Clifford Pattern Matching for Color Image Edge Detection

Michael Schlemmer, Hans Hagen
TU Kaiserslautern
Computer Science Department
D-67653 Kaiserslautern, Germany
{schlemmer||hagen}@informatik.uni-kl.de

Ingrid Hotz, Bernd Hamann
University of California, Davis
Institute for Data Analysis and Visualization
Davis, CA 95616
{ihotz||hamann}@ucdavis.edu

Abstract: Feature detection and pattern matching play an important role in visualization. Originally developed for images and scalar fields, pattern matching methods become increasingly interesting for other applications, e.g., vector fields. To apply pattern matching to vector fields the basic concepts of convolution and fast Fourier transform (FFT) have to be generalized to vector fields. A formalism supporting an elegant generalization of these concepts is provided by the Clifford Algebra, originally developed for describing geometry and geometric operations. We discuss an application of the Clifford Pattern Matching (CPM). We apply CPM to images for "Clifford Color Edge Detection" (C^2ED), an approach for detecting edges and other features in color images. The basic idea is to treat color value tripels as vectors and apply the pattern matching algorithm to the resulting vector field. We introduce vector-valued filters for edge detection and present results.

1 Introduction

Today, large amounts of data are produced by simulations and experiments. Pattern recognition methods have become more important as essential information is mostly hidden "in-between less important data". A good visualization should be capable of highlighting important facts, pointing out key features.

Recently, a new method for visualizing vector fields was introduced based on pattern matching methods. A convolution operator for pattern recognition was constructed and applied to uniform vector field data, see Heiberg et al. [HEWK03], and Ebling and Scheuermann [ES03]. The latter method is based on Clifford algebra.

In signal processing it is common to filter data in frequency domain, as the convolution operation is very expensive and is reduced to a multiplication in frequency domain. To devise a similar method for vector fields a continuous and discrete Fourier transform for multi-vector field data by using a Clifford algebra approach [Sch04, ES05] was adapted. *Multi-vectors* are elements of the Clifford algebra representing a combined set of complex-valued vector and scalar data. We implemented the discrete Clifford Fourier Transform

(CFT) using the FFT for regular grids.

Besides the application to flow vector data, there are various other possibilities. One of these options is the application of these vector methods to color image edge detection. Segmentation based on edge detection is a vital part of current classification systems. These systems are used in various application areas, including astronomy, medicine, robots, etc. In most cases, edge detection is performed on gray-scale images. Therefore, there are many common methods for this task, which are for example discussed in [Jai89], for examples. One of the most important techniques was developed by Canny [Can86], which is discussed in section 2.1.2.

There are also approaches for detection of edges using color information, i.e., the component-wise Canny edge detection applied to each of the RGB color channels. In order to improve color edge detection, we had the idea of regarding the RGB triple of colors as a vector, therefore the color image as vector field. Thus, we are able to apply our Clifford Fourier transform and Clifford convolution to find features.

As the vector field is given on a planar 2d domain, a 3d convolution leads to a 3d result, being inappropriate for our needs. We decided to stay in a 2d domain and consider only a 2d vector field. This could be achieved by switching the color model from RGB to YUV and treat the luminance component (Y) as scalar and the chrominance components (UV) as a 2d vector field. Machuca and Phillips [MP83] proposed similar settings for their method. We found that using any of the filter masks we use for representing special topological features (rotation, convergence, and divergence) allows one to detect edges in a color image. The 2d Clifford algebra is especially suitable for this task, as all components fit perfectly into the mathematical concept.

2 Edge detection

For computer vision one of the basic goals is the recognition of objects in complex scenes. Given an image in digital form, a first step towards segmentation and recognition of objects can be the detection of edges. Using this information a segmentation in for- and background can be performed, and semantics can be assigned through the process of classification.

