

# Biomedical Image Analysis: Rapid prototyping with Mathematica

***Citation for published version (APA):***

Haar Romenij, ter, B. M., & Almsick, van, M. A. (2004). Biomedical Image Analysis: Rapid prototyping with Mathematica. In *Wolfram Technology Conference 2004* (pp. 14-21).

***Document status and date:***

Published: 01/01/2004

***Document Version:***

Accepted manuscript including changes made at the peer-review stage

***Please check the document version of this publication:***

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

***General rights***

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

[www.tue.nl/taverne](http://www.tue.nl/taverne)

***Take down policy***

If you believe that this document breaches copyright please contact us at:

[openaccess@tue.nl](mailto:openaccess@tue.nl)

providing details and we will investigate your claim.

# Biomedical Image Analysis: Education & Applications with *Mathematica*

*B.M. ter Haar Romeny, M.A. van Almsick*

*Eindhoven University of Technology  
Eindhoven, the Netherlands  
Department of Biomedical Engineering  
Biomedical Image Analysis Group*

*B.M.terHaarRomeny@tue.nl    M.A.v.Almsick@tue.nl*

## **Abstract**

We report on 2.5 years of successfully using *Mathematica* as a high-level rapid prototyping tool in education and research of biomedical image analysis.

## **Introduction**

Digital acquisition techniques have caused an explosion in the production of medical images, especially with the advent of multi-slice CT and volume MRI. One third of the financial investments in a modern hospital's equipment are dedicated to imaging. Emerging screening programs add to this flood of data. The capabilities of many recent computer-aided diagnosis (CAD) programs are compelling, and have recently lead to many new CAD companies [1]. This calls for many new algorithms for image analysis and dedicated scientists for the job.

Image analysis software libraries abound [2], but unfortunately are often limited in functionality, are too specific, or need a rather dedicated environment and have a long learning curve. Today's computer vision algorithms are based on solid mathematics, requiring a highly versatile, high level mathematical prototyping environment. We have chosen *Mathematica* (*Mathematica*) by Wolfram Research [3], and describe the successful results of the first 2.5 years of its use in the training of biomedical engineers in image analysis.

## Software Requirements

In 2001 the BME department established a new MSc track on 'Medical Imaging', including our group on Biomedical Image Analysis. As we started from scratch, no software infrastructure was present and we made a choice for a new rapid prototyping environment based on the following requirements:

- A steep learning curve
- Functional programming
- Pattern matching
- Typically short code
- Integration of code and text
- Symbolic functionality
- Interpreter (interactive debugging)
- Fast numerical functionality
- Complete (w.r.t. functionality)
- Professional text, graphics, animations;
- Mathematical symbolic notation;
- Platform independent
- Easy extensions of operations to N dimensions

Typically the design is in two stages: after the development of the (mathematical) routine into a working prototype, a more clinical or industrial version is / can be generated in C++, Java, etc. We use the Visual ToolKit (VTK) and OpenGL for the interactive 3D volume visualizations, as this is not contained in Mathematica.

## Implementation in the department

We decided to promote the use of Mathematica for every student project initiated, to establish rapidly a critical mass of people with expertise. The second author, authorized Mathematica trainer with over 15 years of experience regularly conducts interactive training sessions for new batches of students. All student reports are generated as Mathematica 'notebooks', effectively combining a professionally looking text document with the annotated code. The University has one of the largest Mathematica licenses in the Netherlands, so the program could be installed on all systems, including student and staff laptops for home use. Installation of all campus software is eased through the public folders in MS Outlook. Special Mathematica pages on the group's website [4] summarize all instructions, tutorials, software packages and latest news for staff and students.

For all second-year students (10 groups of each 8 students) a so-called 'casus' was developed: "Image analysis for pathology". Duration 6 weeks, 50%. Students are challenged to find clever algorithms for the analysis of shape, number, texture and size of histological cell images. The many 'inventions' possible, and the ease of programming their ideas in Mathematica was very rewarding for most of the students. It turned out that especially the adoption of a

functional programming style took longer, as this is a new and unconventional way of programming for 2nd year students (e.g. no more loops, use of pattern matching).

