns with Geometric Algebra. *AGACSE*, 2008.

A GPU Supported Approach to Partial Registration of 3D Scan Data

Adrie Kooijman / Joris Vergeest

Delft University of Technology
Faculty of Industrial Design Engineering

Landbergstraat 15
2628CE, Delft, the Netherlands

a.kooijman@tudelft.nl / j.s.m.vergeest@tudelft.nl

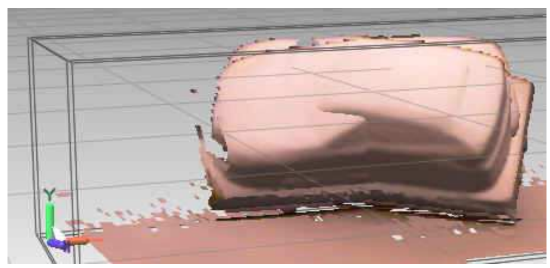
EXTENDED ABSTRACT

We present a method of partially matching two shapes, aimed to support 3D scanning applications. To digitize the surface of a 3D object, a number of scan views are taken from different angles, collectively representing the entire surface of the object. The registration process assigns a transformation to the scan views such that the surface of the object can be reconstructed. The motivation of our work is to provide an initial geometric model to a designer who develops the shape further using Computer-Aided Design. We propose a method of pairwise shape matching of partial scans, without depending on any distinct geometric feature type.

The problem of matching two shape representations *A* and *B* is to find a rigid transformation *M* such that *MB* and *A* have the correct relative placement. *A* and *B* each represent a portion of a surface of a 3D object and there exist subsets of *A* and of *B* representing the same portion of that object, called the overlap region. The congruence of the overlapping portions has been defined by a matching index *I*, which is maximized to attain the right relative placement of the two shapes. At these maxima we have candidate overlaps and can compute the transformation matrix *M*. We propose a method of pre-alignment based on quasi-exhaustive sampling of the 6D configuration space of *M* and *I(A, MB)* representing the degree of consistency of shapes *A* and *MB*. The preprocessing we do is down-sampling *MB* from about 20,000 to 250. Shape *A* is left at full sampling precision. Then we compute the distance of down-sampled *MB* to *A* as:

$$I(A, MB) = \frac{1}{n_B}\sum_{i=1}^{n_B}(c + d_i^{\,2})^{-1}, \quad d_i = \min_{j=1,\, n_A} |b_i - a_j|,$$

where a_j and b_i are points in *A* and *MB*, Damping term *c* will not influence the location of the maximum for *I*. Since the calculation of *I(MB)* is highly parallelizable we have partially implemented the computation on a Graphical Processing Unit (GPU). For each combination of *x*- and *y* shift in *M* a thread block is launched on the GPU. The *z*-shift is handled within the thread block. Each thread calculates the contribution of a single point to index *I*. Rotations of *MB* are performed on the CPU. We have investigated the usability of *I* from two respects: 1) the goodness for detecting overlap regions and 2) efficiency with regard to interactive application. We present preliminary results here, based on a small number of scanned objects.



In the figure above, two overlapping scan views of a simple car model are shown. The number of points in *A* and *B* are 6107 and 6518, respectively. Using 16 steps over 40 degrees for the three rotation axes and 16 steps for the three translation directions over the bounding boxes of *A* and *B*, *I* was computed 16^6 = 16.8 million times. The highest *I* produced the correct *M* and hence registration of the two scan views, found in a few minutes of computation time.

We have not yet achieved sampling of the full 6D configuration space with sufficiently small step size in acceptable times for interactive application. In this case the user should get feedback about success or failure of the matching of the scan view just taken after not much longer than 10s. Using the current algorithm, the method is applicable if the user would follow operation instructions (typically to exert limited rotation) to ensure that sufficient overlap remains between two scans. In order to release these restrictions and to speed up the calculation, several improvements are necessary. The definition of *I* could be changed or *A* could be down sampled too. To our best knowledge there are no previous reports about practically usable results of similar methods.

Realtime KLT feature point tracking for High Definition video

Hannes Fassold[1]
hannes.fassold@joanneum.at

Jakub Rosner[2]
jakub.rosner@joanneum.at

Peter Schallauer[1]
peter.schallauer@joanneum.at

Werner Bailer[1]
werner.bailer@joanneum.at

EXTENDED ABSTRACT

The automatic detection and reliable tracking of feature points is an important part of many computer vision methods. It is used e.g. in algorithms for computing structure from motion, camera calibration, camera motion estimation and in a variety of other computer vision fields. One of the most widely used algorithms for feature point detection and tracking is the KLT tracker proposed by Kanade, Lucas and Tomasi. Although this algorithm provides good quality tracking results, its computational complexity limits its usage to standard definition video for realtime applications when implemented in software. For realtime feature point tracking in high definition video an alternative technology needs to be considered. Because of its tremendous computation capabilities Graphic Processing Units (GPU's) gain significant attention for computer vision tasks. In this paper we report experiences and results of porting the KLT tracker to the GPU. We use the "Compute Unified Device Architecture" (CUDA), a C-like programming language for general-purpose GPU programming proposed by NVIDIA. The two key components of the KLT tracker are feature point detection and feature point tracking. The feature point detection takes into account only corner-like feature points which are well suited for reliable tracking. Detected points are then tracked by calculating the translation which minimizes the dissimilarity between point-centered windows in two consecutive images. For the feature point detection, we propose to do all steps of the detection process, except the last one, on the GPU. The first steps can be ported efficiently to the GPU as they are good parallelizable. In contrast, the last step, which enforces a certain minimum distance of each feature point to the others, is not well suited for execution on the GPU as it is inherently serial. The feature point tracking is done on a multi-resolution image representation (pyramid) to allow for large motion. The calculation of the pyramid, and also the tracking itself is done completely on the GPU. The tracking is well suited for parallelization, as for each pyramid level the estimation of the optical flow for one feature point can be done independently from the others. We compare the CUDA implementation with the highly optimized OpenCV KLT tracking functions in terms of quality and speed. A significant speedup with respect to the OpenCV routines can be noticed. Furthermore, some additional experiments are done to evaluate the influence of different paremeterization (window size, number of tracked points, etc.) on the runtime of the GPU routine. Our GPU implementation achieves realtime (> 25 frames per second) performance for High Definition (HD) video sequences, successfully tracking several thousands of points. As nearly the entire work for feature point detection and tracking is done on the GPU, the CPU is free for doing higher-level computer vision tasks, e.g. for estimating the camera motion from the tracked points. In summary, the GPU implementation achieves a significant speedup compared with an optimized CPU implementation, which allows the analysis of high resolution video sequences in realtime.

Keywords

KLT, feature point tracking, Lucas Kanade, corner detection, optical flow, motion estimation, GPU, CUDA

---

[1] JOANNEUM RESEARCH, Institute of Information Systems & Information Management, Steyrergasse 17-19, 8010 Graz, Austria

[2] Silesian University of Technology, Faculty of Automatic Control and Robotics, Ulica Academicka 2, 44-100 Gliwice, Poland

Preprocessing of microscopy images via Shannon's entropy

Jan Urban
Institute of Physical Biology
Zamek 136
373 33, Nove Hrady
Czech Republic
urban@greentech.cz

Jan Vanek
Institute of Physical Biology
Zamek 136
373 33, Nove Hrady
Czech Republic
vanekyj@kky.zcu.cz

Keywords
Image analysis, phase-contrast microscopy, Shannon's entropy

EXTENDED ABSTRACT

1. INTRODUCTION

This document presents a novel method of image preprocessing based on Shannon's entropy. The method is developed specially for microscopy images captured in phase-contrast mode. But it can be used in many others applications. Many variants of using entropy are described in the paper and differences are discussed. Performance of individual methods is illustrated. Finally, implementation on graphics cards to overpass higher computation requirements of the algorithm is described. The total speed-up of the processing is about 3600x.

2. RESULTS

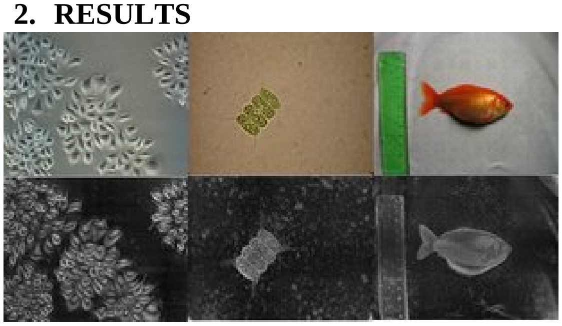


Figure 1. From left to right: *HeLa* cell line experiment using phase contrast; Green alga *Scenedesmus* using bright field microscopy; *Chinese goldfish* digital camera picture

One of the powerful tools is based on evaluation of information entropy using the Shannon equation. Although the formula is well known, there are still many ways how to use it.

To prove the usability of developed approach, we performed the tests on several biological samples from different experiments (Figure 1). For all kinds of images the algorithm performance is very good.

3. GPU IMPLEMENTATION

The algorithm described in the paper gives a very good results but its time-consumption is huge. Some much faster implementation was needed for practical usability of our algorithm. The first implementation was done in Matlab. We started optimization and a speed-up by factor 12x was achieved mainly due to precomputed histograms. The C++ implementation with logarithms in single-float precision gives a speed-up by factor 6. Additional 2x speed-up was obtained by using AMD performance library Framewave. A multi-core implementation is no problem also. The third step of optimization was an implementation on graphics cards. The algorithm was done in CUDA. Times of processing 2Mpix and 6Mpix images and cumulative speed-ups for all implementations are shown in Table 1.

Version	Elapsed Time		Total speed-up
	2Mpix	6Mpix	
Matlab	27 min	>2 hour	-
Matlab optimized	136 s	13 min	12x
C++	21.8 s	128 s	72x
C++ Framewave	11.2 s	64 s	150x
CUDA GPU	0.45 s	2.75s	3600x

Table 1. Processing times and speed-ups

4. ACKNOWLEDGMENTS

This work was supported by grant HCTFOODA/CZ0046/1/0008 of EEA funds**.**

Integrated visualization of vector quantization by multidimensional scaling

Olga Kurasova
Institute of Mathematics and Informatics
Akademijos str. 4
LT-08663, Vilnius, Lithuania
Kurasova@ktl.mii.lt

Alma Molytė
Institute of Mathematics and Informatics
Akademijos str. 4
LT-08663, Vilnius, Lithuania
Alma.Molyte@gmail.com

Keywords
Vector quantization, multidimensional data visualization, integrated visualization.

EXTENDED ABSTRACT

Visualization is a powerful tool, assisted for a human being in the process of knowledge discovery in large data sets. The data represented in a visual form are more informative then the raw numeric data. Here we combine vector quantization and visualization.

Vector quantization is a classical approximation method that usually forms a quantized approximation to the distribution of the input data vectors $X_l \in R^n$, $l=1,...,m$, using a finite number of codebook vectors $M_i \in R^n$, $i=1,...,N$, here N is the number of codebook vectors. Once the codebook is chosen, the approximation of X_l means finding the codebook vector (neuron) M_i closest to X_l [Koh01]. Here two methods based on neural networks are analyzed: self-organizing map (SOM) [Koh01] and neural gas (NG) [Mar91].