2.1 Edge detection in grey-scale images

In practice, edge detection is performed on grey-scale images. We focus on the 2d case and define a 2d-image (see Jähne [Jäh95]):

Definition 1 *A 2d-image is a discrete scalar function* p *on a rectangular grid having* N_1 *grid points in x-direction,* N_2 *in y-direction with distances* $\Delta x, \Delta y$ *and value set* $\mathbb{W} \subseteq \mathbb{C}$*:*

$$p : \{0, 1\Delta x, ..., (N_1 - 1)\Delta x\} \times \{0, 1\Delta y, ..., (N_2 - 1)\Delta y\} \rightarrow \mathbb{W}.$$

For pixels p_{ij} *the following notation is used:*

$$r_{ij} = \binom{i\Delta x}{j\Delta y} \quad p_{ij} = p(r_{ij}).$$

A 2d-image is defined by the collection $P = (N_1, N_2, \Delta x, \Delta y, \mathbb{W}, \{p_{ij}\})$.

Each grey-scale value is assigned to a scalar number $w \in \mathbb{W}$.

2.1.1 Filter operations

There exist many different possibilities for filters for image processing. We concentrate on the class of linear-and-shift-invariant (LSI) filters as their impulse response can be described as a neighborhood representation. For application of LSI filters the performed operation can be either convolution or correlation. While the correlation operator directly performs a pattern matching to a given filter on the given signal, the (in practice more important) convolution uses a mirrored version of the filter. As the neighborhood representation of a filter can also be regarded as an image, one can define (see Jähne[Jäh95]):

Definition 2 *Let g be a 2d image, h be a 2d filter. The discrete convolution* $g * h$ *is defined as*

$$(g * h)_{m,n} = \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} h_{i,j} g_{m-i,n-j}.$$

Convolution is a computationally expensive operation. It is in practice simplified by transferring image and filter into frequency domain, where the convolution operation reduces to a multiplication. The result is then transformed back into spatial domain. As transformation between spatial and frequency domain the fast Fourier transform (FFT) can be used.

Through the application of filters different goals can be achieved. One special class of filters is used to smooth images to reduce various types of noise in an image. Examples for smoothing filters are box filters, binomial filters, and Gauss filters. Another class of filters was especially designed for edge detection. Those filters basically focus on gradients in an image. Examples for filters are the simple gradient filters (first order) and Laplace filter (second order). An optimized version of the gradient filter is the Sobel operator. The Marr-Hildreth operator [MH80] is an enhancement of the Laplace operator. It combines a noice-reducing Gauss filter with a Sobel edge detection filter.

2.1.2 Canny edge detection

Canny developed an optimal edge detector for grey-scale images [Can86]. His method is similar to the Marr-Hildreth approach, see [MH80]. First, a Gaussian is applied for smoothing, then the actual edge detection is performed. The Canny algorithm uses four gradient filter masks to detect horizontal, vertical and diagonal edges, as an edge in an image can be oriented arbitrarily. It is assumed that important edges form continuous traceable lines. The application of thresholding results in a binary image containing line segments that represent edges. The major problem of this approach is finding an appropriate threshold. Another parameter affecting the result is the size of the Gaussian blur filter applied in the beginning. With an appropriate choice of these parameters for a specific grey-scale image, ”optimal” results can be obtained.

2.2 Extensions to color edge detection

As most camera-recorded images contain colors, converting them to grey-scale images comes with a big loss of information. Therefore, several publications focus on the use of colors for segmentation and image understanding. The first color image edge detectors were presented by Robinson [Rob77] and Nevatia [Nev77]. Koschan applied the standard techniques to the three color channels and performed a comparative study [Kos95]. Tao and Huang used cluster analysis to determine better thresholds [TH97]. Most methods apply the well-known grey-scale methods to the three RGB color channels seperately. The most successful method is again the Canny color edge detection scheme using the Canny edge detector for the RGB channels. Furthermore, other authors have proposed the use of different color spaces, e.g., Weeks and Myler [WM95] as well as Machua and Phillips [MP83]. Among others, some of the mentioned methods have recently been reviewed by Koschan and Abidi [KA05].

