OUTLINING A RELEVANT UNDERGRADUATE COURSE ON COMPUTER VISION

Gabriel Thomas

Electrical and Computer Engineering Department, University of Manitoba
thomas@ee.umanitoba.ca

Abstract –Having offered a computer vision course as a 4th year undergraduate elective for almost a decade now prompt me to re-evaluate it, not just with the idea of adding new trends seen at international symposia on a yearly basis but evaluating the course taking into consideration what can be seen as needed outside academia and within academia as a preparation for industry jobs and further studies and research. Thus, this paper suggests the different topics that such a course must cover in order to have a strong background on the necessary steps needed to successfully implement a computer vision system. A discussion regarding software and hardware tools involves what I perceive to be an importance towards covering computer vision based on mobile devices.

Keywords: Computer vision, smartphones and mobile computing, research oriented content, industry oriented content.

1. INTRODUCTION

As most Engineering topics, computer vision has been changing fast with time and the discussion about the value of such a course at the undergraduate level has been established a while ago in favor of its incorporation to the curriculum [1]. Nowadays these courses are being offered all over the world and proof of this can be seen from the increase participation of researchers from a variety of countries at symposia such as the IEEE International Conference on Image Processing. At this point then the important questions to ask should involve discussions that can help to determine what constitutes a computer vision course that is competitive and that can offer undergraduate students the tools and knowledge to successfully develop computer vision systems that they most likely will see either in further studies at the graduate level or in industry; this among a possible fiery competition. This paper then starts with a discussion on what topics should be covered, in the understanding that such a decision will be biased from my personal

experience in the field without de-meriting any particular area of computer vision from my part. The discussion then expands to topics such as the use of languages like Java as it has been proposed as an excellent tool before [2]. Furthermore, the paper addresses the implications that new different hardware such as smartphones [3] and mobile devices that are now outselling traditional laptop PCs [4] can have on the choosing of not only the software platform but also the hardware one. Regarding the software, based on how many people use Matlab for research, one can identify the importance of this language but a quick 2012 search on computer vision jobs in industry yielded a list of positions that not necessarily include this software development tool all the time. The paper also mentions the need to select a subset of topics within the many topics presented in popular textbooks such as [5], [6]. Elements of classification and decision making are obviously important [7] and such chapters must be included but because of time constraints some topics must go and the selection seems to be open for discussion. The importance of practical assignments is discussed [8] as well and some examples are shown. In summary, this work discusses the newer challenges and opportunities presented in the area of computer vision and based on this information, proposes a framework for defining an undergraduate course on computer vision that can address these aspects of the field.

2. WHAT AN OUTLINE OUGHT TO INCLUDE

The fact that a software program cannot solve every possible computer vision problem can be seen as good news because this would imply that a good number of opportunities for specialists in this area exist after graduation. It is now compelling to ask what constitutes a complete set of topics that can give students most of the knowledge needed to develop a successful computer vision system. In the understanding that academically it would be impossible to cover all the exciting topics in the field within a semester, let start with the topics available in a textbook such as Gonzalez and Woods [5] and explain the difficult decision on what to include and what

to skip. In this popular textbook we can find the following topics: image fundamentals, image enhancement in the spatial and frequency domains, restoration, color image processing, wavelets, image compression, morphology, segmentation, image description and fundamentals of object recognition. Elements of image reconstruction have been added and CT imaging is part of this book's content. In previous editions of this textbook a similar block diagram as the one shown in Figure 1 was included in which an example such as identifying the hand written address on envelopes was discussed early on to identify each step on Figure 1 that was needed in the solution. For the sake of giving another general example, let consider a computer vision system that can detect and classify peach defects as accomplished in [9]. The solution required different processing steps that involve image enhancement, segmentation, feature extraction and classification by first correcting shading effects of images, afterwards edges for segmentation as well as region growing techniques were used, finally morphological and spectral features such as mean and variance were used in a rule based decision tree and a Bayes classifiers. Another example in which segmentation and image enhancement were addressed is in the solution found in [10] for aligning a hydraulic cylinder prior to being hooked on a conveyor by a robotic arm. The robotic arm was programmed to assume the cylinder's clevis hole to be perpendicular to the horizontal base of the stamping station; if the cylinder was not in this orientation, the arm would unsuccessfully attempt to hook the cylinder on the conveyor line, dropping it to the concrete floor. Figure 2 shows images of the problem in question in which it is clear that topics such as image reconstruction were not part of the solution.

