OPEN SOURCE TOOLS AND PROJECT-BASED TEACHING AS ENABLERS OF RESEARCH EXPERIENCE IN COMPUTER VISION STUDENTS

Hemerson Pistori – pistori@ucdb.br
Universidade Católica Dom Bosco (UCDB)
Av. Tamandaré, 6000
79117-900 Campo Grande - MS
Mauro Conti Pereira – maurocp@gmail.com
Universidade Católica Dom Bosco (UCDB)
Av. Tamandaré, 6000
79117-900 Campo Grande – MS
IFMS – Instituto Federal de Educação, Ciência e Tecnologia de MS
Av. Julio de Castilho, 4560
Campo Grande - MS
Marco A. Alvarez – malvarez@udel.edu
University of Delaware
19713 – Newark – DE - USA
Xiaojun Qi – Xiaojun.Qi@usu.edu
Utah State University,
84322 - Logan – UT – USA

Abstract*: We present two project-based experiences in teaching computer vision courses to senior undergraduate students and low-level graduate students at two universities. Both use a variety of open source tools to teach the abstract computer vision concepts and algorithms in an intuitive manner using a laboratory equipped with Internet access and appropriate software and hardware. In this way, students can experiment with major image processing and computer vision methods during the lecture period and may ask some insightful questions for deeper understanding. Furthermore, we help students to choose an appropriate semester-long project to work on either independently or in a group of 2 or 3 students. We also actively guide students in surveying literatures, writing research papers, and overcoming any difficulties. The successful project outcomes and research ability oriented grading policies expose students to the intellectual excitement involved in research activities and encourage students to pursue academic jobs or Master/Ph.D. degrees in computer vision and related areas.*

Key-words: *computer vision, open-source tools, project-based methodology, image processing*

1. INTRODUCTION

Computer vision is an important and growing research area which provides several advantages in productivity as well as improving living standards (BEBIS et al., 2003). In fact, vision technology is widely applied ranging from high-level web applications to basic algorithms for leveraging other disciplines such as biology, physics, robotics, and geo-processing. In computer science and related engineering undergraduate programs, image processing and computer vision courses are usually taught at senior years. Likewise, they are in general offered in MSc or PhD programs.

We have analyzed two project-based experiences for teaching computer vision concepts to seniors under two different scenarios. One is the teaching experience at the Computer Engineering (CE) program (http://www.bducdb.ucdb.br/index.php3?curso=130) at Dom Bosco Catholic University (UCDB), Brazil. The other is the teaching experience at the Computer Science (CS) Department (http://www.cs.usu.edu) at Utah State University (USU). Both universities are placed into different cultural and academic environments as well as their underlying infrastructure and available resources differ in quantity and quality. To date, we have extensively exchanged our ideas on improving teaching methodologies and estimating the appropriateness of exposing students to local real-world research projects under such diverse circumstances.

We claim that project-based teaching is a good venue for seniors to learn cutting-edge technologies and make a smooth transition from school to industry or conversely follow an academic path. Moreover, in our case we present evidence that supports our claims under two different scenarios. At CS/USU professors are fully involved with research work. The department offers MSc and PhD programs in Computer Science, having a large number of graduate students enrolled in classes. In this context, previous and on-going funding at CS/USU have contributed to establish computer vision labs fully equipped with hardware and appropriate licensed software.

On the other hand, the CE/UCDB program is offered with no attached graduate programs. In this context, funding and infrastructure is scarce for computer vision projects. We believe that introducing open-source assets as pedagogical tools and encouraging the students to work on real-world problems can contribute to bridge the gap and yield successful and similar outcomes even under such different contexts.

By engaging computer vision students in representative projects we expect to:

- Encourage students to think creatively and independently;
- Encourage students to consider graduate studies;
- Help students to develop general research skills;
- Help students to gain hands-on experience in real world applications by applying computer vision-based technologies to real-world problems;
- Teach students to assimilate latest research, assess their own research, present experimental results, and prepare publications;
- Help students to establish a sense of self-assurance and team-spirit.

