Computer Graphics Education in Different Curricula: Analysis and Proposal for Courses

Eric Paquette [*,1]

LESIA - Software & IT Engineering Dept., École de technologie supérieure, Montréal, Canada

Abstract

This paper studies how Computer Graphics is taught and proposes a course on 2D Computer Graphics and Image Processing as an alternative to the traditional 3D Computer Graphics course. This unconventional course is motivated by an analysis of more than 70 Computer Science curricula. This analysis considers many aspects: Computer Graphics, Image Processing, and Human-Computer Interaction courses; curricula such as Computer Engineering, Computer Science, Information Technology, and Software Engineering; the difference between introductory and advanced courses; and universities known for their leadership in Computer Graphics as well as mainstream universities. The analysis suggests that given the different types of universities and curricula, there should be more alternative courses tailored to the needs of particular curricula. Developing such courses can be difficult and time consuming, so a methodology is proposed to describe a course with information useful for others who could be selecting it or who could be putting it in practice. This methodology is put in practice with the description of a course on 2D Computer Graphics and Image Processing.

Key words: Education, Computer Graphics, Image Processing, Human-Computer

Interaction, Computer Science Curricula

PACS: 01.40.Di, 01.40.Gm,

1 Introduction

Computer Graphics (CG) is a vast, important, and popular discipline. From its beginning around 1970, CG is now a mature discipline built on a strong mathematical basis and with applications in an ever increasing number of areas. This is reflected in the undergraduate curricula of various other disciplines, such as physics, engineering, and architecture, which include an optional introductory CG course [1] in some of their curricula. In Computer Science, CG occupies an important place for a long time [2] and now it is well in place with over 82 percent of the undergraduate curricula having an optional CG course. Furthermore, there are seven percent of specialized curricula where one or more CG courses are mandatory.

CG is interesting because its results are visual and often impressive. It is thus a discipline that attracts many students, but it is also useful for students as it provides them with suitable knowledge of the algorithms they will use to create graphical applications and representations of all sorts of data.

* Corresponding author. Address: Prof. Eric Paquette, Software & IT Engineering Dept., École de technologie supérieure, 1100, Notre-Dame O., Montréal, Québec, Canada, H3C 1K3

Email address: `Eric.Paquette@etsmtl.ca` (Eric Paquette).

URL: `www.logti.etsmtl.ca/profs/paquette` (Eric Paquette).

[1] This research was supported by a PSIRE-Enseignement grant from École de technologie supérieure and a Discovery grant from NSERC.

It is thus worth studying how CG is presented in different curricula and the relationships between CG and other disciplines. This paper focusses on relationships between CG and two disciplines with which it has strong links: **Image Processing (IP)** and **Human-Computer Interaction (HCI)**. Many curricula from five different countries were looked at to gain some insight at how CG is taught. Even though this study of CG focusses on curricula related to Computer Science, it encompasses Computer Engineering, Information Technology, and Software Engineering. It thus includes a wide variety of approaches and contexts in which CG is taught.

The analysis of CG education in Computer Science curricula reveals that even though there is an overlap and a close relationship between CG, IP, and HCI, these topics are relarely presented together such that each discipline would account for an equivalent portion of a single course. Furthermore, there are significant differences in how CG related courses are distributed in the different types of curricula, suggesting that there should be more approaches to teaching CG, tailored to different curricula. In this sense, this paper proposes an approach to detail new CG courses, and applies it to a course on 2D CG and IP. This approach to presenting the details of CG related courses, provides useful information for teachers as well as committees responsible for the development of new curricula. Finally, the paper ends with a look at the different approaches used to introduce CG in three different curricula at our university. Therefore, our main contributions can be summarized as:

- A study and an analysis of CG education in different curricula.
- An analysis of relationships between CG, IP, and HCI.
- An approach to detail CG related courses applied to a course on 2D CG and IP.

- A description of practical experience with different curricula, each with its adapted approach to introduce CG.

