Hindawi Publishing Corporation
EURASIP Journal on Applied Signal Processing
Volume 2006, Article ID 45742, Pages 1–3
DOI 10.1155/ASP/2006/45742

*Editorial*

# Performance Evaluation in Image Processing

**Michael Wirth, Matteo Fraschini, Martin Masek, and Michel Bruynooghe**

*Department of Computing and Information Science, University of Guelph, Guelph, ON, Canada N1G 2W1*

Received 3 April 2006; Accepted 3 April 2006

The scanning and computerized processing of images had its birth in 1956 at the National Bureau of Standards (NBS, now National Institute of Standards and Technology (NIST)) [1]. Image enhancement algorithms were some of the first to be developed [2]. Half a century later, literally thousands of image processing algorithms have been published. Some of these have been specific to certain applications such as the enhancement of latent fingerprints, whilst others have been more generic in nature, applicable to all, yet master of none. The scope of these algorithms is fairly expansive, ranging from automatically extracting and delineating regions of interest such as in the case of segmentation, to improving the perceived quality of an image, by means of image enhancement. Since the early years of image processing, as in many subfields of software design, there has been a portion of the design process dedicated to algorithm testing. Testing is the process of determining whether or not a particular algorithm has satisfied its specifications relating to criteria such as accuracy and robustness. A major limitation in the design of image processing algorithms lies in the difficulty in demonstrating that algorithms work to an acceptable measure of performance. The purpose of algorithm testing is two-fold. Firstly it provides either a qualitative or a quantitative method of evaluating an algorithm. Secondly, it provides a comparative measure of the algorithm against similar algorithms, assuming similar criteria are used. One of the greatest caveats in designing algorithms incorporating image processing is how to conceive the criteria used to analyze the results. Do we design a criterion which measures sensitivity, robustness, or accuracy? Performance evaluation in the broadest sense refers to a measure of some required behavior of an algorithm, whether it is achievable accuracy, robustness, or adaptability. It allows the intrinsic characteristics of an algorithm to be emphasized, as well as the evaluation of its benefits and limitations.

More often than not though, such testing has been limited in its scope. Part of this is attributable to the actual lack of formal process used in performance evaluation of image processing algorithms, from the establishment of testing regimes, to the design of metrics. Selection of an appropriate evaluation methodology is dependent on the objective of the task. For example, in the context of image enhancement, requirements are essentially different for screen-based enhancement and enhancement which is embedded within a subalgorithm. Screen-based enhancement is usually assessed in a subjective manner, whereas when an algorithm is encapsulated within a larger system, subjective evaluation is not available, and the algorithm itself must determine the quality of a processed image. Very few approaches to the evaluation of image processing algorithms can be found in the literature, although the concept has been around for decades. A significant difficulty which arises in the evaluation of algorithms is finding suitable metrics which provide an objective measure of performance. A performance metric is a meaningful and computable measure used for quantitatively evaluating the performance of any algorithm. Consider the process of assessing image quality. There is no single quantitative metric which correlates well with image quality as perceived by the human visual system. The process of analyzing failure is intrinsically coupled with the process of performance evaluation. In order to ascertain whether an algorithm fails or not, you have to define the characteristics of success. Failure analysis is the process of determining why an algorithm fails during testing. The knowledge generated is then fed back to the design process in order to engender refinements in the algorithm. This is a difficult process in applications such as image enhancement primarily because there is usually no reference image which can be used as an "ideal" image. The assessment of image quality plays an important role in applications such as consumer electronics. Metrics could be used to monitor or optimize image quality in digital cameras, benchmark and evaluate image enhancement algorithms. There is no single metric that correlates well with image quality as perceived by the human visual system. Selection of an appropriate evaluation methodology is dependent on the objective of the task. In the context of image enhancement, requirements are essentially different for screen-based enhancement and enhancement that is embedded within an algorithm (as a subalgorithm).

The purpose of evaluating an algorithm is to understand its behavior in dealing with different categories of images, and/or help in estimating the best parameters for different applications [3]. Ultimately this may involve some comparison with similar algorithms, in order to rank their performance and provide guidelines for choosing algorithms on the basis of application domain [3]. Assessing the performance of any algorithm in image processing is difficult because performance depends on several factors, as concluded by Heath et al. [4]:

(1) the algorithm itself,
(2) the nature of images used to measure the performance of the algorithm,
(3) the algorithm parameters used in the evaluation,
(4) the method used for evaluating the algorithm.