3 Clifford algebra in two dimensions

Clifford algebra can be understood as an extension of complex numbers to vectors:

Definition 3 *Let $\mathbb{E}^2$ be the $\mathbb{R}$ vector space with basis $\{e_1, e_2\}$. The* ***Clifford algebra*** *$\mathcal{G}^2$ is the real 2^2-vector space with basis*

$$\{1, e_1, e_2, e_1 \wedge e_2\},$$

where

1. $1e_k = e_k,\ k = 1, 2,$
2. $e_k e_k = 1,\ k = 1, 2,$ *and*

3. $e_k e_l = -e_l e_k,\ k \neq l.$

The elements of a Clifford algebra are called multi-vectors. For the 2d case, they are shown in the following table:

name	grade	dimension	basiselements
scalar	0	1	1
vector	1	2	e_1, e_2
bivector	2	1	$e_1 e_2$

A multi-vector consists of a scalar, a 2d vector and the so-called bivector (pseudo-scalar). Regarding the Clifford algebra as an extension of complex numbers, it can be regarded as a tuple of two complex numbers, one having the scalar as real part and the bivector as imaginary part, the vector having the e_1 component as real and the e_2 component as imaginary part. For detailed information on Clifford algebra, see Hestenes and Sobczyk [HS99].

4 Pattern matching using Clifford algebra

A first definition of a convolution operator was proposed by Heiberg et al. [HEWK03]. It uses the inner (scalar) product for vectors. Ebling and Scheuermann [ES03] defined the convolution using the Clifford product consisting of inner and outer products. In addition, for efficient calculation a fast Fourier transform for Clifford algebra was introduced [Sch04, ES05]. These techniques enabled an image processing-like detection of patterns in vector fields. Patterns can be defined as vector valued LSI filters as neighborhood representation. Figure 1 illustrates the process of pattern matching applied to vector fields using vector-valued masks and Clifford convolution.

5 Clifford color edge detection

5.1 General idea

As color images can be represented as vector fields using the color components as vector components, the general idea of Clifford color edge detection is using pattern matching for vector fields for the detection of edges in color images. The given 3d vectors (RGB) are on a 2d grid. The application of 3d pattern matching would lead to a 3d similarity map as result, being an inappropriate representation as our result has to be projected back into two dimensions. For that reason, we decided to handle the luminance and chrominace part separately. This can be achieved by transferring an RGB image to a corresponding YUV image. The Y channel represents the luminance (grey-scale image), while the chrominance is represented by the 2d vector field UV. After filtering, we obtain two scalar similarity maps that can be combined to a final similarity map representing edges in the color image.

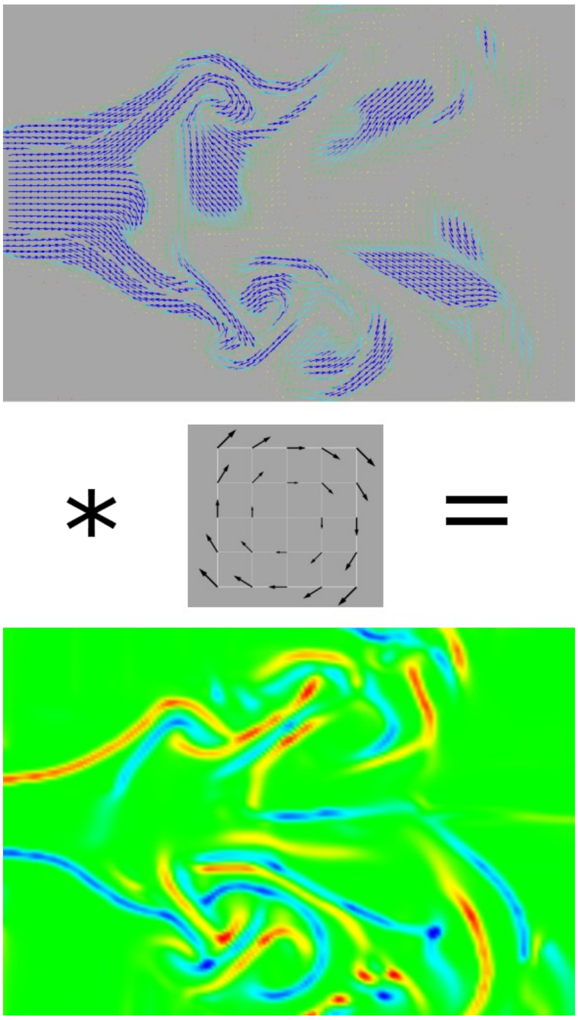

Figure 1: Pattern matching applied to vector fields: a fluid flow dataset (swirling jet entering fluid at rest) is undergoing a convolution operation with a rotation filter mask. The result is a scalar map of ”similarities”, showing regions of high vorticity.