Additionally, by introducing open source tools in teaching activities, students face a smooth learning curve which quickly involves them with implementation details about the abstract concepts being taught. On the other hand, by encouraging students to use open source tools during project development, they have the opportunity to participate in open source communities and therefore share the software derived from their work. Pieces of the software can also be used as effective teaching tools since they are linked to real projects developed by students and are excellent examples to illustrate accomplishments achieved by close collaboration. Two sample projects developed in this manner are a graphical visualizer for particle filters and an ImageJ plugin for Hough Transforms experimentation. The former includes examples and instructions to use the mice tracking system in open-field experiments (GONÇALVES et al., 2007). The latter is freely available at http://rsb.info.nih.gov/ij/plugins/hough-circles.html.

In both experiences, we focus on course projects applied for solving real-world problems. In this way, they can be easily recognized and appreciated by students. Two sample course projects are automatic detection of bovine leather defects using computer vision (VIANA et al., 2007; PARAGUASSU et al., 2010), and multiple mice tracking using particle filters (PISTORI et al., 2010).

The organization of this paper is as follows: Section 2 briefly presents the major topics covered by both courses. Section 3 describes the main open source tools used for pedagogical support. Section 4 explains the project-based methodology and presents various projects generated from both courses. Section 5 reviews the grading policies. Finally, we conclude the paper with the highlights of our experiences.

2. COURSE ORGANIZATION

The computer vision course offered at CE/UCDB is a 4-credit course distributed in 20 weeks while the computer vision course at CS/USU is a 3-credit course offered in 16 weeks. Both courses provide an introduction to the fundamental concepts, theories, and algorithms of computer vision. Additionally, both courses emphasize the importance of mathematical concepts learned at early years in their respective programs (i.e., linear algebra, calculus, probability, statistics, signal and systems and discrete mathematics). In this way, students are prepared to understand a variety of complex computer vision algorithms.

In order to keep pace with changing technologies, we incorporate practical components into lectures. This integration helps students gain an intuitive comprehension of fundamental computer vision concepts. We also explain specific topics related to current computer vision research projects conducted by both research groups without compromising course objectives. Thus, we achieve seamless integration between teaching and research and expose students to hands-on experiences at early years in their academic life. Here, we present a summary of the covered topics:

(1) Introduction to Computer Vision: history, relationship with other fields, tools, sub-areas and applications;
(2) Image formation and representation models: human vision and image capture devices, illumination, color, texture, shape, image representation, and noise models;
(3) Filters: convolution, spatial and frequency domains, image enhancement, and edge detection;
(4) Morphological operations: definitions, dilation and erosion, opening and closing, and examples;
(5) Feature extraction and selection: statistical moments, filter banks, co-occurrence matrices, principal component analysis, and discriminant analysis;
(6) Segmentation: seeded region growing, thresholding based, clustering based, model fitting based, texture based, and watershed;

(7) Tracking: concepts, optical flow, and predictive filtering; and
(8) Supplemental topics: stereoscopic vision, augmented reality, automatic learning and pattern recognition, statistical methods for segmentation and object recognition.

For each topic, we revisit basic mathematics, signal processing, computer science, and operations research concepts when needed. For example, vector spaces and linear transformations are explored in conjunction with color spaces and feature vectors. Deeper explanation of eigenvectors and eigenvalues is used to make the essence of several feature selection algorithms clear to students. Differential calculus, specially the concept of gradient, is covered intuitively when edge detection and optical flow algorithms are presented. Signal processing is revised during the study of filters at the frequency domain. Statistics are revisited for introducing segmentation and tracking methods and analyzing the performance of different algorithms. Some concepts in automata theory, formal languages, and graph theory are also revisited and extended in syntactic pattern recognition algorithms. Operations research, graph theory, and linear algebra are presented along optimization problems in computer vision.

3. OPEN SOURCE TOOLS

Open source tools and libraries are excellent resources to support teaching methods and student activities. In our project-based experiences, we rely on such packages. The main reasons to choose open source tools are:

- They are affordable for low-funding schools and are available in a professional quality;
- They provide students with an opportunity to inspect the source code and modify it if necessary;
- Their libraries often facilitate the development, testing, and maintenance phases;
- They help students to understand complex concepts through the analysis of their source code; and
- They motivate students to participate and collaborate with software development communities.