After the networks training, n-dimensional input vectors X_l have been mapped – each input vector is related to the nearest neuron. The neurons related with the input vectors are called neurons-winners.

In the case of the rectangular topology of SOM, we can draw a simple table with cells corresponding to the neurons. However, the table does not answer the question, how much the vectors of the neighbouring cells are close in the n-dimensional space. A natural idea arises to apply the distance-preserving method to an additional visualization of the neurons-winners. Multidimensional scaling may be used for this purpose.

Multidimensional scaling (MDS) refers to a group of methods that are widely used for a dimensionality reduction and visualisation of multidimensional data. The starting point of the multidimensional scaling is a matrix consisting of pairwise proximities of the data. The goal of MDS is to find lower-dimensional vectors, such that the distances between the vectors in the lower-dimensional space were be as close to the original distances (or other proximities) as possible [Bor05].

In [Dze06], the integrated combination of SOM and the MDS-type method is proposed and discussed. Multidimensional vectors are visualized by MDS taking into account the process of SOM training. It is shown that smaller quantization errors are obtained by NG than SOM [Kur09], so it is purposeful to combine NG with MDS, too.

In this paper, we propose the integrated combination of neural gas and multidimensional scaling.

REFERENCES

[Dze06] Dzemyda, G., and Kurasova, O. Heuristic Approach for Minimizing the Projection Error in the Integrated Mapping. European Journal of Operat. Research, 171(3), pp. 859–878, 2006.

[Bor05] Borg, I., and Groenen, P. Modern Multidimensional Scaling, Springer, 2005.

[Koh01] Kohonen, T. Self-Organizing Maps, 3rd ed, Springer Series in Information Science, 30. Springer, 2001.

[Kur09] Kurasova, O., and Molytė, A. Combination of Vector Quantization and Visualization. In: Perner, P. (ed.,) Machine Learning and Data Mining in Pattern Recognition, LNAI, 5632, pp. 29–43, 2009.

[Mar91] Martinetz, T.M., Schulten, K. J. A Neural-Gas Network Learns Topologies. In: Kohonen, T., Mäkisara, K., Simula, O., Kangas J. (eds.) Artificial Neural Networks, pp. 397–402, 1991.

Adaptive Segmentation of Deforming Mesh sequence into Near-Rigid Clusters

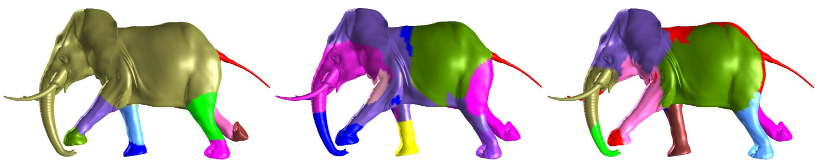
Rachida Amjoun and Wolfgang Straßer
WSI / GRIS
University of Tübingen, Germany

ABSTRACT

Segmenting 3D deforming triangular mesh into near-rigid components is useful for many practical applications. In this paper, we introduce a new hierarchical and adaptive approach for clustering mesh vertices with similar motions using local characteristics. We extend motion based static clustering developed in the context of compression, to find a more flexible clustering process. We want to minimize the deviation in the local coordinate frame of each cluster to obtain *low-motion partitioning*.
The algorithm starts with an initial partitioning of very small number of clusters (using static clustering), then iteratively find new seed point of large motion, add a new cluster and update the partitioning until the cost function converges or the predefined number of clusters is attended. This algorithm can be well applied to different kinds of deforming meshes whose connectivity and the number of vertices does not change over time, and no information about how the motion is generated, is necessary.
To evaluate the clustering approach, we introduced a metric defined by the average deviation of the local coordinates over all clusters, over time. Visually as well as metrically, iterative clustering approaches exhibit better partitioning than static clustering or region growing based algorithms.

Keywords: dynamic meshes, clustering, local coordinate system, euclidian distance.

Segmentation examples: the elephant (10 clusters), cow (6 clusters), chicken (10 clusters), dolphin (9 clusters) and dance (14 clusters) animations.



A comparison between the iterative clustering algorithm (left), static clustering (middle) and region growing (right) based algorithms for segmenting the same elephant sequence.

Using Geometric Algebra for Navigation in Riemannian and Hard Disc Space

Werner Benger
Center for Comp. & Technology
Louisiana State University
239 Johnston Hall
Baton Rouge, LA 70803, USA
werner@cct.lsu.edu

Andrew Hamilton
Center for Astrophysics
and Space Astronomy
JILA, University of Colorado
Boulder, CO 80309, USA
Andrew.Hamilton@colorado.edu

Mike Folk / Quincey Koziol
The HDF Group
1901 So. First St.
Suite C-2
Champaign, IL 61820. USA
mfolk@hdfgroup.org

Simon Su
Princeton Institute for
Computational Science and
Engineering
345 Peter B. Lewis Library
Princeton, NJ 08544, USA
simonsu@princeton.edu

Erik Schnetter
Center for Comp. & Technology
Dept. of Physics & Astronomy
Louisiana State University
Baton Rouge, LA 70803, USA
schnetter@cct.lsu.edu

Marcel Ritter / Georg Ritter
Department of Computer Science
University of Innsbruck
Technikerstrasse 21a
A-6020 Innsbruck, Austria
csab7885@uibk.ac.at

A “vector” in 3D computer graphics is commonly understood synonymous to a triplet of three floating point numbers, eventually equipped with a set of functions operating on them. This hides the fact that there are actually different kinds of vectors, each of them with different algebraic properties and consequently different sets of functions. In this article we will discuss what meta-information is required to uniquely identify a specific type of vector in addition to its purely numerical values. Differential Geometry (DG) and Geometric Algebra (GA) are the appropriate mathematical theories to describe these different types of “vectors”. They consistently define the proper set of operations attached to each class of “floating point triplet”. We shortly review the various types of “vectors” in 3D computer graphics such as “normal” and “axial” vectors, their relations to rotations and quaternions, and connect these to the terminology of co-vectors and bi-vectors in DG and GA. We will elaborate on less known facts about the use of bi-quaternions in relativity, allowing for instance a more insightful formulation to determine the Newman-Penrose pseudo scalars from the Weyl tensor.

We will show the concrete application of these concepts in two independently developed computer graphic software packages, where Geometric Algebra is used for navigating the camera position in space and time. It will be demonstrated that use of GA and the sometimes mystified concept of spinors eases implementation and intuition significantly. Another application example is given by a simulation code solving Einstein’s equation in general relativity numerically on supercomputers, outputting the Newman-Penrose pseudo scalars as primary quantities of interest to study gravitational waves, both for visualization and observational verification.

Beyond using these data types internally within an application in volatile memory, we will also describe how the metadata information required per “vector” can be provided in persistent storage, such as for writing datasets of vectors archived within files. Large datasets being expensively collected or generated by simulations requiring millions of CPU hours, it is increasingly important and difficult to be able to share and correctly interpret such datasets years after their generation, across different research groups from different fields of science. A unique, standardized, extensible identification of the geometric properties of the dataset elements is a necessary pre-requisite for this.

Apart from visualization, post-processing and further analysis of existing or newly generated datasets is important in the high performance community (HPC). Frequently, different tools expecting different data formats have to be coupled in this analysis chain. Data have to be transferred between different HPC or data centers, and have to be converted to different file formats. A standard way of expressing geometric properties of non-scalar (vector or tensor) values can prevent conversion errors, similar to the way in which the IEEE standard for floating point values enables sharing floating point values. We utilize the mechanisms as provided by the HDF5 library here, a generic self-describing file format developed for large datasets as used in high performance computing. It allows specifying metadata in addition to the purely numerical data, providing an abstraction layer for specifying the mathematical properties on top of the lower-level binary layout. We will give concrete examples on how to use this abstraction layer to formulate vector properties for DG and GA, and discuss the benefits and potential that lays in this approach.

Goodness-of-fit tests for brightness distribution in grey level images

Jose A. Malpica
Alcala University
Mathematics Department
Escuela Politecnica E-225
Campus Universitario
28871 Madrid, Spain
josea.malpica@uah.es

Maria C. Alonso
Alcala University
Mathematics Department
Escuela Politecnica E-227
Campus Universitario
28871 Madrid, Spain
mconcepcion.alonso@uah.es

Keywords
Kolmogorov-Smirnov test, low-level image similarity, Monte Carlo Sobol sequence, image retrieval.

EXTENDED ABSTRACT

1. INTRODUCTION

The important of images analysis is increasing in many real world applications. Among the many operations that could be done with images, retrieval and matching are very common. Text-based image retrieval is used but its main limitation comes from the difficulty in describing an image with a limited, generally sort, number of words. An alternative is Content-Based Image Retrieval, which try to reinforce the low level visual feature analysis with the richness of human semantics. In the last few years many paper have being published in this area, a review can be seen in [Liu07]. They conclude that there is not a generic approach for high level semantic-based image retrieval. The research is on the specific application with limited results. A lot more research is needed in the characteristic of images; obtaining relevant in the low level feature could be crucial for the semantic level. Our paper concentrates on looking for good measurement for image matching and retrieval in the low level.

2. THE KS TEST

The statistical comparison of distributions is important for data analysis. One popular method of comparing two empirical distributions is the test of Kolmogorov-Smirnov (KS) ([Kol33] , [Smi39]). This is a non-parametric tests relate to the greatest discrepancy between two cumulative distributions.

The KS test was extended for data in two dimensions by Peacock (1983). Unlike for the one-dimensional case where the work of Kolmogorov gave a good yield for theoretical grounds, the Peacock´s KS test for two dimensions has no theoretical justification for asymptotic convergence; therefore the significance of some parameters has to be found from numerical experimentation.

3. APLICATION TO IMAGES

The KS test has been used in this paper to compare the brightness of two images, and a parameter is given in order to say if two images are similar or not. The test is performed in order to see if the values in the two images are coming from the same distribution and therefore are similar images. To do this the Sobol quasi-random sequence for two dimensions is used.

4. RESULTS

The method is straightforward in the sense that there is not parameter tuning. Given the two images the method is going to tell is the images are similar or different to a signification level. The limitations are that is not rotational, translation or scale invariance.

5. REFERENCES

[Kol33] Kolmogorov, A.N. Sulla determinizione empirica di una legge di distribuzione (On the derivation of a distribution law) Giornale del´Istituto Italiano degli Attuari, 4, pp. 83-91, 1933.

[Liu07] Liu Y., D. Zhang, G. Lu and W.-Y. Ma. 2007. A survey of content-based image retrieval with high-level semantics. Pattern Recognition 40, No. 1, pp. 262-282, 2007.

[Smi39] Smirnov N.V. On the estimation of the discrepancy between empirical curves of distribution function for two independent samples. Bulletin of Mathematics University of Moscow, 2, pp 3-14, 1939.