The ease to which an algorithm can be evaluated is directly proportional to the number of parameters it requires. For example, a segmentation algorithm which has no parameters bar, the image to be processed will be easier to evaluate than one which has three parameters which need to be tailored in order to obtain optimal performance. The nature of the image itself also impacts performance. Evaluation with a set "easy" images may produce a higher accuracy than the use of more difficult images containing complex regions. There are no rigid guidelines as to exactly how the process of performance evaluation should be characterized, however there are a number of facets to be considered [5]: testing protocol; testing regime; performance indicators; performance metrics, and image databases.

The first of these, *testing protocol* relates to the successive approach used to perform testing. There are three basic tenets: *visual assessment*, *statistical evaluation*, and *ground truth evaluation*. The first stage of performance evaluation involves obtaining a qualitative impression of how well an algorithm has performed. For example, when design begins on a new algorithm, a few sample images may be used in a coarse analysis of the usefulness of existing algorithms by means of visual assessment. Visual assessment usually implies comparing the processed image with the original one. Algorithms judged useful at the first stage are investigated in the next stage as to their accuracy using quantitative performance metrics and ground truth data. The "final" stage of evaluation looks at aspects of performance such as robustness, adaptability, and reliability. This process may iterate through a number of cycles. Next is the *testing regime* which relates to the strategy used for testing the images. There are four basic testing categories. The first of these is *exhaustive* testing, which is a brute force approach to testing whereby an algorithm is presented with every possible image in a database to test. Such an approach can be overwhelming, and should be limited to the verification component of the design process. Next is *boundary value* testing, which evaluates a subset of images identified as being representative. The third regime relates to *random* testing in which images are indiscriminately selected. This relates to a more statistically based process of evaluating an algorithm providing more realistic conditions. For instance, is it realistic to test a mass detection algorithm on a database of mammograms containing only malignant masses and assume it works accurately? What happens when the algorithm is faced with a normal mammogram: will it mark a feature as false-positive? The final testing regime concerns *worst-case* testing. What happens when an algorithm processes images containing rare or unusual features? Performance evaluation relies on the use of performance indicators. Such indicators convey the *qualities* of an algorithm. They are often loose characterizations used in the specification of an algorithm, and in themselves are difficult to measure. Typical performance indicators include [5]

(1) *accuracy*: how well the algorithm has performed with respect to some reference;
(2) *robustness*: an algorithm's capacity for tolerating various conditions;
(3) *sensitivity*: how responsive an algorithm is to small changes in features;
(4) *adaptability*: how the algorithm deals with variability in images;
(5) *reliability*: the degree to which an algorithm, when repeated using the same stable data, yields the same result;
(6) *efficiency*: the practical viability of an algorithm (time and space).

Finally there is the notion of the *image database*: which images should be selected to test an algorithm? This relates to the diversity and complexity of the selected images, how many databases are used in the selection process, and the significance of the images to the segmentation task.

The goal of this special issue is to present an overview of current methodologies related to performance evaluation, performance metrics, and failure analysis of image processing algorithms. The first seven papers deal with aspects of performance evaluation in image segmentation, from metrics derived for video object relevance, to skew-tolerance evaluation of page segmentation algorithms and evaluation of edge detection. The last five papers deal with diverse areas of performance evaluation. This includes a methodology for designing experiments for performance evaluation and parameter tuning, the verification and validation of fingerprint registration algorithms, and using performance measures in feedback. As both consumer and commercial electronics evolve, spanning applications as diverse as food processing, biometrics, medicine, digital photography, and home theatres, it is increasingly essential to provide software which is both accurate and robust. This requires a standardized methodology for testing image processing algorithms, and innovative means to tackle quantifying and automatically resolving issues relating to algorithm functioning. The assessment and characterization of image processing algorithms is an emerging field, which has been growing for the past three decades. We hope that this special issue will direct more energy to the problem of performance evaluation, and revitalize interest in this burgeoning field.