In particular, we use laboratories equipped with Internet access and appropriate software for our lectures. Internet access plays a significant role to explore abstract concepts in practice. To this end, students are encouraged to independently search the web for useful information on topics covered in class. In this way, they can explore different perspectives when learning new concepts given in lectures and ask more insightful questions to gain deeper understanding. Useful links and resources found by students are shared through the course website. Regarding the installed software, basically we rely on two types of software: open source packages and java applets. Packages are useful for teaching because they provide working algorithms for computer vision concepts. Moreover, their implementations are in general easy to extend. ImageJ (ABRAMOFF et al., 2004; RUSBAND, 2007), and Scilab (PIRES & ROGERS, 2002; GOMEZ, 2007) are good examples of packages used in our teaching activities. Additionally, Java applets are exploited in lectures in order to facilitate the understanding of specific concepts. The links for sample packages and applets are also available at the CE/UCDB course website (http://www.gpec.ucdb.br/pistori/disciplinas/visao).

The ImageJ package is a public domain multi-platform version of the NIH Image Software. We use ImageJ to learn several abstract concepts (e.g. histogram equalization, edge detection, Fourier and Hough transforms, convolutions, morphological operations, and so on) by experimenting with different parameter tuning options. Figure 1 shows a screenshot of running the Color Inspector 3D ImageJ plugin (BARTHEL, 2007) on a sample color image to visualize its 3D histograms in different color spaces.

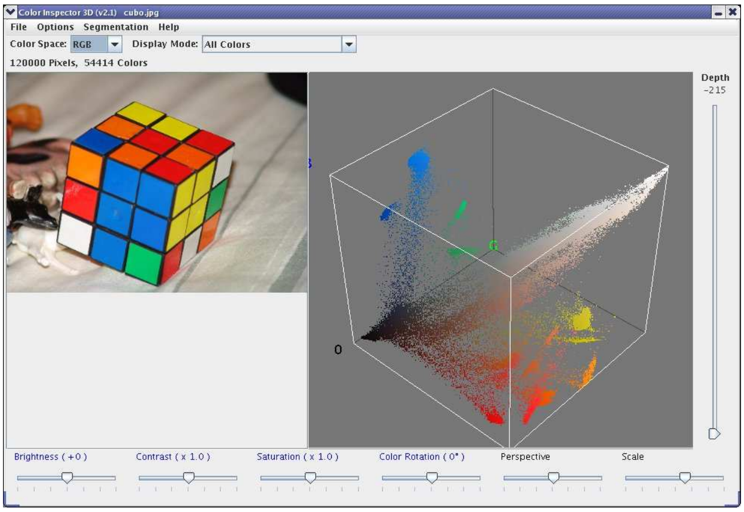

Figure 1. 3D Histogram, in the RGB color space, produced by the *Color Inspector 3D plugin*