Creating Editable 3D CAD Models for Point Cloud Slices

Antonis Protopsaltou
Dept of Computer Science
University of Ioannina,
Ioannina, Greece, GR45110
antonis@cs.uoi.gr

Ioannis Fudos
Dept of Computer Science
University of Ioannina
Ioannina, Greece, GR45110
fudos@cs.uoi.gr

Keywords
Reverse engineering, slice reconstruction, thinning, optimal fitting

EXTENDED ABSTRACT

Reverse Engineering is a complex process that is central to industry, arts, archaeology and architecture. In this paper we focus on re-engineering solid objects for which we have acquired the point cloud of their boundary. Subsequently we wish to obtain a 3D CAD model which is editable and manufacturable. Most previous approaches have dealt with this problem considering only mechanical parts and employing feature-based knowledge to detect and represent holes, chamfers, extrusions, protrusions etc. It is important to provide means for editing 3D objects that respect object morphology and topology. Various authors have considered creating reverse engineered 3D models. Some researchers have dealt with the tedious task of making their model editable. This is often accomplished by incorporating local and global geometric constraints in the CAD model. In plain solid reconstruction a geometric model is captured directly from the geometry of the point cloud acquired by 3D laser scanning. This method is commonly used in modeling sculptures in arts. These techniques are quite accurate but do not support large scale modifications, additions or other high level operations to the extracted model.

In this paper, we present a novel computer aided re-engineering paradigm based on careful slicing of a 3D point cloud and sophisticated post processing of all resulting cross sections. Post processing aims to eliminating noise and partitioning the point set to point sequences that correspond to low degree curve segments. The curve segments are then approximated using quadratic rational Bezier curves. We then subdivide the curve segments in equilength chord segments and use the corresponding points to perform 3D skeleton-driven mesh reconstruction. Figure 1 illustrates the overall process.

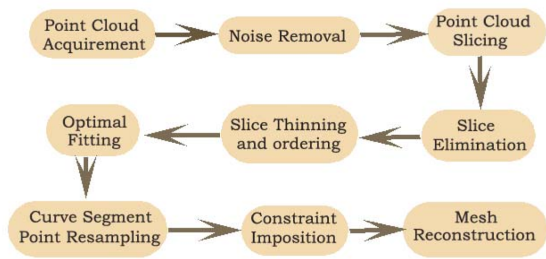
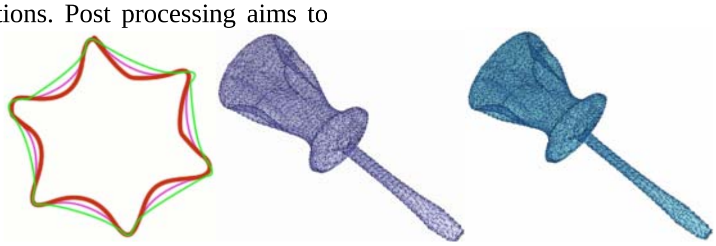


Figure 1: Process Steps

We have developed an effective and efficient method to build a 3D CAD model from a given point cloud that represents accurately the surface of the object. We have evaluated the usability of our method with very good results even for users with no former CAD software experience. Our method provides tools for robust and accurate editing of the produced CAD model prior to remanufacturing.



(a) (b) (c)

Figure 2. (a) Automatic intermediate slice generation. First slice is in red, second slice is in green and the intermediate slice is shown in purple. (b) Reconstructing the original object (c) Edited object

Image normalization under projective transforms

Mashtalir Sergey
ass. prof. of Computer Science Department
Kharkov National Univercity of Radio Electronics
Lenin Ave, 14
Ukraine, 61166, Kharkov
mashtalir_s@kture.kharkov.ua

Putyatin Evgeniy
prof., chief of Computer Science Department
Kharkov National Univercity of Radio Electronics
Lenin Ave, 14
Ukraine, 61166, Kharkov
informatika@kture.kharkov.ua

Keywords
Projective group, geometric image transformation, normalization

EXTENDED ABSTRACT

1. INTRODUCTION

Image processing under geometrical transformations induced by change of a mutual location and/or orientation of object and video sensor has been widely considered for the last half of the century. Depending on the viewing conditions next models of transformations were used: translation, scaling, rotation, skewing, Euclidian motion, Euclidean similarities, Affine and projective groups of transformations. Starting with pioneer works [Hu62a, Ama66a], the progress development took two directions. First (the basic one) is a construction of invariants to the given kind of deformations. Second, named as image normalization, provides search of the transformation parameters and alignment (by back propagation) of the image to template. This approach besides recognition via comparing with the template, provides spatial transformations evaluations, i.e. perform the image tracking. It should be noted that the most valuable results have been obtained for the linear geometrical transformations. For the nonlinear image transformations till now the acceptable results were obtained in [Kin03a, Put90a] for separate subgroups of projective group only. In this paper on the base of groups representation theory a method for image projective transformations normalizators synthesis is proposed.

2. PROJECTIVE TRANSFORM

The task is formulated as following. Let image $\mathrm{B}(x, y)$ be connected with some template $\mathrm{B}_0(x, y)$ by the projective group effect, i.e.

$$\mathrm{B}_0(x, y) = \mathrm{B}\left(\frac{a_{11}x + a_{12}y + a_{13}}{a_{31}x + a_{32}y + a_{33}}, \frac{a_{21}x + a_{22}y + a_{23}}{a_{31}x + a_{32}y + a_{33}}\right)$$

or in the homogeneous coordinates

$$\mathrm{B}_0(u, v, z)\big|_{z=1}^{\mathrm{T}} = \mathrm{B}(\mathrm{A}(u, v, z)^{\mathrm{T}}),$$

where $x = u/z$, $y = v/z$, A is a projective transformation matrix, $det\mathrm{A} \neq 0$. Search of the matrix parameters provides image normalization (alignment to the template).

Projective transformations of the distortions are represented by the action of $\mathrm{SL}(3, \mathbb{R})$ group. The core of matrix A search method consists in constructing the equivariant mapping from the image space into the set of the homogeneous polynomials of second and forth order, and then quadratic form pairs. In other words images projective transformation are replaced by linear transformation and unknown parameters are defined by reducing the quadratic forms pair to canonical form. Unfortunately computational model of this procedure is rather complicated. Specifically for the concomitant constructing the images second derivatives are used, what tends to the significant reduce of stability. Nevertheless despite of the number of the limitations using the finite-dimensional representations of the geometrical transformations groups has certain perspectives for 3D object transformations evaluations.

REFERENCES

[Ama66a] Amari, S. Theory of Normalization of Pattern Signals in Feature Spaces. Electronic and Communication in Japan, Vol. 49, No. 7, pp.104-113, 1966.

[Hu62a] Hu, M.K. Visual pattern recognition by moment invariants, in IRE Transaction on Information Theory, Vol. IT-8, No. 2, pp.179-187, 1962.

[Kin03a]Kinoshenko D., Mashtalir V., Orlov A., Yegorova E. Method of Creating of Functional Invariants under One-Parameter Geometric Image Transformations. Pattern Recognition. Lecture Notes in Computer Science. Berlin-Heidelberg: Springer-Verlag. Vol. 2781, pp.68–75, 2003.

[Put90a]Putyatin Ye. P., Averin S.I. Image processing in robotics. Mashinostroyeniye, 1990, 319p.

Neural Network-Based Large Datasets Visualization Using a Reduced Training Subset

Viktor Medvedev
Institute of Mathematics and Informatics
Akademijos str. 4
LT-08663 Vilnius, Lithuania
Viktor.M@ktl.mii.lt

Gintautas Dzemyda
Institute of Mathematics and Informatics
Akademijos str. 4
LT-08663 Vilnius, Lithuania
Dzemyda@ktl.mii.lt

Keywords
Neural Network, Visualization, SAMANN, Large Multidimensional Datasets, Parallel Computing.

EXTENDED ABSTRACT

1. INTRODUCTION

The research area of this work is visualization of large multidimensional datasets. Visualization is a useful tool for data analysis, especially when the data are unknown. However, if the dimension is huge, it is difficult to produce robust visualization. Therefore, the dimensionality reduction technique is needed. The aim of the projection method is to represent the input data items in a lower-dimensional space so that certain properties of the structure of the data set were preserved as faithfully as possible.

This paper focuses on dimensionality reduction methods as a tool for the visualization of large multidimensional datasets. The most common methods allocate a representation of each data point in a lower-dimensional space and try to optimize these representations so that the distances between them were as similar as possible to the original distances of the corresponding data items.

2. DATA VISUALIZATION USING NEURAL NETWORK

SAMANN [Mao95], [Med06] is an unsupervised backpropagation algorithm for training a multilayer feed-forward neural network in order to perform Sammon‘s projection [Sam69]. This algorithm preserves all the interpattern distances as well as possible. The SAMANN network offers a generalization ability of projecting new multidimensional data, which are not present in the original Sammon’s algorithm.

A drawback of using SAMANN is that the training process is extremely slow. The ways of speeding up the network training process are: a) to use parallel computing; b) to train the network by a subset of the primary (multidimensional) datasets.

The investigations of the visualization quality using the SAMANN neural network with sets of different sizes of the primary datasets have showed that it is possible to train the network using only a part of the analyzed dataset without loss of accuracy. In this work real datasets were used to investigate the ability to visualize large multidimensional datasets using SAMANN.

The results of experiments have shown that it is suffices to form a subset of 5–10% of the analyzed dataset for training the SAMANN network. In such case, lower projection errors are obtained faster (more than 10 times) than by training with all the points of the set.

The usage of the reduced dataset for SAMANN training enabled us to process large datasets and to get good enough results within reasonable time. Parallel computing has been used in the experiments to find the optimal subset for the neural network training.

3. ACKNOWLEDGEMENT

The research is partially supported by the Lithuanian State Science and Studies Foundation project “Information technology tools of clinical decision support and citizens wellness for e.Health system” (No. B-07019).

4. REFERENCES

[Sam69] Sammon, J.W. A Nonlinear Napping for Data Structure Analysis, IEEE Transactions on Computers 18, pp. 401–409, 1969.

[Mao95] Mao, J. and Jain, A.K. Artificial Neural Networks for Feature Extraction and Multivariate Data Projection, IEEE Trans. Neural Networks Vol. 6, pp. 296–317, 1995.

[Med06] Medvedev, V. and Dzemyda, G. Optimization of the SAMANN network training. Journal of Global Optimization, Springer, Vol. 35(4), pp. 607–623, 2006.

Fourier and Wavelet Transformations in Geometric Algebra

Eckhard Hitzer

Department of Applied Physics
University of Fukui
3-9-1 Bunkyo
Japan 910-8507, Fukui

hitzer@mech.fukui-u.ac.jp

Keywords

Clifford geometric algebra, quaternions, space-time algebra, multivector-valued functions, Clifford Fourier transform, Clifford wavelet transform, Clifford Gabor wavelets, uncertainty principle

EXTENDED ABSTRACT

First we introduce basic Geometric Algebra (GA) multivector functions. For these functions we define the coordinate independent *vector differential* and the *vector derivative*. This includes explicit examples and basic differential geometric calculus rules.

Then we motivate and define the *GA Fourier Transformation* (GA FT) for the GA of real Euclidean 3-space and give examples of its application. We give an overview of its most important properties (higher dimensional geometric generalizations of scalar complex Fourier Transformation properties). We show how to *generalize* the GA FT to *higher dimensions* and explain what role characteristic GA non-commutativity plays. Known applications include uncertainty, LSI filters (smoothing, edge detection), signal analysis, image processing, fast (multi)vector pattern matching, visual flow analysis, sampling, (multi)vector field analysis. GA FTs can be discretised and fast GA FT algorithms are available.