In general terms, the topics in Figure 1 would constitute an excellent minimum set to cover in an undergraduate course. Textbooks such as [11, 12] include all the topics in Figure 1. That does not mean that more specialized textbooks such as [13] that includes more diverse topics such as comparative rendering are not important; the point here is that some, if not all of the elements in Figure 1, would be present in a typical computer vision system. Thus, during the years, my decision has been to have topics such as image compression left as possible topic to be covered based on time availability at the end of the semester.

One thing that I do emphasize is the big square on the centre of Figure 1 that says *knowledge based*; students are encouraged to use common sense and keep the final goal in mind on every needed step from Figure 1, so that for example, image enhancement can be done not to necessary improve the visual quality of an image but to prepare it for segmentation. Take image specification as a technique used in a different way as presented in [5] to

illustrate the point made before. Figures 3 and 4, show the results when using the image enhancement technique, histogram specification, where a specified final histogram is defined in order to be able to enhance the segmentation of the different objects in Figure 4 [14].

Fig. 1. Block diagram depicting the most common elements of a computer vision system.

Fig. 2. Two types of cylinder clevises, both at incorrect positions.

Fig. 3. Original (blue) and specified (green) histograms used to obtain the results shown in Fig. 4.

Note how histogram specification is used not to acquire a good image via contrast enhancement but to improve the segmentation of the bacteria shown in Figure 4. The segmentation algorithm is based on a series of basic morphological operations such as dilation, closing and opening. The edges of the image are calculated first using a Sobel mask and these edges are then modified by the morphological operations to obtain a final segmentation. Note how the modified image presents

smoother regions and how it can detect an object which the original segmentation algorithm missed. Note also how the segmentation appears to follow the objects better.

This thinking outside the box should be encouraged so that students learn new computer vision methods in such a way that they can find unique usages of image processing approaches that can be used on not only to what originally were intended to but also in alternative ways to process digital images. For example, in [15] we can find a very interesting way to use median filtering in the frequency domain to reduce periodic noise. Histogram equalization was used in [16] to improve on the quality of random noise generation. Examples like these can show students that computer vision problems they may face in the future most likely require the application of the knowledge acquired in the course in non-conventional ways. No amount of end of the chapter problems can substitute the understanding of the basic theory well enough to successfully integrate this knowledge into computer vision solutions.

Fig. 4. Segmentation using original and contrast enhanced image based on the specified histogram shown above.

3. THE IMPORTANCE OF PRACTICAL ASSIGNMENTS

As indicated in [8], with today's fast computers and good computer vision libraries, students have the possibility of building complex vision systems within the allocated time of a 3 hour laboratory session or assignment. Practical assignments can help students on the understanding of basic concepts by developing low-level algorithms to obtain solutions to high-level problems. For example, the classification of three different varieties of dates (*dactylifera*) as shown in Figure 5 can be a good practical assignment that looks quite challenging but that it can be implemented with standard computer vision techniques within a reasonable time. The first step on this assignment would be to implement segmentation by thresholding to obtain the white background shown in Figure 5. If Matlab is the software platform, thresholding can be easily used as Matlab has Otsu's method implemented and it is also explained in the textbook [5]. Of the many different thresholding techniques, one can also teach Minimum

Error Thresholding [17] as it has been found to be one of the best approaches on a survey published in [18].

Continuing with the classification example, the shape of these date fruits cannot really help on the definition of useful features. One can infer that the images coming from the dates named Fard appear darker and that at least the mean value would be a good feature to discriminate this type. This hypothesis of using statistical features can be proposed after examining all the different images available, a process that is highly recommended, that of understanding the problem thoroughly. Figure 6 shows the mean and skewness values, note how the mean would help on classifying the darker Fard type as indicated before and how skewness may be the feature that can help on identifying the Kalas class.