*Michael Wirth*
*Matteo Fraschini*
*Martin Masek*
*Michel Bruynooghe*

## REFERENCES

[1] R. A. Kirsch, "SEAC and the start of image processing at the National Bureau of Standards," *IEEE Annals of the History of Computing*, vol. 20, no. 2, pp. 7–13, 1998.

[2] R. A. Kirsch, L. Cahn, C. Ray, and G. H. Urban, "Experiments in processing pictorial information with a digital computer," in *Proceedings of the Eastern Joint Computer Conference*, Washington, DC, USA, December 1957.

[3] Y. J. Zhang, "Evaluation and comparison of different segmentation algorithms," *Pattern Recognition Letters*, vol. 18, no. 10, pp. 963–974, 1997.

[4] M. D. Heath, S. Sarkar, T. Sanocki, and K. Bowyer, "Robust visual method for assessing the relative performance of edge-detection algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 12, pp. 1338–1359, 1997.

[5] M. A. Wirth, "Performance evaluation of image processing algorithms in CADe," *Technology in Cancer Research and Treatment*, vol. 4, no. 2, pp. 159–172, 2005.

**Michael Wirth** has a Ph.D. degree in computer systems engineering from RMIT University in Australia. He is currently an Associate Professor in the Department of Computing and Information Science at the University of Guelph, where his research group is investigating the application of image processing to diverse fields such as cultural heritage, document analysis, food industry, and biomedicine. His past work has included the design of algorithms for preprocessing of mammograms including mammogram segmentation, suppression of artifacts, and registration. He now devotes some of his time to methodologies related to performance evaluation of image processing algorithms. This includes the design of evaluation frameworks, quantitative metrics, and comparative studies of algorithms. The rest of his time is focused on the application of image processing algorithms to emerging domains such as cultural heritage and document imaging. He is investigating the analysis of historical documents and the restoration and enhancement of historical photographs, such as albumen prints. Part of this work is devoted to using techniques such as registration to compare attributes of structures in photographs over time. His interests outside imaging include algorithm design, programing languages, and pedagogy in computer science.

**Matteo Fraschini** is an Assistant Professor of computer engineering in the Department of Medical Science of the University of Cagliari. He is a Member of the GIRPR (Italian Research Group in Pattern Recognition) and MILab (Medical Image Laboratory, University of Cagliari). His research interests include medical imaging, pattern recognition, and signal and image processing.

**Martin Masek** is currently a lecturer in computer programing, and the coordinator of the Games Programing Major at Edith Cowan University, Perth, Western Australia. From 2003 to 2005, he worked as a lecturer in the School of Electrical, Electronic, and Computer Engineering at The University of Western Australia and received his Ph.D. and B.E. degrees from there in 2004 and 1998, respectively. His areas of interest in teaching and research include computer vision and image processing, graphics, and applications to computer game development.

**Michel Bruynooghe** received the Engineering degree from the Ecole Nationale des Ponts et Chaussées (Civil Engineering School in Paris) in 1967. He received a Ph.D. degree in statistical mathematics and a State Doctorat (habilitation) degree in computer science from the University of Pierre and Marie Curie (Paris VI), in 1977 and 1989, respectively. From 1967 to 1973, he was a Research Scientist at the Department of Operational Research at the Transportation Research Institute, Arcueil, France. From 1973 to 1980, he was an Associate Professor at the University of Aix-Marseille II. He was a consultant for "Electricité de France" from 1976 to 1978. Then, from 1979 to 1981, he was a consultant for Solmer Steelwork, Fos-sur-Mer, France. From 1981 to 1989, he was an Associate Professor at the University of Besançon, and for a period of five years (1985–1989), he was a Research Scientist at the Laboratory for Spatial Astronomy (CNRS, Marseille, France). Since 1989, he is a Professor of Computer Science at the University Louis Pasteur of Strasbourg. He was a consultant for Philips Electronics Laboratories from 1992 to 1995. His fields of research are multidimensional data analysis, clustering analysis, statistical pattern recognition, and medical image processing. He is currently doing research in the field of computer-aided detection for the early detection of breast cancer in digital mammography images. Since 1997, he has served as an Associate Editor of the International Journal of Pattern Recognition and Artificial Intelligence.