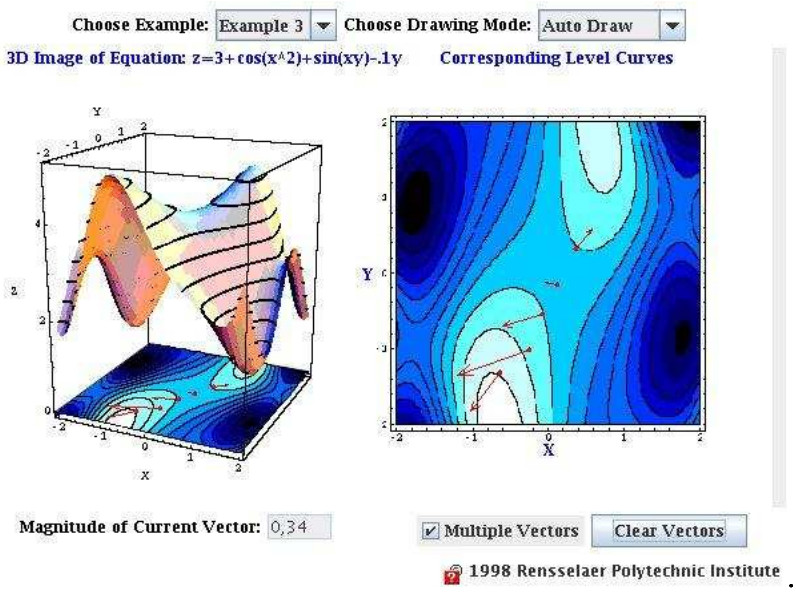

Figure 2. Gradient vectors, in red, generated by the *Visualizing the Gradient 2D applet*

Scilab is a scientific software package for numerical computation and provides a powerful tool for engineering and scientific applications. It includes hundreds of mathematical functions with the possibility of adding external contributions or libraries. It also has excellent documentation and provides technical support for users. An image processing toolbox for Scilab, the SIP (FABBRI 2003, 2007), has several available resources and is now in advanced development stage. We use Scilab primarily for illustrating algebra and statistics concepts.

Other open source packages include the project LITE (BRECHER, 2007) and the Snakes (ANDERSSON, 2007). The project LITE offers a comprehensive list of applets and educational materials for clarifying the nature of light, optics, color, and perception; and the Snakes allows the user to experiment with three types of noise (Gaussian, salt & pepper and speckle) for contour detection. Among other examples of applets are Watershed (STADELMANN, 2007), Converging Lens (KISELEV, 2007), Visualizing the Gradient (2D) (RPI 2007), Color Spectrum (VISHNEVSKY, 2007), Joy of Convolution (CRUTCHFIELD, 2007). Figure 2 demonstrates one example on gradient vectors using the Visualizing the Gradient (2D) Applet.

During lectures, we use Java applets and animations to visually explore different stages of algorithms and therefore easily motivate students to learn. In addition, most applets have options for parameter tuning and adding different types and levels of noise. Students can interactively examine the effect of different configurations on computer vision algorithms.

4. PROJECT-BASED METHODOLOGIES

Project-based computer vision courses at both universities require students to finish a project towards the end of the semester. Along these lines, instructors are committed to provide prompt feedback to submissions at different stages throughout the semester. Both courses encourage students to directly use libraries or ready-to-use implementations so as to focus on the main idea of the solution, its possible improvement, and the analysis of their results. Both courses also motivate students to adapt their course projects to solve real-world problems. This adaptation is carried out through close interaction with the Research Group in Engineering and Computing (GPEC) at UCDB and the Computer Vision and Image Processing (CVIP) laboratory at USU, respectively.

However, both courses adopted different methodologies in employing research components in teaching. Specifically, CE/UCDB requires a group of 2 or 3 students to choose a project satisfying mutual interests. The CS/USU requires each student to work independently on a project to his/her own interest. The CE/UCDB also requires each group to turn in a final paper, whereas the CS/USU does not have this requirement so students have more time in development and debugging. In the following, we compare these two methodologies and analyze the reasons for these differences.

4.1. Research Integration

Different approaches are taken at the two universities.

Project at UCDB

Three milestones are established at CE/UCDB for ensuring that each group works in the right pace:

Preliminary paper. Students form the group based on their interests during the 1st week. Each group starts reading several survey papers related to their chosen project in the second week. The instructor helps each group to clarify any doubts and explain some technical details if needed. Each group starts writing a preliminary paper, which contains an extensive literature review about the chosen project and an introduction to the motivation, goals, and methodologies to be employed in the project. Each group submits a digital copy of the paper (3 to 6 pages), following the LATEX format of the Brazilian Computer Society (http://www.sbc.org.br/template) by the end of the fourth week..

Intermediate paper. Each group consults the instructor about specific techniques proposed in the preliminary paper. Students are also advised in how to write a research paper. Students distribute their workload and responsibility and pursue the project. By the end of the 10th week, students turn in a digital copy of the detailed prospectus and the updated progress report.