Next we introduce several types of socalled *Quaternion Fourier Transformations* (QFT). We explain their mutual relations, genuine 2D phase properties, their geometric transformation properties, discrete versions and fast numerical implementations. Applications include, partial differential systems, colour image processing, filtering, etc. Geometric Algebra relationships enable wide ranging higher dimensional generalizations. As an example we generalize to a *Spacetime Fourier Transform*, which naturally leads to *multivector wave packet analysis* in physics, and *directional uncertainty*, now with additional geometric insight.

While the GA FT is global, we introduce in the last part the *local GA wavelet* concept in the GA of real Euclidean 2-space and Euclidean 3-space, using 2D (3D) translations, dilations and rotations combined in the similitude group SIM(2), or SIM(3), respectively. Multivector mother wavelet functions need to fulfill the admissibility condition, which includes an admissibility constant with scalar and vector parts. We define the invertible *GA wavelet transformation* and discuss its main properties. An explicit example is the *GA Gabor multivector wavelet*. Generalizations to n = 2,3(mod 4) and in $Cl_{0,n}$ to n = 1,2 (mod 4) are straight forward.

ACKNOWLEDGMENTS

Soli deo gloria. I do thank my dear family, B. Mawardi, A. Hayashi, D. Hildenbrand, V. Skala.

REFERENCES

[Hitz07] Hitzer, E. Quaternion Fourier Transform on Quatern. Fields and Generalizations, Adv. in Appl. Clifford Algebras, 17(3), pp. 497-517 (2007).

[Hitz08] Hitzer, E., and Mawardi, B. Clifford Fourier Transform on Multivector Fields and Uncertainty Principles for Dimensions n = 2 (mod 4) and n = 3 (mod 4), in P. Angles (ed.), Adv. App. Cliff. Alg. Vol. 18(S3,4), pp. 715-736 (2008).

[Hitz09] Hitzer, E. Real Clifford Algebra $Cl_{n,0}$, n = 2,3(mod 4) Wavelet Transform, submitted to T. E. Simos et al., (eds.), Proc. of ICNAAM 2009.

[Maw07] Mawardi, B. and Hitzer, E. Clifford Algebra Cl(3,0)-valued Wavelet Transf., Clifford Wavelet Uncertainty Inequality and Clifford Gabor Wavelets, Int. J. of Wavelets, Multires. and Inf. Proc., 5(6), pp. 997-1019 (2007).

Geometric Algebra Computers

Dietmar Hildenbrand
TU Darmstadt, Germany
dhilden@gris.informatik.tu-darmstadt.de

ABSTRACT

In this paper, we investigate computers suitable for geometric algebra algorithms. While these *geometric algebra computers* are working in parallel, the algorithms can be described on a high level without thinking about how to parallelize them. In this context two recent developments are important. On one hand, there is a recent development of geometric algebra to an easy handling of engineering applications, especially in computer graphics, computer vision and robotics. On the other hand, there is a recent development of computer platforms from single processors to parallel computing platforms which are able to handle the high dimensional multivectors of geometric algebra in a better way.
Geometric algebra covers a lot of other mathematical systems like vector algebra, complex numbers, Plücker coordinates, quaternions etc. and it is geometrically intuitive to work with. Furthermore there is a lot of potential for optimization and parallelization. We present our geometric algebra compilation approach for current and future hardware platforms like reconfigurable hardware, multi-core architectures as well as modern GPGPUs.

Keywords: Geometric algebra, GPGPU, multi-core-architecture, Verilog, OpenCL, CUDA, OpenMP, Ct, Larrabee.

1 EXTENDED ABSTRACT

The foundation of geometric algebra was laid in 1844 by Hermann Grassmann whose 200th birthday we are celebrating this year. His work was continued by the English mathematician W. K. Clifford in 1878. Due to the early death of Clifford, the vector analysis of Gibbs and Heaviside dominated most of the 20th century, and not the geometric algebra. Geometric algebra has found its way into many areas of science, since David Hestenes treated the subject in the '60s. In particular, his aim was to find a unified language for mathematics, and he went about to show the advantages that could be gained by using geometric algebra in many areas of physics and geometry. Many other researchers followed and showed that applying geometric algebra in their field of research can be advantageous, e.g. in engineering areas like computer graphics, computer vision and robotics. During the past decades, especially from 1986 until 2002, processor performance doubled every 18 months. Currently, this improvement law is no longer valid because of technical limitations. Now, we can recognize a shift to parallel systems and most likely these systems will dominate the future. Thanks to multi-core architectures or powerful graphics boards for instance based on the CUDA technology from NVIDIA or on the future Larrabee technology of INTEL, one can expect impressive results using the powerful language of geometric algebra.

There is already a very advanced pure software solution called Gaigen as well as some pure hardware solutions geometric algebra algorithms (see a survey in [2]).

We propose to combine the advantages of both software and hardware solutions. We use a two-stage compilation approach for geometric algebra algorithms. In a first step we optimize geometric algebra algorithms with the help of symbolic computing. This kind of optimization results in very basic algorithms leading to highly efficient software implementations. These algorithms, foster a highly degree of parallelization which are then used for hardware optimizations in a second step. As examples for geometric algebra computers we present

- a FPGA(field programmable gate array) implementation of an inverse kinematics algorithm. This solution is about 300 times faster [1] (3 times by software optimization and 100 times by additional hardware optimization) than the conventional software implementation.
- examples on how to implement geometric algebra algorithms on multi-core architectures. Multivectors of a n-dimensional geometric algebra are 2^n-dimensional. Since all of their coefficients can be computed in parallel, arbitrary geometric products benefit a lot from highly parallel structures.
- a OpenCL/CUDA implementation for arbitrary geometric products.

REFERENCES

[1] Dietmar Hildenbrand, Holger Lange, Florian Stock, and Andreas Koch. Efficient inverse kinematics algorithm based on conformal geometric algebra using reconfigurable hardware. In *GRAPP conference Madeira*, 2008.

[2] Dietmar Hildenbrand, Joachim Pitt, and Andreas Koch. Gaalop - high performance parallel computing based on conformal geometric algebra. In Eduardo Bayro-Corrochano and Gerald Scheuermann, editors, *Geometric Algebra Computing for Engineering and Computer Science*. Springer, 2009.

Determining an Even Versor in n-D Geometric Algebra from the Known Transformation of n Vectors

Leo Dorst

University of Amsterdam, The Netherlands, L.Dorst@uva.nl

ABSTRACT

Suppose we only know of some elements how an even versor transforms them, how do we then solve for the unknown versor V? We present a method that works in n-D geometric algebra for n exactly known vector correspondences.

The method is based on the definition of a quantity D_k, defined as the sum of reconstructions of the projections of the versor V on basis k-blades:

$$D_k = \sum_{\binom{n}{k} \text{ basis } k\text{-blades } X} X^r \, (X \rfloor V)/V,$$

where X^r denotes the reciprocal of X on the chosen basis.

- On the one hand, we demonstrate that this quantity D_k can be computed from the data for any k, when one knows the correspondence " x_i goes to x_i' " between n independent vectors ($i = 1, \cdots, n$).

  The crucial step is to show that any k-blade X in the expression $(X \rfloor V)/V$ is expressible as a weighted sum of geometric products of the x_i and x_i'. To be specific, for $k = 1$ this is:

  $$(x_i \rfloor V)/V = \tfrac{1}{2}(x_i V - V x_i)/V = \tfrac{1}{2}(x_i - x_i'),$$

  in which we recognize the *chord* x_i under the transformation; for $k = 2$ it is:

  $$((x_i \wedge x_j) \rfloor V)/V = \tfrac{1}{4}(x_i x_j - x_i x_j' + x_j x_i' - x_j' x_i').$$

  The computability of D_k for other k is demonstrated by giving an iterative recipe.

- On the other hand, D_k can be shown to be directly related to weighted sum of the parts of various grade V_K of the versor V:

  $$D_k = \sum_{K=0,2,\cdots}^{n} \binom{K}{k} V_K/V,$$

  using result (2.41) from [4].

We demonstrate that from these equivalences, one can always construct a set of independent linear equations to find the V_K/V from the measured correspondences, and with that exactly determine the original versor V (modulo an irrelevant scale) through:

$$V/V_0 = (1 - \sum_K V_K/V)^{-1}.$$

This method leads to a closed form solution for V/V_0 for any dimension n. As such, it is more structural than an *ad hoc* method for low-dimensional cases in [4] (their pg. 111).

We show that our method includes the useful special case (from [6]) that the rotor in $\mathbb{R}^3$ rotating a frame $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ to a frame $(\mathbf{e}_1', \mathbf{e}_2', \mathbf{e}_3')$ is proportional to $1 + \sum_i \mathbf{e}_i' \mathbf{e}_i^r$ (where the superscript r denotes the reciprocal).

But we can apply the method much beyond that: in the conformal model of geometric algebra for instance [1, 3, 5], it can be used to determine a Euclidean motion in 3-D from a frame correspondence. The required 5 conformal vector correspondences are then: the frame location, the three frame directions, and the invariance of the point at infinity. We prove that this rigid body motion versor is proportional to $8 - 5\widetilde{D}_1 + 2\widetilde{D}_2$, or equivalently to $1 - \widetilde{D}_2 + 5\widetilde{D}_4$ (among several alternative forms).

However, the method can not be applied to determine an arbitrary conformal transformation from point and frame data alone, since crucial weight information on the versor transformation is then lacking. In that case, one should use [2] instead (which incidentally uses our n-D Euclidean result for part of its computation).

Keywords: geometric algebra, versor, correspondence data, conformal model, Euclidean motion, rigid body motion

REFERENCES

[1] P. Anglès, *Conformal Groups in Geometry and Spin Structures*, PMP 50, Birkhäuser, 2008.

[2] C. Cibura, L. Dorst, *From Exact Correspondence Data to Conformal Transformations in Closed Form Using Vahlen Matrices*, GRAVISMA 2009.

[3] L. Dorst, D. Fontijne, S. Mann, *Geometric Algebra for Computer Science*, Morgan Kaufmann, 2007.

[4] D. Hestenes, G. Sobczyk, *Clifford Algebra to Geometric Calculus*,

[5] D. Hestenes, A. Rockwood, H. Li, *System for encoding and manipulating models of objects*, U.S. Patent 6,853,964, granted February 8, 2005

[6] J. Lasenby, A.N. Lasenby, C.J.L. Doran, W.J. Fitzgerald, *New Geometric Methods for Computer Vision: an application to structure and motion estimation*, IJCV 36(3), pp.191-213, 1998.

Exponentials and Motions in Geometric Algebra

Leon Simpson
Department of Mechanical Engineering
University of Bath
Bath BA2 7AY, United Kingdom
l.c.simpson@bath.ac.uk

Glen Mullineux
Department of Mechanical Engineering
University of Bath
Bath BA2 7AY, United Kingdom
g.mullineux@bath.ac.uk

ABSTRACT

The use of geometric algebra to generate rigid body motions is investigated. In particular, means for handling the exponential function within the algebra are presented and used to create smooth motions.

Keywords

Geometric algebra, exponential function, smooth motion, interpolation.

1. EXPONENTIALS AND MOTIONS

The ideas of geometric (Clifford) algebras have been known since the 1800's. However they fell into disuse, and interest was only regained in the last ten years or so. Now they are used in a number of areas, including quantum physics and computer vision. One particular application is in describing smooth motions, such as the kinematics of robotic systems and mechanisms [Etz99]. In this context, and in others, the use of geometric algebra is closely related to the use of quaternions and dual quaternions [Pur08].