Fig. 5. Three different varieties of dates: First row Fard, second row Madina, third row Kalas

At this point the last step is to implement a classifier that will use the statistical features computed previously. Numerical classifiers such as minimum distance, Bayes and neural network based ones are discussed in [5]. Using the mean, entropy, skewness and kurtosis and training a two layer neural network, classifications better than 90% can be quickly obtained. Thus, this example can be implemented using the neural network pattern recognition tool available in Matlab that students can easily use by selecting a series of parameters such as the number of hidden neurons taking advantage of the graphical user interface that Matlab provides to define such classifier as seen in Figure 7.

In general, looking at images can help on intuitively deciding which type of features to be used and looking at the feature values can help on deciding which ones of these features can actually be of good practical value. Nevertheless, as one calculates more features by either adding different type of features or simply by calculating those same features on each of the colour planes, the number of features increases and it might be good to include in a computer vision course a discussion on

feature selection. Once again Matlab has several methods to do so and a brief introduction can help students on using such analysis. It can be argued that for example the same approach has to be taken for the neural network section of the course which briefly introduces a popular and powerful classifier with enough discussion for students to feel comfortable designing a neural network architecture.

Fig. 6. Statistical features in the form of mean (top) and kurtosis (bottom) values extracted from images such as the ones shown in Fig. 5.

4. ON THE DIFFERENT SOFTWARE AND HARDWARE OPTIONS

As mentioned in the introduction, a decision has to be made regarding the software tool that students will be using in the course. Based on how popular Matlab is at the research level, one can see this by noticing that the great majority of examples in computer vision and image processing journals are generated with this software, one can think that Matlab is a requirement for most computer

vision jobs. This does not necessarily translate to what can be found in industry.

Fig. 7. Matlab neural network pattern recognition tool.

To illustrate this, here is a very short list of what companies such as Siemens and Amazon.com as well as smaller firms are looking for regarding computer vision software development skills: C++, Python, OpenCV, OpenGL, and Java. Recently companies are also looking for CUDA, GeForce, and graphics accelerators expertise. Before even trying to commit to any of these tools for a computer vision course, the future may point towards mobile computing as smartphones [3] and mobile devices are now outselling traditional laptop PCs [4]. For example, Stanford University has introduced a series of lectures on Mobile Image Processing on Android: http://stanford.edu/~dmchen/documents/EE3682011_GuestLecture_Part1.pdf. Being in Canada, I joined the BlackBerry Academic Program and received a series of BlackBerry phones and RIM tablets which developer tools include HTML5, Android, Adobe Air and BlackBerry Theme Studio. Based on this variety of alternatives, I can only mention what my own personal decision has been: take advantage of Matlab GPU computing support for Nvidia CUDA-Enabled GPUs and design a series of labs that will accelerate solutions that do not use GPU computations. Thus, Matlab will remain the main computer vision tool to be used in the course. Furthermore, remote access to computers with the necessary graphic accelerators can encourage the topic on coupling cloud computing with mobile devices especially due to computer power limitations that mobile exhibit [19]. Because of the popularity of Android, offer a series of lectures and assignments on mobile computing based on programming for Android. Once again, these are personal choices but it is important to note that mobile computing and GPU programming are two important topics that ought to be considered in a computer vision course.

5. CONCLUSIONS

The paper tries to identify a series of topics that are needed for undergraduate students to take on the designing of a complete computer vision system at the software level. Based on new hardware trends, a discussion is also offered on what new topics can be included that are so new that traditional computer vision textbooks will not necessarily include chapters on these new developer platforms. As the importance on mobile devices grows in the future, Android or HTML5 can be introduced in the course and by using Matlab as the main software tool, GPU programming can also be added if the correct graphic accelerators supported by Mathworks are available to students.