Final paper. Each group continues working on their project under the guidance of the instructor. Some deviations from the proposed intermediate paper may be granted after thorough discussions with the instructor. By the end of the semester, students submit all the artifacts used in the project, including source code, additional resources, libraries, the user manual, and a final version of the paper.

Project at USU

Project activities at CS/USU take a different route. First, the instructor integrates different potential research topics into the individual assignments. In this way, students can quickly grasp the main idea of some components of the research project. Second, each student turns in a preliminary paper (1 to 2 pages) to explain the details of the project to be completed by the 10th week. To this end, each student needs to choose a high-quality research paper as a base to build his/her project. Each student is also expected to include complete discussions on data sets to test on, assumptions for solving the problem, details of the algorithms, any possible improvement over the original research paper, and special data structures to be used for a large-scale experiment, etc. Third, each student presents his/her work in class during the final week and receives comments from peers and the instructor. Fourth, some good students might be recruited by the instructor to continue improving the semester project and possibly integrate some components into the projects conducted at the CVIP lab at USU.

Comparing both approaches

The differences in research integration at both universities mainly come from different curriculum prerequisites. Senior students at CE/UCDB correspond to the fifth year of studies, instead of the fourth year for CS/USU senior students. At CE/UCDB, seniors are suggested to take the computer vision course and are required to finish a senior project before graduation. As a result, all the registered students should have already worked or started working on real research projects when taking the computer vision course. Some also worked or are working on computer vision related projects through the "Scientific Initiation" program (PIBIC) supported by the university. The extensive research experiences of UCDB seniors can help them to digest the main ideas of various technical papers and decide on a research path at the initial weeks of the course. At CS/USU, senior students and low-level graduate students are suggested to take the computer vision course. However, senior students are not required to do the senior project before graduation. As a result, registered students may not have any experience in working on a research project. This makes it difficult to let students start reading survey papers in the first couple of weeks.

The wide range of student backgrounds also makes it difficult to assign a group project to 2 or 3 students. However, software engineering classes at USU prepares seniors for the necessary skills for team work..

Another factor also contributes to the differences in research integration at two universities. At CE/UCDB, senior students are required to take two courses per semester allowing students to focus on conducting a research project. Thus, students have more time to work on a relatively large project, which typically requires a team of 2 or 3 students. The instructors at CE/UCDB also provide tutoring before choosing the topic for each group. However, at CS/USU, registered students normally take 3 or 4 courses per semester. Consequently, the instructor encourages students to work independently on small projects rather than larger assignments.

4.2. **Grading Policies**

Different grading policies are adopted in both courses. At CE/UCDB, participation in lectures and the course mailing list, involvement in extracurricular activities and ad-hoc algorithm implementations for research projects at the research group (GPEC) account for major percentages of the grading. This grading policy emphasizes students' ability to read scientific literature, write technical and scientific papers, implement computer vision related projects, conduct experiments, and analyze experimental results.

At USU, the instructor puts more emphasis on implementing computer vision based algorithms, presenting scientific results, and assessing algorithms. In this way, students can learn useful techniques for conducting scientific research and also making an easy transition from school to industry.

In spite of these differences, we agree that we are consistent in one regard. That is, the ability to complete a literature survey and conduct research is more important than memorizing formulas. This has been taken into account in both grading policies.

4.3. **Outcomes**

Despite the differences, both courses have achieved encouraging outcomes using the project-based teaching approach. To further illustrate the effectiveness of this approach, we list some sample projects and their resultant publications during the past four years.

- CE/UCDB: Feature extraction. Extract useful motion and shape features for studying mouse behavior in a confined 2D space and monitoring this behavior through a webcam. Students studied features such as velocity, acceleration, distance following, postures, etc. by using a segmentation method, physics and geometry formulas, and statistical moments. Several classification algorithms like backpropagation, decision trees, bayesian nets, and naive bayes were tested for identifying mouse behavior. Students successfully developed the effective features for studying mouse behavior using ImageJ and WEKA1 tools (SILVA et al., ????).
- CE/UCDB: Multiple mice tracking. Track multiple mice using the particle filters technique, where ParticleAnalyzer and MultiTracker plug-ins in ImageJ are used for implementation. A prototype system was successfully built and some papers were published (GONCALVES et al., 2007). Another group of students are now engaged

in the development of a system to track Aedes aegyptti larvae in order to automate bioessays for new larvicides (PISTORI et al., 2010).