A consequently relevant question is whether it is possible to use geometric algebra to represent rigid body transforms (translations and rotations) in the same form. This has been shown to be possible [Mul04] provided that the square of one of the basis vectors is either infinite or zero. The former approach is effected by treating infinity as the reciprocal of a small real scalar which is allowed to become vanishingly small. The geometry thus modelled is a projective geometry, in this case RP^3.

One of the reasons why quaternions, in particular, are now used regularly for rotations in computer graphics is the ability to interpolate using them in equal steps [Sho85]. This has given rise to the construction known as the *slerp*, which effectively makes use of the exponential function. Exponentials can also be used in a similar fashion in conformal geometric algebra [War08] and with matrix representations of transforms [Ozg07].

The aim of this paper is to investigate the use of the exponential function in the geometric algebra model of projective geometry, and in particular how to deal with the vanishingly small terms.

The exponential is constructed in two equivalent ways. The first makes use of an isomorphic representation of the algebra as matrices over the quaternions. The second is a direct application of the power series, making allowance for the fact that more than one even grade element can induce the same rigid body transform. Use is made of Chasles's theorem to obtain the logarithm function.

Finally, the use of the exponential function to obtain smooth motions interpolating points is described.

2. REFERENCES

[Etz99] Etzel, K. R. and McCarthy, J. M. Interpolation of spatial displacements using the Clifford algebra of E^4. Journal of Mechanical Design, 121, pp. 39-44, 1999.

[Mul04] Mullineux, G. Modelling spatial displacements using Clifford algebra. Journal of Mechanical Design, 126, pp. 420-424, 2004.

[Ozg07] Özgören, M. K. Kinematics analysis of spatial mechanical systems using exponential rotation matrices. Journal of Mechanical Design, 129, pp. 1144-1152, 2007.

[Pur08] Purwar, A., Jin, Z. and Ge, Q. J. Rational motion interpolation under kinematic constraints of spherical 6R closed chains. Journal of Mechanical Design, 130, pp 062301:1-9, 2008.

[Sho85] Shoemake, K. Animating rotation with quaternion curves. ACM Siggraph, 19, pp. 245-254, 1985.

[War08] Wareham, R. and Lasenby, J. Mesh vertex pose and position interpolation using geometric algebra. In: Articulated Motion and Deformable Objects, Springer, pp. 122-131, 2008.

From Exact Correspondence Data to Conformal Transformations in Closed Form Using Vahlen Matrices

Carsten Cibura
Universiteit van Amsterdam
C.Cibura@uva.nl

Leo Dorst
Universiteit van Amsterdam
L.Dorst@uva.nl

ABSTRACT

1 ABSTRACT

In the present paper we will derive a method to determine a conformal transformation in 3D in closed form given exact correspondences between data. We will show that a minimal dataset needed for correspondence in arbitrary-dimensional Euclidean space is a localized vector frame and an additional point.

In order to determine the conformal transformation we will use the representation of the conformal model of geometric algebra by Vahlen matrices [1].

Conformal transformations are locally angle-preserving. They map circles to circles and are closely related to analytic - that is (complex) differentiable - functions. Classically, they are studied in the complex plane, which somewhat restricts their applicability to complex-valued or two-dimensional problems. However, there is a framework that seems naturally suited for conformal transformations in arbitrary dimensions: the conformal model of geometric algebra (CGA).

First and foremost it employs geometric algebra, which introduces the invertible but non-commuting *geometric product* between vectors in order to establish a graded algebra over a given vector space. Secondly, it enhances a Euclidean vector space $\mathbb{R}^n$ of arbitrary dimension n to a Minkowski space $\mathbb{R}^{n+1,1}$ of dimension $n+2$ and signature $(n+1,1)$. As a result vectors and multivectors from that geometric algebra $\mathscr{G}(\mathbb{R}^{n+1,1})$ can be thought of as representing geometric objects in the n-dimensional Euclidean space. Moreover, certain elements of CGA, so-called even *versors*, can be used to perform conformal transformations on the objects represented in this way.

More specifically, given an object x and an even versor s, the conformally transformed object x' is given by $x' = sxs^{-1}$. If, in this equation, we treat the versor s as a variable to be determined from correspondence data x and x', we encounter a number of complications. First of all, while the geometric product is linear, the Minkowski signature of the base space makes straightforward application of well-established methods from linear algebra impossible. Secondly, since the geometric product is not commutative, the two-sided versor product makes it difficult to isolate s and solve for it. Also, the higher dimension of the enhanced vector space slightly increases the complexity of the problem.

Crucial insights on how to solve this problem are gained from the study of Möbius transformations and Vahlen matrices [3]. The most general conformal transformation can be expressed as the most general *Möbius transformation* acting on points by *fractional linear transformation*. I.e. in the complex case, it can be expressed as $\mathbb{C} \ni \mathbf{z} \mapsto \mathbf{z}' = \frac{a\mathbf{z}+b}{c\mathbf{z}+d} \in \mathbb{C}$ with a, b, c, d being complex valued constants. This makes the transformations accessible by matrix algebra. Enhancing a complex number by homogeneous coordinates, we can represent $z \simeq \binom{\mathbf{z}}{1}$ and find $z' \simeq \binom{\mathbf{z}'}{1} \simeq \binom{a\mathbf{z}+b}{c\mathbf{z}+d} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \binom{\mathbf{z}}{1}$. The matrix $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ representing the Möbius transformation - i.e. fulfilling a number of constraints - with entries $a, b, c, d \in \mathbb{C}$ is called a *Vahlen matrix*.

The concept of Vahlen matrices can be extended to CGA. Then the versor equation can be expressed in matrix notation. By introducing intermediate transformations with very simple matrix representations and enforcing the algebraic conditions for a Vahlen matrix we can reduce the problem to finding a Euclidean rotor that aligns two given vector frames by pure rotation.

Thus, the result is a reversion to the n-dimensional Euclidean base space and a decomposition of the necessary equations into geometrically meaningful parts. By this we are able to present an algorithm for a closed form solution for the even versor s given data correspondences x and x' as in the equation above. A counting argument on the degrees of freedom present in the correspondence data and the generality of the formulas used suggests that our method is applicable in arbitrary dimensions n employing methods introduced in [2].

Keywords: Geometric Algebra, Versors, Closed-Form Solution, Conformal Mappings, Conformal Transformations, Differentiable, Vahlen

REFERENCES

[1] Leo Dorst. Conformal geometric algebra by extended vahlen matrices. In *GraVisMa*, 2009.

[2] Leo Dorst. Determining an even versor in n-d geometric algebra from the known transformation of n vectors. In *GraVisMa*, 2009.

[3] Udo Hertrich-Jeromin. *Introduction to Möbius Differential Geometry*. Cambridge University Press, 2003.

A Monogenic Signal on the Two-Sphere with Applications in Omnidirectional Vision

Oliver Fleischmann
Cognitive Systems Group
University of Kiel, Germany
ofl@ks.informatik.uni-kiel.de

Lennart Wietzke
Cognitive Systems Group
University of Kiel, Germany
lw@ks.informatik.uni-kiel.de

Gerald Sommer
Cognitive Systems Group
University of Kiel, Germany
gs@ks.informatik.uni-kiel.de

ABSTRACT

The analytic signal and its two-dimensional extension, the monogenic signal, are valuable representations in signal processing and are able to detect the features local phase, local orientation and local amplitude. They are based on the Hilbert transform on the real line and the Riesz transform in the upper half plane. Problems in computer vision require the analysis of spherical signals but so far no analogue to the monogenic signal exists on the sphere. We use the Hilbert transform on the two-sphere known from Clifford analysis to construct a monogenic signal on the two-sphere and a corresponding Poisson scale space. Its interpretation is given by an expansion into spherical harmonics which allows the rotation invariant detection of the features local orientation, local phase and local amplitude on the two sphere. We provide a scale invariant feature detector on the two-sphere in terms of the new spherical monogenic signal and apply it to omnidirectional images obtained by catadioptric cameras.

Keywords: Feature detection, Scale invariant features, Spherical signals, Spherical convolution, Clifford analysis, Scale space, Cauchy transform, Hilbert transform

1 INTRODUCTION

An important task in low-level signal processing is the detection of certain features, depending on a chosen signal model. In one-dimensional signal processing the properties local phase and local amplitude of a sinuoid signal are valuable features that describe the signal structure. They are obtained by the analytic signal [3], which is based on the Hilbert transform of the signal. The extension of the analytic signal to the Euclidean plane is known as the monogenic signal [2]. The underlying signal model are intrinsically one-dimensional signals in the plane. In addition to the features local amplitude and local phase it is able to determine the local orientation of these signals. Instead of relying on the one-dimensional Hilbert transform, it is based on its two-dimensional extension also known as the Riesz transform. Computer vision tasks often require the processing of spherical input signals like omnidirectional images. It is therefore desirable to construct a monogenic signal on the two-sphere. We construct a monogenic signal on the sphere with the help of the Hilbert transform on the two-sphere known from Clifford analysis [1].

2 PRELIMINARIES

In the Clifford analysis setting, the classical Hilbert transform arises as the non-tangential boundary value of the Cauchy transform in the upper half plane with the real line as its boundary. This concept can be extended to the upper half space $\mathbb{R}^n_+$ where it is known as the Riesz transform. In the Fourier domain they act as multipliers which allow an intuitive interpretation in the frequency space. Furthermore the Cauchy transforms naturally provides a scale space concept, the Poisson scale space. The Hilbert transform on the two-sphere can also be defined as the non-tangential boundary value of the Cauchy transform in the unit ball. Nonetheless, in its convolution integral nature, it prohibits an intuitive interpretation.

3 OUR WORK

We interpret the Hilbert transform as a group convolution over the rotation group $SO(3)$ and expand it into a Fourier series. We obtain a Fourier series expansion of the convolution kernels in terms of its spherical harmonic coefficients. It turns out that the components of the Hilbert transform on the two-sphere act as angular derivative operators on the solution of the Dirichlet problem in the unit ball with respect to the azimuthal and zenithal angles. Therefore we obtain, in analogue to Gaussian derivatives which arise as derivatives of the heat equation solution, derivatives in the Poisson scale space concept. The resulting features that can be obtained by this model are local orientation, local phase and local amplitude of the signal projected to plane tangent to the sphere at the point of interest. We construct a feature detector on the two-sphere. Instead of the Gaussian scale space we use the Poisson scale space which arises naturally from the Cauchy transform. Possible applications are image matching and object detection in omnidirectional images.

REFERENCES

[1] R. Delanghe. On some properties of the Hilbert transform in Euclidean space. *Bull. Belg. Math. Soc. Simon Stevin*, 11(2):163–180, 2004.

[2] M. Felsberg and G. Sommer. The monogenic signal. *IEEE Transactions on Signal Processing*, 49(12):3136–3144, 2001.

[3] D. Gabor. Theory of communication. *Journal of the IEE (London)*, 93:429–457, 1946.