References

- [1] G. Bebis, D. Egbert, M. Shah, "Review of computer vision education," *IEEE Transactions on Education*, vol. 46, no. 1, pp. 2- 21, February 2003.
- [2] D. Sage, M. Unser, M. "Easy Java programming for teaching image-processing," in *International Conference on Image Processing*, vol. 3, pp. 298-301 vol. 3, 2001.
- [3] A. Hoover , "Computer vision in undergraduate education: modern embedded computing," *IEEE Transactions on Education* , vol.46, no.2, pp. 235- 240, May 2003.
- [4] M. Rasoff, iPads Are Outselling Desktop PCs, And Are Now Equal To 17% Of The PC Market, Business insider, January 24, 2012
http://articles.businessinsider.com/2012-01-24/tech/30658335_1_tablet-market-ipad-market-pc-market
- [5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Upper Saddle River, New Jersey, Prentice Hall, 2002 (2nd ed.), 794 pp. {ISBN: 0-201-18075-8}
- [6] B. A. Maxwell, "A survey of computer vision education and text resources" *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 15, no. 5, pp. 757-773, 2001.
- [7] M. Georgiopoulos, R. F. DeMara, A. J. Gonzalez, A. S. Wu, M. Mollaghasemi, E. Gelenbe, M. Kysilka, J. Secretan, C. A. Sharma, A. J. Alnsour, "A Sustainable Model for Integrating Current Topics in Machine Learning Research Into the Undergraduate Curriculum," *IEEE Transactions on Education* , vol. 52, no. 4, pp. 503-512, Nov. 2009.
- [8] Y. Gonzalez and J. Rocha, "Some Practical Assignments in Computer Vision", In *Proceedings of the Workshop on Education and Practice in Artificial Vision*, pp. 101-104, 2003.
- [9] B. K. Miller and M. J. Delwiche, "Peach defect detection with machine vision," *Transactions of the ASAE*, vol. 34, no. 6, pp. 2588-2597, 1991.
- [10] G. Thomas, J. E. Kaye, R. Jayas, and C. Kaye, "Cylinder Rotational Orientation Based on Circle Detection," *Springer Lecture Notes in Computer Science*, Vol. 3656, pp. 499-504, 2005.
- [11] M. Sonka, V. Hlavac, R. Boyle, *Image Processing, Analysis and Machine Vision*, 2008 (3rd ed.), Thomson, 829 pp. {ISBN: 978-0-495-08252-1}
- [12] C. Solomon, T. Breckon, *Fundamentals of Digital Image Processing, A Practical Approach with Examples in Matlab*, Wiley-Blackwell, 2011, 328 pp. {ISBN: 978-0-470-84473-1}
- [13] S. Mann, *Intelligent Image processing*, Wiley series in adaptive and learning systems for signal processing, communications, and control systems, 2002, 339 pp. {ISBN: 0-471-40637-6}
- [14] G. Thomas, "Image Segmentation Using Histogram Specification," in *IEEE International Conference on Image Processing*, pp. 589-592, San Diego California, October 2008.
- [15] I. Aizenberg, C. Butakoff, "Frequency domain median-like filter for periodic and quasi-periodic noise removal," in *SPIE Proceedings of Image Processing: Algorithms and Systems*, vol. 4667, pp. 181–191, January 2002.
- [16] J. J. Mijares-Chan, B. Sharma, J. Lv, G. Thomas, R. Thulasiram, and P. Thulasiraman, "True random number generator using GPUs and histogram equalization techniques," in *IEEE International Conference on High Performance Computing and Communications*, Banff, Canada, September 2011.
- [17] J. Kittler and J. Illingworth, "Minimum Error Thresholding," *Pattern Recognition*, Vol. 19, No. 1, pp. 41-47, 1986.
- [18] M. Sezgin and B. Sankur, "Survey Over Image Thresholding Techniques and Quantitative Performance Evaluation," *SPIE Journal of Electronic Imaging*, Vol. 13, No. 1, pp. 146-168, January 2004.
- [19] R. Ferzli, I. Khalife, "Mobile cloud computing educational tool for image/video processing algorithms," *IEEE Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE)*, pp.529-533, January 2011.