- CE/UCDB: Eye tracking. Apply a series of low-level image processing techniques to estimate eye-gaze positions. Gaussian and Sobel filters are also employed to remove noise. The Hough circle transformation is used to detect the pupil. A prototype system was successfully built using ImageJ and a conference paper was published. (MORETTO et al. 2006). This initial work has evolved into a set of computer vision tools that aid in the implementation of equipments to help persons with special needs (PISTORI, 2007).
- CE/UCDB: Live and dead yeast counting. Automate the process of live and dead yeast counting, from microscopic colored images, in order to improve the quality of microbiological control in sugar and alcohol factories. This project is of great regional interest as Brazil is a large sugar cane producer. Some students are receiving grants from local industry interested in the results.
- CS/USU: Image Orientation Detection. Apply supervised self-organizing map to automatically detect and correct image orientations for intelligent image processing. A prototype system was successfully built using and a conference paper was published (DATAR & QI 2006).
- CS/USU: Content-Based Image Retrieval. Apply the relevance feedback technique together with the long-term learning mechanism to learn the user's query concept and retrieve the images of the user's interests. Seven students built different learning mechanisms using Matlab on the COREL image database. Seven prototype systems were successfully built and seven papers were published (WACHT et al. 2006); (SHKURKO & QI 2007); (ROYAL et al. 2007); (LINENTHAL & QI 2008); (BARRETT et al. 2009); (FECHSER et al. 2010); (GILBERT et al. 2010).
- CS/USU: Digital Watermarking. Create a digital watermarking system to be robust against common image processing or geometric attacks. One student applied the improved Harris-corner detector to locate robust interest points in an image for developing a geometric attack resilient watermarking system. Another student applied the quantized index modulation technique in the wavelet domain together with a human visual system adaptivity to build a system against common image processing attacks. Two conference papers were published (WEIN-HEIMER et al. 2006); (Mckinnon & QI 2006). Built upon prior work of the CVIP lab at USU, two other conference papers were published (FIRST & QI 2007); (QI et al. 2008).
- CS/USU: Fire detection. Apply a computer vision-based approach for automatically detecting the presence of fire in stable video sequences. A cumulative time derivative matrix was used to detect areas with a high frequency luminance flicker. The fire color of each frame culminating in a cumulative color matrix was analyzed by a new color model which considered the saturation of the red components. The same cumulative time derivative matrix was further used to analyze the temporal variation of fire intensity and the spatial color variation of fire for detecting fires. One journal paper was published (QI & EBERT 2009).

Dr. Qi at USU has successfully integrated interesting research components into teaching and coauthored with undergraduate students on thirteen publications. Six students attended graduate school to further their education. Two of her graduate students are working in a company using computer vision-based techniques. At UCDB, eleven students are currently enrolled in or have already graduated in computer vision related departments. Before the implementation of the proposed methodology we had no students applying for graduate school in computer vision. Also at UCDB, some students have started new companies to focus on solving real-world problems using computer vision techniques. All these successful stories prove that our innovation in combining project-based research into teaching achieves the expected outcomes.

5. FINAL REMARKS

This paper presented details about two different experiences in teaching computer vision concepts to senior or graduate students. In both cases, we use open source tools during lectures to enhance teaching activities so students can understand abstract concepts and algorithms in an intuitive manner, at the same time they develop a sense of accomplishment and collaboration by producing new tools and solutions. We also rely on a project-based methodology, where we actively guide students in surveying literature, encourage students to write research papers, help students to overcome any difficulties encountered during the development of the project, and use a research-focused grading policy to expose students to the intellectual excitement involved in research activities.