Rigid body motion and conformal geometric algebra

Anthony Lasenby
University of Cambridge
Astrophysics Group
Cavendish Laboratory
Cambridge, CB3 0HE, U.K.
a.n.lasenby@mrao.cam.ac.uk

Chris Doran
University of Cambridge
and Geomerics Ltd.
Sidney Sussex College
Cambridge, U.K.
chris.doran@geomerics.com

Robert Lasenby
University of Cambridge
St. John's College
Cambridge
CB2 1TP, U.K.
rnl22@cam.ac.uk

Keywords
Conformal geometric algebra; rigid body dynamics; fast frictional dynamics.

EXTENDED ABSTRACT

1 INTRODUCTION

Rigid body dynamics is a still-thriving field of classical mechanics. Modern applications include not just real-world objects, but objects inside a computer. The ability to be able to calculate rigid body motion quickly and efficiently is important, for example, in computer graphics and gaming. In the extension to the case where rigid bodies interact via contact forces and friction, very sophisticated effects are possible if many bodies and their interactions can be treated simultaneously via an underlying fast method.

The idea of the work described here is to look at the Conformal Geometric Algebra (CGA) approach to rigid body dynamics, where we work in an overall space that is 2 dimensions higher than the base 3d Euclidean space. The CGA as a tool for geometry was first introduced by David Hestenes [Hes01], and has been developed by several others (e.g. [Las05, DL03]). Likewise the application to rigid bodies has been pursued by many. However, in our view there are still some gaps in this which mean that it is useful to attempt a coherent unified treatment, based on a Lagrangian formulation, and with the idea of *covariance* under rigid translations and rotations at its heart. Furthermore, the application is not limited to non-interacting rigid bodies, but seeks to parallel the fast treatment of multiple interacting rigid bodies developed in [KEP05] known as 'Fast Frictional Dynamics' (FFD).

Our setup has three positive square basis vectors, e_1, e_2, e_3, with two extra vectors adjoined, e and $\bar{e}$, satisfying $e^2 = +1$, $\bar{e}^2 = -1$. (See [DL03] for notation, which differs from that in [Hes01].) From these we form the null vectors $n = e + \bar{e}$ and $\bar{n} = e - \bar{e}$. The representation function we will use has the origin as $-\bar{n}/2$, i.e.

$$X = \frac{1}{2\lambda^2}\left(x^2 n + 2\lambda x - \lambda^2 \bar{n}\right)$$

where x is the ordinary position vector in Euclidean 3-space, and λ is a positive scalar giving the dimensions of length.

The overall idea is to set up a Lagrangian which is covariant with respect to the 5d geometry, but for which the energy is just the ordinary 3d rigid body energy. Then we should get equations of motion which are correct at the 3d level, but covariantly expressed in 5d. Additionally, the way we will express the current state of the rigid body configuration is via a combined rotation/translation rotor, so that translations and rotations are integrated as much as possible. Expressing position X in space in terms of position on a reference body X_{ref}, via $X = \psi X_{\text{ref}} \tilde{\psi}$, where ψ is a 5d spinor, we find that the correct Lagrangian is the scalar part of

$$-\frac{1}{2}\Omega_B \cdot I(\Omega_B) - \mu(\psi\tilde{\psi} - 1) - IU\psi\tilde{\psi} - V(\psi n \tilde{\psi} - n)$$

where μ is a scalar, I the 5d pseudoscalar, U and V are general 5d vectors appearing as Lagrange multipliers and $I(\Omega_B)$ is the 5d 'inertia tensor', which is a function of the body angular velocity (again in 5d) Ω_B.

From these beginnings, we show how motion under gravity can be incorporated, and then consider the effects of external forces and torques, which are unified within the 5d representation. Interacting bodies are then considered, and some details given of adapting these methods to parallel the FFD development of [KEP05]. Finally video demonstrations showing how the motions of large numbers of interacting bodies can be computed successfully in real-time in this approach will be given.

REFERENCES

[DL03] C. Doran and A.N. Lasenby. *Geometric Algebra for Physicists*. Cambridge University Press, 2003. Paperback edition, 2007.

[Hes01] D. Hestenes. Old wine in new bottles: a new algebraic framework for computational geometry. In E. Bayro-Corrochano and G. Sobczyk, editors, *Geometric Algebra with Applications in Science and Engineering*, page 3. Birkauser, Boston, 2001.

[KEP05] Danny M. Kaufman, Timothy Edmunds, and Dinesh K. Pai. Fast frictional dynamics for rigid bodies. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pages 946–956, New York, NY, USA, 2005. ACM.

[Las05] A.N. Lasenby. Recent applications of conformal geometric algebra. In H. Li, P.J. Olver, and G. Sommer, editors, *Computer Algebra and Geometric Algebra with Applications (Lecture Notes in Computer Science)*, page 298. Springer, Berlin, 2005.

Precise Image Resampling Algorithm

Pavel Zemcik
Faculty of Information Technology
Brno University of Technology
Bozetechova 2
CZ 612 66, Brno, Czech Republic
zemcik@fit.vutbr.cz

Bronislav Pribyl
Faculty of Information Technology
Brno University of Technology
Bozetechova 2
CZ 612 66, Brno, Czech Republic
xpriby12@stud.fit.vutbr.cz

ABSTRACT

This contribution introduces a precise image resampling image intended for corrections of image distortions caused by lenses or similar devices. The algorithm is intended for correction of small distortions in terms of pixel displacement but with very high subpixel precision. The geometrical description of the correction is through a square mesh with a displacement specified for each node of the mesh and bi-linear position interpolation is used inside the squares. The contribution describes the algorithms itself, its features, implementation issues and data formats and also discusses implementation issues specifically for programmable hardware (FPGA).

Keywords

Image resampling, lens distortion, subpixel resampling.

1. INTRODUCTION

Image acquisition processes are often affected through imperfections of lenses or other image acquisition equipment. The geometry of such image distortions is often known or can be measured. In some applications, the knowledge of the distortion can be used at some point of the algorithm fro compensation, in other algorithms this approach is difficult or impossible and it is necessary to "correct" the raster image. The correction process, however, is relatively difficult to define and sometimes it can be time-consuming. This contribution introduces a precise subpixel resampling algorithm that is possible to implement in software and also in programmable hardware, such as FPGA. The full contribution will describe more application and mathematics background and literature references.

2. RESAMPLING METHOD

The resampling method introduced here is intended for small changes in the image geometry. Therefore, angular changes in the image are not expected. The resampling is performed through application of 2D FIR convolution core. For performance reasons, only separable FIR convolution cores are considered. The convolution cores are selected for each output pixel separately (from a table) based on the subpixel position of the center of pixel in the original iamge. While this approach presents a small limitation in resampling core features, it allows for much higher implementation efficiency comparing to the cores that are not separable. The maximum considered resampling core is 7x7 samples. While this is not critical in the principle, the size limit is reasonable from the point of view of preserving the frequency features of the resulting image (limitation of maximum spatial frequency) and also from the point of implementation.

The actual contribution will introduce description of the convolution cores used and their features as well as the mathematics background.

3. IMPLEMENTATION ISSUES

Implementation issues of the presented algorithm are critical from the point of view of performance and precision.

First, the discussion of the pixel formats will be shown. The pixel formats (bit width) has an important effect on the necessary precision of the intermediate values on the algorithm. Next, the precision of the pixel co-ordinates will be given. This is important from the point of view of FIR core selection. Also, the overall efficient structure of the algorithm and its possible block diagram will be shown. Finally, the results of FPGA implementation of the algorithm will be shown. The current status indicates that implementation in FPGA capable of processing of 50 MPixels/s or more is possible.

4. CONCLUSIONS

In the conclusions, the contribution will summarize the results and applications along with the intended future implementation work. Additionally, the literature references will be included.

Mining and Visualizing Web Logs for Users Behavior Analysis

Virginijus Marcinkevčius
Institute of Mathematics and Informatics
Akademijos str. 4
LT-08663 Vilnius, Lithuania
VirgisM@ktl.mii.lt

Viktor Medvedev
Institute of Mathematics and Informatics
Akademijos str. 4
LT-08663 Vilnius, Lithuania
Viktor.m@ktl.mii.lt

Keywords

Web Logs, Data Mining, Visualization, Users Behavior.

EXTENDED ABSTRACT

1. INTRODUCTION

Web sites access logs contain substantial data about user access patterns. Properly exploited, the logs can reveal useful information about each user's interests in a site; but restructuring a site's structure to individual user interests increases the computation at the Web server to an impractical degree. One way to solve this problem is to group users on the basis of their Web interests and then organize the site's structure according to the needs of different groups. The visualization methods play an important role in exploratory data analysis, where visual representations can help to build up an understanding of the content [Kei02]. Most of the currently availably Web site analysis tools provide only explicitly and statistical information without real useful knowledge.

Knowledge discovery in databases (KDD) process can be used for this purpose. KDD process consists of two main phases: Web log preprocessing in order to get multidimensional data and data mining. In this paper for data mining we are using clustering, pattern discovery, classification methods to transform multidimensional data into suitable form for investigation. In this paper for investigation we use a data visualization methods to discover hidden relationships among the Web Logs data. The clustering and visualization algorithms (unsupervised learning algorithm Self Organizing Map (SOM), SAMANN) [Bri07, Med07] can be used to discover usage patterns from Web logs. The introduced methods perform clustering of input data and map it to a two-dimensional space, doing analysis and visualization at the same time. In our approach, with the Web usage data from high dimensional input data a two-dimensional projection (map) of Web usage patterns with different clusters could be formed

The main goal of this paper is to use these clusters to analyze visitors of the website and characterize or distinguish them in way to improve the structure of the website. We extracted some features from the web logs and used combined data mining methods (clustering and visualization) for this purpose.

The data used for the paper are the web pages and corresponding access logs of the Institute of Mathematics and Informatics: www.mii.lt

2. ACKNOWLEDGMENTS

The research is partially supported by the Lithuanian State Science and Studies Foundation project "Information technology tools of clinical decision support and citizens wellness for e.Health system" (No. B-07019).

3. REFERENCES

[Bri07] P. Britos, D. Martinelli, H. Merlino, and R. Garcia-Martinez. Web usage mining using self organized maps. *IJCSNS International Journal of Computer*

Science and Network Security, 7(6):45–50, 2007

[Kei02] D. A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.

[Med07] V. Medvedev. *Tiesioginio sklidimo neuroniniu tinklu taikymo daugiamaciams duomenims vizualizuoti tyrimai*. PhD thesis, Matematikos ir informatikos

institutas, 2007

About contour words of balls in tessellations of the hyperbolic plane

Maurice Margenstern
Université Paul Verlaine – Metz
LITA, EA 3097, UFR-MIM
Île du Saulcy, 57045 METZ Cédex, FRANCE
Email: margens@univ-metz.fr

Keywords
pushdown automata, iterated pushdown automata, tilings, hyperbolic plane, tessellations, contour words.

1. INTRODUCTION

In this paper, we give an application of iterated pushdown automata to contour words of balls in infinitely many tilings of the hyperbolic plane. We extend this result to the tiling $\{5, 3, 4\}$ of the hyperbolic $3D$ space and to the tiling $\{5, 3, 3, 4\}$ of the hyperbolic $4D$ space as well. These two latter applications cannot be generalized to any dimension as, starting from dimension 5, there is no tiling of the hyperbolic space which would be a tessellation generated by a regular polytope.