The grey-scale part (Y) can be treated exactly as done in the common approaches, and the pattern matching applied to the color part (UV) adds additional information.

5.2 Data structure

The Clifford multi-vectors in the 2d case are suitable for this approach. We can assign the values as follows:

name	grade	dimension	values
scalar	0	1	Y
vector	1	2	U, V
bivector	2	1	0

The Y component becomes the scalar part of our multi-vector at each grid position. The vector component is represented by UV, using U as real part with basis e_1 and V as imaginary part with basis e_2. The imaginary scalar part with basis $e_1 \wedge e_2$ is set to zero, as there is no imaginary component for the scalar part.

5.3 Choice of patterns

For the scalar part the choice of filters is simple, as it reduces to grey-scale edge detection. For the color component UV, the main question for using vector pattern matching is what patterns to search for. Since the topologically interesting features like rotation, divergence, and convergence are related to the derivation operator, these patterns are a reasonable choice. We used four different vector patterns for filtering the UV part. They are presented in Figure 2.

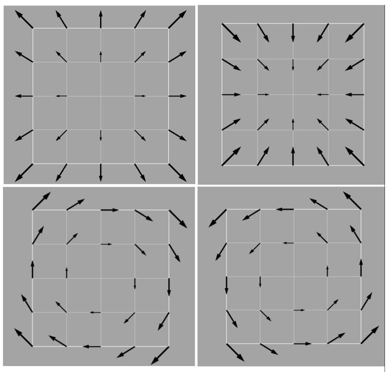

Figure 2: Four different vector pattern mask used as filters for the UV part: divergence (upper left), convergence (upper right), clockwise rotation (lower left) and counter-clockwise rotation (lower right).

5.4 Summary

We did not mentioned, that a blur filter could be applied prior to edge detection, as done in Marr-Hildreth and Canny edge detection schemes. It could be applied separately to all color channels, before converting an image to YUV, as well as to the YUV representation, since the conversion is a linear operation. As mentioned in section 2.1.2, an appropriate choice for these filters could enhance the final result.

The complete process of the Clifford color edge detection process is illustrated in Figure 3. We obtain two resulting similarity values. One real similarity value for the scalar part and a complex one for the vector part. Computing the magnitude of the complex value, the resulting two real values can either be combined or seperately further processed according to Canny's algorithm.

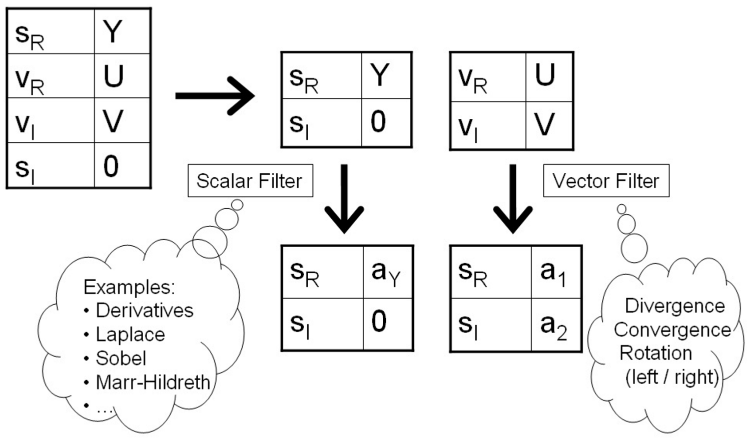

Figure 3: Illustration of Clifford color edge detection. Image given in YUV color space in a multi-vector structure, grey-scale edge detection performed as usual, UV part filtered with vector pattern matching. Result is a multi-vector of similarities.

6 Results

Given a color image, our algorithm computes two sets of similarity values: a set of real values and a set of complex values. The real set describes a fuzzy representation of the edges in the grey-scale image, while the magnitude of the complex value indicates edges in the color part of the image. The exact structure of the complex value depends on the used filter, but the unsigned magnitude of the complex similarity values turned out to be equal for all four filters that we applied (two rotation pattern, a convergence, and a divergence pattern, see Figure 2).