Despite the difference in infrastructure and resources in both universities, we successfully have achieved comparable outcomes. In both cases, students were able to publish their work and most of them followed an academic career or have been recruited by the high-tech computer vision industry. We point out that open source tools together with project-based teaching effectively contributed to engage students in research activities. Furthermore, projects targeting local real-world problems create interesting didactical opportunities to link arid abstract mathematical concepts to concrete and meaningful situations for the students. We also believe that lectures must be given inside a laboratory. Students can therefore experiment with major image processing and computer vision methods during the lecture period and may ask some insightful questions for deeper understanding.

The teaching strategies presented in this paper slightly differ in course contents, grading policies, and project development. In the same way, different universities may adapt our research-based methodologies with open source tools in their computer vision course or other senior courses.

Acknowledgements

Dr. Qi at USU was supported by a National Science Foundation grant "REU Site Program in Computer Vision and Image Processing" (IIS0453552; 2/1/2005-1/31/2008) and is currently supported by the NSF grant "REU Site Program in Computer Vision and Multi-Agent Systems" (IIS-0850825; 05/15/2009-04/30/2012) to enroll undergraduate students to participate in computer vision and image processing related projects.

6. REFERENCES

ABRAMOFF, M.; AES, P.M.; RAM, S. Image Processing with ImageJ. Biophotonics International, 11(7), 36–42, 2004.

ANDERSSON, M. **Snakes Applet.** http://www.s2.chalmers.se/research/image/Java/NewApplets/ACM/ Accessed em: 25 Nov 2007.

BARRETT, S.; CHANG, R.; QI, X. A Fuzzy Combined Learning Approach to Content-Based Image Retrieval. In IEEE International Conference on Multimedia & Expo (pp. 838–841), 2009.

BARTHEL, K.U. **Color Inspector 3D**. http://rsb.info.nih.gov/ij/plugins/color-inspector.html , Accessed em: 11 Nov 2007..

BEBIS, G.; EGBERT, D.; SHAH, M. Review of computer vision education. IEEE Transactions on Education, 46(1), 2–21, 2003.

BRECHER, K. **Project LITE**. http://lite.bu.edu/ Accessed em: 2 Nov 2007.

CRUTCHFIELD, S. The Joy of Convolution Applet. http://www.jhu.edu/~signals/convolve/. . Datar, M., & Qi, X. (2006). Automatic Image Orientation Detection Using the Supervised Self-Organizing Map. In IASTED International Conference on Signal and Image Processing, Honolulu, US, 2007.

FABBRI, R. Desenvolvimento de um Pacote de Processamento de Imagens para o Scilab. Projeto de Conclusão de Curso. SCC/ICMC da Universidade de São Paulo, 2003 Dissertação (Mestrado).

FABBRI, R. **SIP ToolBox**. http://siptoolbox.sourceforge.net/ Accessed in: 22 Nov 2007.

FECHSER, S.; CHANG, R.; QI, X.. Inter-query Semantic Learning Approach to Image Retrieval. In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'07) (pp. 1246–1249), 2010.

FIRST, E.; QI, X.. A Composite Approach for Blind Grayscale Logo Watermarking. In IEEE International Conference on Image Processing (ICIP'07) (pp. 265–268), 2007.

GILBERT, A.; CHANG, R.; QI, X. A Retrieval Pattern-Based Inter-Query Learning Approach for Content-Based Image Retrieval. In to appear in IEEE International Conference on Image Processing (ICIP'10) 2010.

GOMEZ, C. **Scilab**. http://www.scilab.org/ Accessed em: 5 Nov 2007.

GONCALVES, W.N.; MONTEIRO, J.B.; SILVA, J.A., MACHADO, B.B., ODAKURA, V.; PISTORI, H. Multiple Mice Tracking using a Combination of Particle Filter and K-Means. In SIBGRAPI - Brazilian Symposium on Computer Graphics and Image Processing, Belo Horizonte, Brazil, 2007.

INSTITUTE, R.P. **Visualizing the Gradient 2D Applet.** http://www.ibiblio.org/links/applets/appindex/gradientvis.html . Accessed in: 10 Oct 2007.