2. ITERATED PUSHDOWN AUTOMATA

Here, we follow the notations of [1] where references on iterated pushdown automata can be found. We refer to this paper and to [4] for formal definitions and an illustrative example on the language of the words 1^{f_n}, where f_n is the Fibonacci sequence with $f_0 = f_1 = 1$.

Intuitively, an iterated pushdown automaton is defined from the notion of iterated satck. An k+1-iterated stack is a stack of k-iterated stacks, where a 0-stack is reduced to the empty word. The i-stacks are labelled by the elements of a finite alphabet Γ, fixed in advance.

We are provided with a function *topsym* which gives the top of the stack and, recursively, as the content of the top is an iterated stack, the top of the just stored stack. Similarly, we can *pop* any element which is accessed by function *topsym* as well as *push* a new element on the tops accessed by the function.

A k-iterated pushdown automaton is a finite automaton which works with a k-iterated stack. Each time it reads a letter of the imput word, the automaton performs an operation, *pop* or *push* on the iterated stack, ϵ-transition being allowed. A word is accepted if, when completely read, the stack is empty. The recognized language is the set of accepted words.

3. COUNTOUR WORDS IN TESSALLATIONS OF THE HYPERBOLIC PLANE

We assume the reader to be familiar with hyperbolic geometry. We also refer him/her to [2], [3] for both an introduction to this field with suitable references and the definition of specific tools underlying the proofs of the result stated in the paper.

A **tessellation** is a tiling generated from a regular polygon by reflection in its sides and, recursively, of the images in their sides. In the Euclidean case, there are only three tessellations up to similarities. In the hyperbolic plane, there are infinitely many of them denoted by $\{p, q\}$, where p is the number of sides of the polygon and q is the number of them needed to exactly cover a small enough neighbour of any vertex. We shall focus on the tilings $\{p, 4\}$ and $\{p+2, 3\}$.

In this setting, a path joining two tiles P and Q is a finite sequence of tiles $\{T_i\}_{0 \leq i \leq n}$ with $T_0 = P$, $T_n = Q$ and T_i sharing a side with T_{i+1} for $0 \leq i < n$. The distance between the tiles is the length of a shortest path. A ball of radius k around a tile A is the set of tiles whose distance from A is at most k. Denote it by $B_k(A)$. Its border, $\partial B_k(A)$, is the set of tiles whose distance from A is k. Let w_k be the number of tiles on $\partial B_k(A)$. Then, a **contour word** of a ball in the tiling $\{p, 4\}$ or $\{p+2, 3\}$ is a word of the form 1^{w_k}.

Then, we can state:

Theorem 1: see [4] – *The language of the contour words of the balls in $\{p, 4\}$ and that of the balls in $\{p+2, 3\}$ can be recognized by a 2-iterated pushdown automaton. This is also the case for the balls in $\{5, 3, 4\}$, the tessellation of the hyperbolic $3D$ space by the rectangular dodecahedron, and also for the balls in $\{5, 3, 3, 4\}$, the tessellation of the hyperbolic $4D$ space by the 120-cell.*

ACKNOWLEDGMENT

The author would like to thank Professor Václav Skala for inviting him to present this work at **GraVisMa'2009**.

REFERENCES

[1] Fratani S., Sénizergues G., Iterated pushdown automata and sequences of rational numbers, *Annals of pure and applied logic*, **141**, (2006), 363-411.
[2] Margenstern M., Cellular Automata in Hyperbolic Spaces, Volume 1, Theory, *OCP*, Philadelphia, (2007), 422p.
[3] Margenstern M., Cellular Automata in Hyperbolic Spaces, Volume 2, Implementation and computations, *OCP*, Philadelphia, (2003), 360p.
[4] Margenstern M., Iterated pushdown automata and hyperbolic contour words, *arXiv*:0907.4957, (2009), 15p.

Hyperspectral Images: Compression and Visualization

Bruno Carpentieri
D.I.A. - Università di Salerno
84081 Fisciano (SA), Italia
bc@dia.unisa.it

Keywords
Hperspectral Images, Data Compression, Visualization

EXTENDED ABSTRACT

Remote acquisition of high definition electro-optic images has been increasingly used in military and civilian applications. These air-borne and space-borne acquired hyperspectral images are used to recognize objects and to classify materials on the surface of the earth. It is possible to recognize the materials pictured in the hyperspectral image by analyzing the spectrum of the reflected light. Hyperspectral detector technologies have made possible the recording a large number of spectral bands over the visible and reflected infrared region. These instruments have sufficient spectral resolution to allow an accurate characterization of the spectral reflectance curve of a given spatial area. As an example, the images acquired with the NASA JPL's Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) have pixels covering an area of approximately 20x20 meters, with reflectance decomposed into 224 narrow bands, approximately 10nm wide each, in the range 400 to 2,500nm, the spectral components are represented with a 16 bits precision.

Higher resolution, space-borne spectrometers will be available in the near future. In fact, increasing the number of bands, i.e. the spectral resolution, allows for more sophisticated analysis and increases the data rate by only a linear amount. The acquisition of hyperspectral images produces huge amounts of highly correlated data (e.g., in the range 140MB to 1GB for AVIRIS images) that are stored as a two dimensional image matrix where each pixel consists of many components, one for each spectral band (e.g., 224 16-bit values for a pixel in an AVIRIS image). This brings an important need for efficient compression. Since hyperspectral imagery is acquired at significant cost and often used in delicate tasks (for instance classification or target detection), compression algorithms that provide lossless or near lossless quality are generally required. Moreover it may be desirable to have low complexity that allows efficient on board implementation with limited hardware, when the compression task has to be carried out on an airplane or a satellite before transmitting the data to the base. Remote sensed images, like AVIRIS, show two forms of correlation: spatial (the same material tends to be present in many adjacent pixels) and spectral (one band can be fully or partly predicted from other bands).

For hyperspectral images the spectral correlation is much stronger than the spatial correlation. Furthermore, dynamic range and noise levels (instrument noise, reflection interference, aircraft movements, etc.) of AVIRIS data are higher than those in photographic images. For these reasons the standard image compression techniques, as for instance the spatial predictor of LOCO I, fail on this kind of data. Spectral oriented Least SQuares (SLSQ) is a lossless hyperspectral image compressor that uses least squares to optimize the predictor for each pixel and each band. In this paper we describe SLSQ and show experimental results with AVIRIS images. SLSQ is the current state of the art in hyperspectyral image compression.

Each band in an hyperspectral image can be seen as a two-dimensional image. By visualizing the content of each band it becomes clear that the spectral correlation is high between most of the bands, but it is also clear that a few consecutive bands are not strongly spectrally correlated. This can lead to improvements to the SLSQ algorithm if we take into account the possibility of using different predictors for different bands.

REFERENCES

[NASA] NASA, "AVIRIS home page," http://popo.jpl.nasa.gov , February 2003.

[Rizzo et al.] F. Rizzo, B. Carpentieri, G. Motta, and J. A. Storer, "Low Complexity Lossless Compression of Hyperspectral Imagery via Linear Prediction", IEEE Signal Processing Letters, Vol. 12, n.. 2, pp. 138-141, February 2005, ISSN:1070-9908.

Hypergraph-based software visualization

Peter Kapec

Faculty of Informatics and Information Technologies
Slovak University of Technology
Ilkovičova 3
842 16, Bratislava, Slovakia

kapec@fiit.stuba.sk

Keywords

Software visualization, hypergraphs, zoom interface

EXTENDED ABSTRACT

1. INTRODUCTION

In this paper an alternative approach to software visualization is presented based on hypergraph representation of software artifacts. Our approach builds upon known techniques that rely on graphs for visualization. Using hypergraph formalism offers significant advantages not only in the visualization process but also in data preparation, filtering and context retrieval. Our aim is to create a unified graphical environment that is capable to visualize relations between various levels of software artifacts, ranging from source code up to the project management tasks.

2. SOFTWARE VISUALIZATION

Software visualization is an ongoing research direction. Although in past twenty-thirty years many software visualization systems were created, current software engineers program in textual languages and use standard 2D GUI interfaces. Several fully visual programming languages have been developed, but their usability in practice is very limited and questionable. However the benefits of software visualization in software comprehension were already shown [Lew02]. Many software visualization systems use graphs to present software in a more understandable way. Our aim is to generalize the visualization process by using hypergraphs instead.

3. HYPERGRAPHS

Hypergraphs are generalized graphs in which an edge can connect more than two nodes [Aui02]. This property allows us to represent more complicated relations between several objects for which standard graphs would require several additional nodes and edges. In our approach nodes represent different software artifacts and edges represent different relations between them. Using hypergraphs as data repositories allows us to look at software as knowledge. This unified structure enables us to query relevant information and visualize results directly. Query syntax is also based on hypergraph notation and queries always produce hypergraphs as results.

4. VISUALIZATION

Software artifacts and their relations represented through hypergraphs can be visualized using common graph visualization techniques, because hypergraphs can be transformed into bipartite graphs. For the purpose of our context sensitive GUI we plan to use force directed layout algorithms [Coh92]. Instead of using graphical programming languages we use common text editors embedded in hypergraph nodes displayed in 3D space. The text editors are also positioned using the force directed layout algorithm and are displayed as billboards floating in space. This is more familiar for developers and allows us to use existing programming languages. Developers can quickly zoom out for overview, focus on nodes for relations and may edit code if needed – all in the spirit of information seeking mantra principle [Shn04]. [1]

5. REFERENCES

[Aui02] Auillans, e.a. A formal model for topic maps. ISWC '02, Springer-Verlag, pp. 69-83, 2002.

[Coh92] Cohen, F.R. A Framework for Dynamic Graph Drawing, Congressus Numerantium, vol. 42, pp 149-160, 1992.

[Lew02] Lewerentz, C., Simon, F. Metrics-based 3D Visualization of Large Object-Oriented Programs. Proc. VISSOFT'02, 2002.

[Shn04] Shneiderman, B., and Plaisant, C. Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition). Pearson Addison Wesley, 2004

[1] This work was partially supported by the grant VG 1/0822/08: Intelligent embedded systems.

THE BPT ALGORITHM (BRIANCHON – POINT – TRIANGLE) - DETECTING CONICAL CURVES IN RASTER GRAPHICS.