A user can adjust the binary threshold and the ratio of grey-scale and color contribution. It is possible to configure the identified edges as desired. We have chosen manual adjustment, as the automatic thresholding problem known from Canny's algorithm still applies to our case. Using the YUV model, we allow a user to control luminance and chrominance separately. This representation is more intuitive than a representation in RGB space, as the human sensory system processes luminance seperately from chrominance.

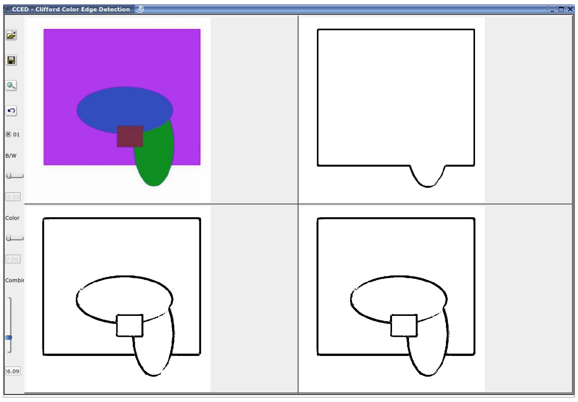

Figure 4: Example where grey-scale edge detection fails, while color edge detection succeeds. Upper left: (contrast enhanced) original image, upper right: result of a grey-scale edge detection, lower left: the color edge detection, lower right: the combined edge detection, showing all edges in the original image.

We have processed various images using this method. It generally performed better than simple grey-scale edge detection. Figure 4 shows an example where a grey-scale recognition fails. Examples for real-world image data are shown in Figures 5 and 6. We illustrate the enhancement using our color approach by comparing with the common grey-scale algorithm. Our algorithm turned out to be equal in performance to an optimally config-

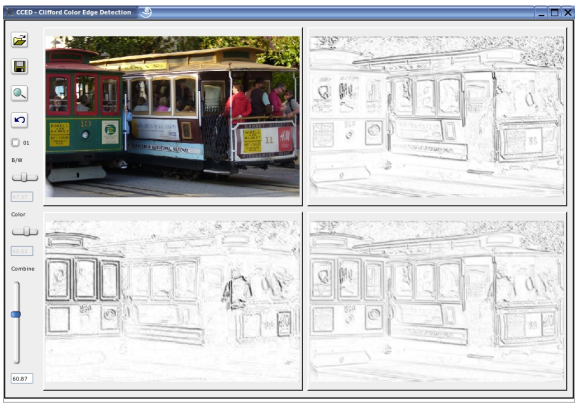

Figure 5: C^2ED applied to example image, resulting in a fuzzy representation of the similarity values. Upper left: original, upper right: Y filtered, lower left: UV filtered, lower right: weighted combination of Y and UV parts

ured color edge detection method using the RGB color model. The transformation from RGB into YUV space does not change the result when all component results are weighted equally. The vector-based approach in Clifford algebra for handling the UV part is an operation applied to complex-valued scalars. Those can again be rewritten component-wise, splitting them into a real and an imaginary part. Rewriting the vector filter pattern (see Figure 2) in components yields the commonly used filters for edge detection for scalar images in both axis directions. The four different types of filters only differ in their algebraic signs for the real and imaginary values. Since in the final step of our algorithm the magnitude of the complex result is computed, the search pattern can be either one of the given ones to obtain the same result.

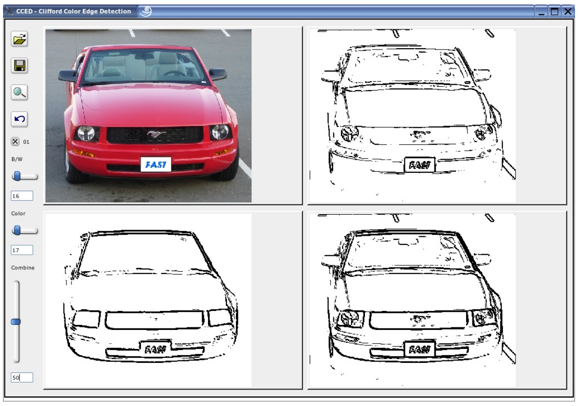

Figure 6: For this image of a car (upper left) thresholding was performed, resulting in a binary edge representation (upper right: Y, lower left: UV, lower right: combined). Results can be further improved using Canny’s method.