KISELEV, S. **Converging Lens** Applet. http://www.physics.uoguelph.ca/applets/Introphysics/kisalev Accessed in: 15 Oct 2007.

LINENTHAL, J.; QI, X.. An Effective Noise-Resilient Long-Term Semantic Learning Approach to Content-Based Image Retrieval. In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'08) (pp. 1213–1216), 2008.

Mckinnon, B., & QI, X. Adaptive Wavelet-Based Family Tree Quantization for Digital Image Watermarking. In IASTED International Conference on Signal and Image Processing, Honolulu, Hawaii, US, 2006.

MORETTO, E.G.; REICHEL, M.; PISTORI, H. Rastreamento Dinâmico de Objetos: Um Experimento Did´atico Integrando Conceitos de Hardware e Software. In 7th Workshop on Free Software - WSL, Porto Alegre, Brazil, 2006.

PARAGUASSU, W.A.; PISTORI, H.; PEREIRA, M.C.; JACINTO, M.A. Attributes Reduction applied to Leather Defects Classification. In 23rd SIBGRAPI - Conference on Graphics, Patterns and Images, 2010.

PIRES, P.S.M.; ROGERS, D.A. Free/Open Source Software: An Alternative for Engineering Stu-dents. In Proceed. of FIE - Frontiers in Education, 2002.

PISTORI, H. Computer Vision and Digital Inclusion of Persons with Special Needs: Overview and State of Art. (pp. 33–39). Taylor Francis, 2007.

PISTORI, H., VIANA AGUIAR ODAKURA, V.V., MONTEIRO, J.B., GONÇALVES, W.N., ROEL, A.R., SILVA, J.A., et al. Mice and larvae tracking using a particle filter with an auto-adjustable observation model. Pattern Recogn. Lett., 31(4), 337–346, 2010.

QI, X.; BIALKOWSKI, S.; BRIMLEY, G. An Adaptive QIM- and SVD-Based Digital Image Water-marking Scheme in the Wavelet Domain. In IEEE International Conference on Image Processing (ICIP'08) (pp. 421–424), 2008.

QI, X.; EBERT, J. A Computer Vision-Based Method for Fire Detection in Color Videos. Inter-national Journal of Imaging, 2(S09), 22–34, 2009.

ROYAL,M.; CHANG, R.; QI, X. Learning From Relevance Feedback Sessions Using a K-Nearest-Neighbor-Based Semantic Repository. In IEEE International Conference on Multimedia and Expo (ICME'07), Beijing, China (pp. 1994–1997), 2007.

RUSBAND, W. ImageJ. http://rsb.info.nih.gov/ij/ , Accessed in 6 Jan 2008.

SHKURKO, K.; QI, X. A Radial Basis Function and Semantic Learning Space Based Composite Learning Approach to Image Retrieval. In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'07), Honolulu, Hawaii, US (pp. 945–948), 2007.

SILVA, J.A.; GONCALVES, W.N.; MACHADO, B.B.; PISTORI, H.; SOUZA, A.S.; SOUZA, K.P. Comparison of Shape Descriptors for Mice Behavior Recognition. In CIARP Iberoamerican Congress on Pattern Recognition, 2008.

STADELMANN, D. Watershed Applet. http://bigwww.epfl.ch/demo/jwatershed/index.html. Acesso em 12 Jan 2010.

VIANA, R.; RODRIGUES, R.; ALVAREZ, M.A.; PISTORI, H. SVM With Stochastic Parameter Selection For Bovine Leather Defect Classification. Lecture Notes in Computer Science, 4872. Vishnevsky, E. (2007). The Color Spectrum Applet. http://www.cs.rit.edu/~ncs/color/ , 2007.

WACHT, M.; SHAN, J.; QI, X. (2006). A Short-Term and Long-Term Learning Approach for Content-Based Image Retrieval. In International Conference on Acoustics, Speech, and Signal Processing, Toulouse, France (pp. 389–392).

WEINHEIMER, J.; QI, X.; & QI, J. Towards a Robust Feature-Based Watermarking Scheme. In International Conference on Image Processing, Atlanta, GA, US (pp. 1401–1404), 2006.