Krzysztof T. TYTKOWSKI
The Silesian University of Technology
Geometry and Engineering Graphics Centre – RJM-4
POLAND, 44-100 Gliwice, ul. Krzywoustego 7
krzysztof.tytkowski.@polsl.pl

Abstract

The paper introduces the new algorithm of detecting conical curves. The BPT algorithm was created basing on Brianchon's theorem on projective geometry. The algorithm also uses properties of raster graphics. 8 tangent lines

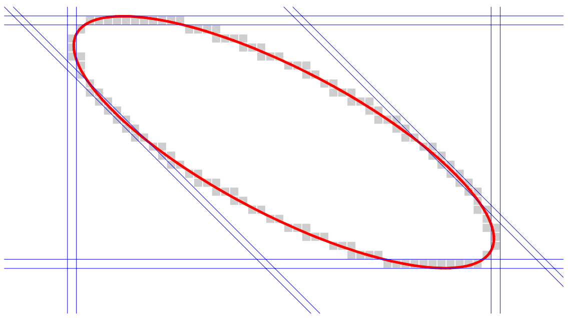


Fig. 1 Tangent lines to ellipse – exactness of 1 pixel

are found with 1 pixel precision (Fig.1): two horizontal and two vertical ones and two at the angle of 45^{o} . 6 out of 8 points are selected orderly (28 sets of tangents). Based on these 6 points acc. to Brianchon's theorem 3 lines are constructed. If these lines intersect in one point then the 6 lines are tangent to conical curve. If not, the value of the created triangle is the measure of deflection of these lines from tangent lines. For every 6 out of 8 tangent lines,

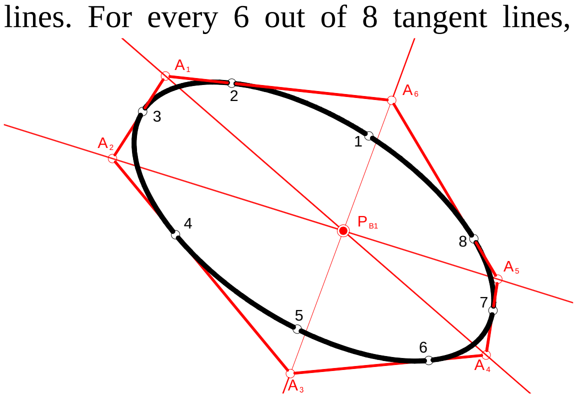


Fig. 2 Tangent point 234678

point P_{Bn} (Fig.2) is found.

Analyzing the remaining Brianchoin's triangles (Fig.3) it is possible to determine which of them are closer to the conical curve and which further away. Additional advantage of this method is a possibility of fast verification of obtained data as well as quick evaluation if the tangent lines are correctly drawn.

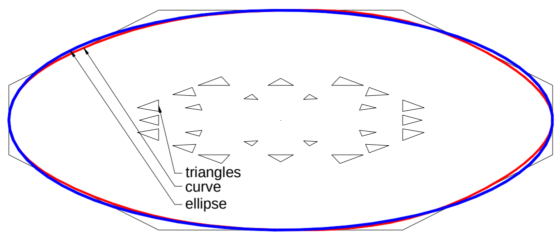


Fig. 3 Generation of Brianchon's triangles

Keywords: algorithm, Brianchon's theorem, conical curves, detection,

The Three Point Perspective Pose Estimation Problem revisited with invariants obtained from polynomial triangulation and interval analysis

François Anton
Technical University of Denmark
Department of Informatics and Mathematical Modelling
Richard Petersens Plads
2800 Kongens Lyngby, Denmark
fa@imm.dtu.dk

Henrik Aanæs
Technical University of Denmark
Department of Informatics and Mathematical Modelling
Richard Petersens Plads
2800 Kongens Lyngby, Denmark
haa@imm.dtu.dk

Fredrik Kahl
Lund University
Centre for Mathematics, Division of Mathematics,
PO Box 118, 221 00 Lund, Sweden
fredrik@maths.lth.se

Extended Abstract

A very common reference cited in many publications that deal with the three point perspective pose estimation problem is revisited in this paper. Our main contribution is first a methodology for automated derivation of invariants from the triangulation of the algebraic system defining the three point perspective pose estimation problem, then the application of this methodology to get an exhaustive analysis of the geometrical invariants that intervene in the three point perspective pose estimation problem, and finally it is the use of an hybrid methodology combining the triangularized system obtained by Wu's algorithm and interval analysis to get the actual solving of the three point perspective pose estimation problem in a concrete case.

Keywords
three point perspective pose, invariants, Gröbner bases, interval analysis, Wu's algorithm

1. INTRODUCTION

In this paper the three point perspective pose (finding the position of a camera from three known points) is revisited [HLON94]. Three point perspective pose is one of the fundamental problems of photogrammetry and computer vision, e.g. a cornerstone of robot navigation. To the best of our knowledge, there does not exist a comprehensive treatment of the geometrical invariants involved in the three point perspective pose estimation problem. Also, there does not seem to exist an automated methodology to get the geometrical invariants from a given system of algebraic equations. There have been some specialized methodologies to get the invariants for specific problems. Our main contribution is twofold: the automated derivation of geometrical invariants from a system of algebraic equations and the application of this methodology to the exhaustive analysis of the geometrical invariants of the three point perspective pose estimation problem.

2. METHODOLOGY

Starting from the system of equations defining the zero-dimensional algebraic set of the problem, we are following Wu's algorithm or Buchberger's algorithm to transform the initial system into an equivalent system such that at least one variable of the problem has been eliminated from the two polynomials having the lowest leading monomials in the graded reverse lexicographic order. By regrouping all the formal coefficients for each monomial, we get polynomials that are invariants for the given problem. We rewrite the original system by replacing the invariant polynomials by new formal coefficients. We repeat the process until the system has been triangularized. We apply it to the three point perspective pose estimation problem where the triangularized system is solved using interval analysis. We obtain the syzygies of the invariants by applying the Buchberger or Wu's algorithm on the invariants.

Posters

Forecasting Strategy Changes in Organizations

Dr. Sasipalli VSRao
Ramtej R&D Center (CECTech)
Takeyacho-7-20-203, Naka-Ku
Hiroshima, Japan 730-0048
Dr.SVSRao@cect.ramtej.org

ABSTRACT

Strategy is planned, executed and future changes forecasted. Organizations do several types of planning such as business plan, operational plan, and strategy plan but forecasting the changes in the strategy plan is not common or ignored. Scientific methodologies rarely are associated in decision making in mid-sized organizations as they do not have in-house R&D divisions. We propose forecasting methodologies using localization, term-plans concept and time series techniques to forecast the strategy changes for strategy planning in organizations. Simulations with the data from various organizations are provided in examples of analysis and forecasting. With these forecasting methodologies organizations can benefit greatly to optimize their investments and planning in the direction of current trend. Not only organizations but also planning divisions in government can benefit with these ideas and methodologies.

Keywords

Strategic plan, term-plan, time series analysis, forecasting

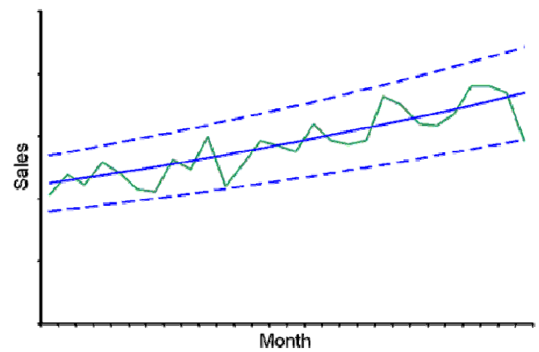
1. INTRODUCTION

Strategic plan is the highest level plan in any organization, which includes vision of the organization suitable to the current environment and business opportunity. Once done it shall be reviewed continuously based on the outcomes of various plans and projects that are derived to realize the targets of the strategic plan.

Forecasts are critical part of business planning, management and strategy, but they are always wrong. Business people need to realize that there is a certain error band around it which the statisticians know very well. Though it looks true that the forecasts are accurate if the actual outcomes fall within that error band, but for businesses the band is too wide which directly proportional to the investments.

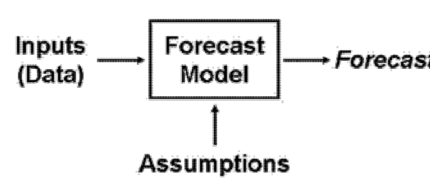
Usually after investing time and money in forecasting systems, business people develop overconfidence in the forecasts. A small example of sales can give (1) an idea of error range, (2) how the error range can be wide and (3) importance of forecasting the changes in targets.

In the sales prediction graph, green line is actual sales, blue solid line is regression line of forecast, and dotted lines are error band. Comparing the error band width at the first and last months, one can see how the variance is getting wide. This prompts to split the period into shorter periods and do the forecast and see the changes and finally forecast the global change.



2. Structure of Forecasting Models

It does not matter how accurate the trend is forecasted. Trend changes with various effecting parameters. It means that methodologies or forecasting models play an important role in forecasting the global change. Some forecasting models are discussed here. There are four types of forecasting models: Casual, Extrapolative, Intuitive, Hybrid. Casual or basic model:



Investigation of thermal anisotropy from thermovisual video data

Povilas Treigys
Institute of Mathematics and Informatics
Akademijos str. 4
LT-08663, Vilnius, Lithuania
Treigys@ktl.mii.lt

Gintautas Dzemyda
Institute of Mathematics and Informatics
Akademijos str. 4
LT-08663, Vilnius, Lithuania
Dzemyda@ktl.mii.lt

Vincentas Veikutis
Institute for Biomedical Research
Eiveniu str. 4
LT-50009, Kaunas, Lithuania
Vincentas.veikutis@med.kmu.lt

Keywords
Thermovision, heart tissue, video and image analysis, radiofrequency ablation

EXTENDED ABSTRACT

1. INTRODUCTION

Nowadays corrections of many cardiac disorders are treated by applying some destructive energy sources. One of the most common sources and the related methodology is to use radiofrequency ablations. In medical practice treatment capabilities have expanded almost to the whole area of the cardiac texture zones. However, despite the latest technical, navigational methods for localizing the affected zone, the rate of the risk of complications including a disease recurrence remains very high [Tan08]. Thus, in order to understand the nature of possible complications, the technique of thermovisual monitoring can be introduced. This technique is a non-invasive method and enables us to track the temperature changes over the time. Thermovisual data offer a possibility to use a wide range of analysis methods in terms of video or image processing. The object of the research proposed is to explore the impact of temperature on the heart texture anisotropy based on the thermovisual data. The pilot research has proved that the thermovisual data are rather promising for estimating the heart state [Lek09]. Also, these results have shown the necessity to extend the research. By applying the analysis of color models we have shown that not all the models had proper information about the affected zone. Investigation revealed that the most informative are L*ab* and CMYK color models, especially *a* and M bands. Also authors investigate the dependency parameters of the radiofrequency ablation (RFA) procedure by registering the dynamics of absorption and spread of heat on various cardiac structures over a certain time in a video stream. Thus, by incorporating the methods of mathematical morphology and transformations can approximate the thermal anisotropy zone. Moreover, that enables us to automatically identify, register, and track the changes in the structure as well as to evaluate the dynamics in time of the affected zone (see Fig. 1).

2. RESULTS

Further we performed the color mapping analysis. Principal component analysis of initial color mapping, as well as calculation of point Euclidian distances in L*ab* color space, exposed that the task comes to a two class separation problem, regardless of the manner in which the original color model is mapped to intensity interval [0 ... 255]. Conducted a preliminary analysis of the spread of risk area, according to electrode ablation power of 30 or 50 watts and the operation time, showed that the tissue thermal anisotropy dynamics, during heating and cooling stages remains the same.

Figure 1. Automated approximation of the RFA affect zone.

3. REFERENCES

[Tan08] Tan E., Rienstra M., Wiesfield A., et al. Long-term outcome of the atrioventricular node ablation and pacemaker implantation for symptomatic refractory atrial fibrillation. Europace 10(4): pp. 412–418, 2008.

[Lek09] Lekas R, Jakuska P., et. al. Monitoring changes in heart tissue temperature and evaluation of graft function after coronary artery bypass grafting surgery. Medicina 45(3): pp. 221-225, 2009.