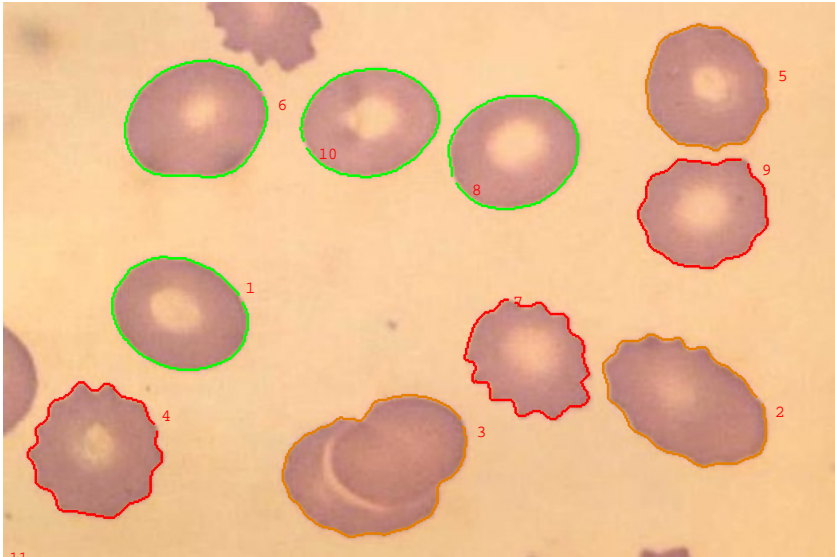


Figure 1: Example of output of a cell classification project by 2<sup>nd</sup> year BME students: red outline for acanthocyte cells, yellow for unknown or overlapping, green for round normal blood cells

'Design oriented projects' (DOP's) are 3rd and 4th year's student projects where they solve a practical problem in small groups. Duration: 6 resp. 12 weeks, 50%. Students successfully worked on the following topics, all implemented and documented in Mathematica notebooks (which are all available at [4]):

- Finding the hardly visible Adamkiewicz vessel in 3D MRA datasets;
- Image matching by mutual information maximization;
- Edge preserving smoothing for 2D and 3D data by PDE evolution (fig. 1.);
- Trabecular bone morphology statistics;
- Volume visualization, perspective rendering;
- Volume visualization of color microscopy 3D images;
- Maximum Intensity Projection (MIP) visualization for 3D color microscopy images;
- Deblurring CT Gaussian slice thickness blur for enhanced multi-planar reformatting;
- Face recognition by Eigenfaces;
- Active shape models for 3D shape variability of the normal and infarcted mouse heart;
- Histogram equalization;
- MRI background equalization by entropy minimization (fig. 3.);
- The Visible Mouse: segmentation and visualization of high-resolution 3D 6.3 Tesla MRI data (brain, heart) of the mouse for molecular imaging;
- Segmenting the intervertebral space in high resolution 3D CT data for individualized design of vertebral disk implants;
- Thickness color map on surface renderings
- 3D phantoms and realistic test images;
- Computer-Aided Diagnosis (CAD) of lung pathology by multi-scale texture analysis;



Figure 2. Perspective 3D volume rendering of the 'lobster' dataset.

In the 4th year a facultative course [6] is given by the first author: "Front-end vision and multi-scale image analysis". This course takes a multi-scale differential geometry approach to computer vision, based on bio-mimicking mathematical models of the first physiological stages of human visual perception. This intensive 2-week course is also a national PhD course. The textbook of this course is completely written in Mathematica [3], and has an included CD-ROM with the full code / text. All topics can be interactively experimented with. During the interactive computer laboratories after the lectures students individually or in pairs study the tasks in the chapters of the book by Mathematica implementation. From the review forms collected it was clear that this lab was highly appreciated. The learning curve was steep.

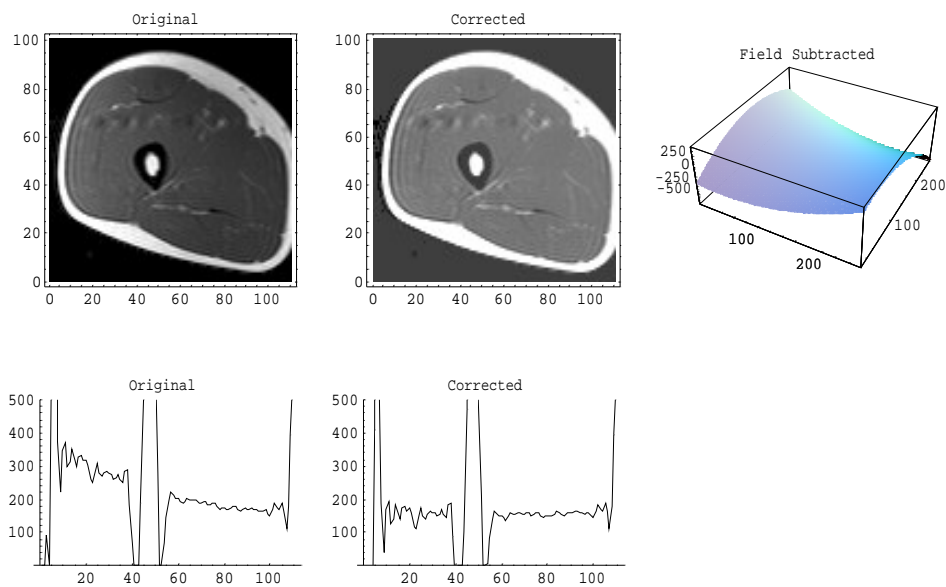


Figure 3. 2D MRI background inhomogeneity removal by finding a second order polynomial subtraction surface by entropy minimization and gradient descent.