7 Conclusions

We have presented a new method for color image edge detection for color images using Clifford algebra. We handle grey-scale data seperate from the color part of the image. The luminance part is handled using common methods, while the color part is filtered with a vector-valued filter. Those two approaches fit perfectly in the data structure of Clifford algebra’s multi-vector setting. Our results have shown that this approach outperformes the grey-scale edge detection in most cases, since additional information is gained through the processing of the color part. However, it turned out to be equal in performance when compared to other color edge detection methods working component-wise on RGB images. For non-automatic detection, we implemented a framework that offers the possibility to adjust thresholds manually. For manual adjustment the YUV color model is more intuitive. Our algorithm can also be combined with the optimal Canny edge detector.

A remaining challenge is the threshold problem. There is still a need for an automatic method to determine a ”good” threshold.

8 Acknowledgements

We would like to thank the members of the Visualization and Computer Graphics Research Groups at the Institute for Data Analysis and Visualization (IDAV) at the University of California, Davis, and at the University of Kaiserslautern, Germany as well as the members of our International Research Training Group (DFG IRTG 1131) for fruitful discussions, especially Christoph Garth, Younis Hijazi, and Oliver Rübel. We thank Wolfgang Kollmann, UC Davis, for providing the fluid flow data, and Manuel Heringer and Florian Morr for programming support for its visualization. The IRTG is supported by the German Research Foundation (DFG). This work was also supported in part by the National Science Foundation under contract ACI 9624034 (CAREER Award) and a large Information Technology Research (ITR) grant. We further gratefully acknowledge the support of the W.M. Keck Foundation provided to the UC Davis Center for Active Visualization in the Earth Sciences (CAVES).

References

[Can86] J. Canny. A Computational Approach to Edge Detection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 8, pages 184–203, 1986.

[ES03] J. Ebling and G. Scheuermann. Clifford Convolution And Pattern Matching On Vector Fields. In *Proceedings of IEEE Visualization 2003*, pages 193–200, 2003.

[ES05] J. Ebling and G. Scheuermann. Clifford Fourier Transform on Vector Fields. *IEEE Trans. Vis. Comput. Graph.*, 11(4):469–479, 2005.

[HEWK03] E. Heiberg, T. Ebbers, L. Wigström, and M. Karlsson. Three-Dimensional Flow Characterization Using Vector Pattern Matching. *IEEE Trans. Vis. Comput. Graph.*, 9(3):313–319, 2003.

[HS99] D. Hestenes and G. Sobczyk. *Clifford Algebra to Geometric Calculus, A Unified Language for Mathematics and Physics*. Fundamental Theories of Physics. Kluwer Academic Publishers, Dordrecht, Boston, London, 1999.

[Jäh95] B. Jähne. *Digital image processing*. Springer-Verlag, Berlin, third edition, 1995.

[Jai89] A.K. Jain. *Fundamentals of digital image processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.

[KA05] A. Koschan and M. Abidi. Detection and Classification of Edges in Color Images. *IEEE Signal Processing*, 22(1):64–73, January 2005.

[Kos95] A. Koschan. A comparative study on color edge detection, 1995.

[MH80] D. Marr and E. Hildreth. Theory of Edge Detection. *Proceedings of the Royal Society of London*, B207:187–217, 1980.

[MP83] R. Machuca and K. Phillips. Applications of Vector Fields to Image Processing. *IEEE Trans. Pattern Anal. Machine Intell.*, 5(3):316–329, May 1983.

[Nev77] R. Nevatia. A Color Edge Detector and Its Use in Scene Segmentation. *IEEE Trans. Syst. Man Cybern.*, 7(11):820–826, November 1977.

[Rob77] G. Robinson. Color Edge Detection. *Optical Engineering*, 16(5):479–484, September 1977.

[Sch04] M. Schlemmer. Fourier Transformation and Filter Design for Clifford Convolution. Master's thesis, 2004.

[TH97] H. Tao and T.S. Huang. Color Image Edge Detection Using Cluster Analysis. In *IEEE International Conference On Image Processing*, pages 834–836, 1997.

[WM95] A.R. Weeks and H.R. Myler. Edge Detection of Color Images Using the HSL Color Space. 2424:291–301, 1995.