In the 5th year Master's projects of 9 months duration are carried out, most of them oriented towards algorithm development:

- Catheter and guidewire detection in low-dose fluoroscopy images;
- Multi-scale optic flow analysis of the moving heart wall from MRI tagging, using Lie

derivatives;

- Multi-scale level sets and watershed segmentation for 3D datasets of the Visible Mouse;
- Image retrieval by multi-scale toppoints;
- Motion analysis by multi-scale toppoints;
- Perceptual grouping of edge contours for generic segmentation of biomedical data by multi-scale, biologically inspired oriented filterbank analysis.

## MathVisionTools

We are developing a rapid prototyping AddOn in Mathematica for biomedical image analysis applications: MathVisionTools. The output of the student's projects is moderated by the authors of this paper and added to this collection of routines. Students always are guaranteed to use the latest release by reading the package code from the internet with Mathematica's GetURL command. We consider to be on-line as the default situation while carrying out study or research. This option also turned out to be a convenient option by sharing images or code in a collaborative project.

As image analysis often involves geometric reasoning, the use of differential geometric methods is abundant. One of the key elements of the package is a multi-scale, N-dimensional, speed optimized Gaussian differential operator convolution function, to take multi-scale derivatives of discrete data to any order. We also added the functionality to read DICOM files (the international / industrial image standard format for medical images), including older .ani and .spi dialects, and returning all descriptive groups.

A WebMathematica [9] server is installed, enabling the design of a user-interface in a web browser, communicating the Mathematica commanda to a dedicated server. Mathematica is an interpreter, so code cannot be compiled and given away e.g. to doctors, but code can be 'published' on the web. This could be an effective mechanism to offer remote image analysis services, such as web-mediated computer-aided diagnosis for hospitals.

## Additional Infrastructure

Image processing and computer vision algorithms typically require fast processors and large amounts of memory. At the TU/e all students are equipped with a partly sponsored laptop when they enter the University (currently 6800 students). Students which acquired their laptop some years ago are faced with poor specifications for today's jobs. Mathematica consists of two programs, a front-end and a kernel. The kernel can run on a remote computer. We implemented several rack mounted server PC's with dual processor 2.8 GHz processors, running Linux, for remote Mathematica kernels. This solved the capacity problem with a minimal investment and great student's satisfaction.

## Discussion

The implementation of a high level prototyping programming environment has been an important step to facilitate especially the design phase of image analysis routines, both for education and research. Mathematica has turned out to be an excellent choice, being a coherent program with both powerful symbolic and numerical capabilities. The symbolics enables the student to 'play with math' again, and the numerics now outperform many competitive matrix oriented programs.

The code written in our department will be freely available for those labs that join us (on an exchange basis) in creating a high-level MathVisionTools box.

## Acknowledgement

We thank the BMIA students for their enthusiasm in pioneering and useful feedback. This paper has in modified form been presented at the X<sup>th</sup> Mediterranean Conf. on Biomedical Engineering, Medicon '04, August 2004, Napoli.

## References

- [1] CAD companies: [www.r2tech.com](http://www.r2tech.com), [www.deustech.com](http://www.deustech.com), [www.cadxmed.com](http://www.cadxmed.com), etc.
- [2] Computer vision software list:  
[www-2.cs.cmu.edu/afs/cs/project/cil/ftp/html/v-source.html](http://www-2.cs.cmu.edu/afs/cs/project/cil/ftp/html/v-source.html).
- [3] Mathematica: [www.wolfram.com](http://www.wolfram.com).
- [4] TU/e Biomedical Image Analysis Group:  
[www.bmi2.bmt.tue.nl/image-analysis/](http://www.bmi2.bmt.tue.nl/image-analysis/)
- [5] B. M. ter Haar Romeny, "Front-End Vision and Multi-Scale Image Analysis. Multi-Scale Computer Vision Theory and Applications, written in *Mathematica*". Dordrecht: Kluwer Academic Publishers, 2003.
- [6] TU/e Front-End Vision course: [www.bmi2.bmt.tue.nl/image-analysis/education/courses/FEV/course/index.html](http://www.bmi2.bmt.tue.nl/image-analysis/education/courses/FEV/course/index.html).
- [7] ter Haar Romeny, B.M. (2002): 'Computer Vision & Mathematica 4', *Computing and Visualization in Science*, 5, 1, pp. 53-65.
- [8] MathVisionTools: [www.mathvisiontools.com](http://www.mathvisiontools.com).

[9] [www.wolfram.com/products/webmathematica](http://www.wolfram.com/products/webmathematica)

[10] Suinesiaputra, A. , Florack, L.M.J., Westenberg, J.J.M., ter Haar Romeny, B.M., Reiber, J.H.C., Lelieveldt, B.P.F. (2003): ‘Optic Flow Computation from Cardiac MR Tagging Using a Multiscale Differential Method: A Comparative Study with Velocity-Encoded MRI’. Lecture Notes in Computer Science, vol. 2878, Randy E. Ellis and Terry M. Peters (Eds.), pp 483-490.