2 How Computer Graphics is Presented

CG is typically presented in an introductory course that focusses on 3D [3–5]. Such typical courses represent 61 percent of the introductory CG courses. These courses on 3D also have to deal with IP since the main goal is image synthesis. The introductory courses thus have to deal with typical IP topics such as pixels, aliasing, filtering, image composition, and image enhancement. Creating and interacting with 3D CG is a difficult task and introductory courses also have to deal with typical HCI topics such as rapid feedback, human perception of 3D space, complex tasks of moving in 3D space, and creating 3D objects by looking at a 2D computer screen. The relationship between CG, IP, and HCI are thus seen in the topics that overlap between these disciplines. These relationships can also be seen in Figure 1 which presents the correlation between the presence of CG, IP, and HCI in the same curriculum. It can be [Fig. 1] observed that there is a tendency to have both CG and IP courses in the same curriculum (correlation of 0.22). In the same way there is a tendency to have both IP and HCI courses in the same curriculum (correlation of 0.20). Yet at the same time, there does not seem to be any relation between CG and HCI courses, as their occurrence in the same curriculum is not correlated (correlation of -0.01). We will thus focus on the CG and IP relationship in the description of such an hybrid course in Section 3.

As a basis to the work presented in this paper, undergraduate Computer Science curricula were looked at with a focus on courses related to CG. This collection is a work in progress. The long-term plan is to have a collection of links to typical introductory and advanced courses that can be investigated when a course is to be created or evolved to a different focus. This set of CG courses in Computer Science curricula is composed of data from 5 countries, 44 universities, 74 undergraduate Computer Science curricula, and 176 courses related to CG. Details on the dataset can be found in Appendix A.

With respect to other works at surveying CG courses [3,6], this survey does not try to capture all the topics presented in every course, but focusses on CG courses at a higher level and the context in which they are presented: traditional 3D CG introduction, advanced courses, courses that mix CG with IP or HCI, and stand alone courses on IP and HCI. The curricula that were analysed were related to Computer Science and include Computer Engineering, Information Technology, and Software Engineering.

In the context of an undergraduate Computer Science curriculum, various authors agree that there are many ways in which CG can be introduced [7–10]. The distribution of first courses closely related to CG, IP, and HCI is presented in Figure 2. There are so many topics to cover in Computer Science [Fig. 2] (programming, software engineering, databases, networking, computer architecture, artificial intelligence, *etc.*) that it may not be appropriate to have courses on CG, courses on IP, and additional courses on HCI. Figure 3 presents the average number of CG, IP, and HCI courses in different curricula. As it [Fig. 3] can be seen, the average number of courses is often close to or below one. An explanation for this is that some curricula already have so many courses that it becomes difficult to add courses on CG, IP, and HCI. It would thus be of great interest to cover some parts of CG with another discipline. Even though there are links between CG, IP, and HCI, and there is an interest in mixing these topics together, only three percent of the introductory courses do so (see Figure 2).

Another fact that can be seen in Figure 3 is the substantial differences in the average number of CG and IP courses in the different types of curricula. This suggests that these curricula, even though they are all closely related, have different goals and audiences. It would thus be of interest to adapt CG courses to the particular context in which they are taught.

Despite the interest in having CG courses in Computer Science and even curricula outside of Computer Science, teaching CG may become difficult since it is a vast discipline. Typical courses present only an introduction [11] and many advanced topics such as animation [12] and global illumination [13] are not covered to an appropriate depth, because of the lack of time to do so. Thus, apart from the introductory courses, additional courses exist that require the students to follow a prerequisite CG course. It is interesting to note that the proportions of introductory CG courses with respect to advanced CG courses varies significantly from one type of curricula to another (see Figure 4). This [Fig. 4] is again in favor of tailoring the CG courses to their particular curricula.

There are still some curricula that do not have a CG course and in some universities, the CG courses might not be taught by CG specialists. It is thus important to look at the differences between typical universities and universities where there is a strong leadership in CG. Appendix A details how this concept of CG leadership is defined and how universities were randomly selected to define the group of typical universities. As can be seen in Figure 5 and as could be expected, CG leadership universities offer many more CG courses, as well as more IP and HCI courses. It was already noted when examining [Fig. 5]

Figure 3, that the average number of courses per curriculum is often close to or smaller than one. When considering this with respect to typical universities, the average number of courses per curriculum is even much lower, which is again in favor of covering more than one discipline in a single course. Figure 5 also shows that the number of advanced courses is significantly smaller in typical universities. In fact, even the proportion of advanced courses decreases as can be seen in Figure 6. This means that introductory courses for such universities should concentrate on providing a good coverage of CG, IP, and HCI, since there are very few or even no advanced courses on these disciplines. [Fig. 6]

From this analysis of how the CG, IP, and HCI disciplines are presented in different curricula, two major conclusions can be established: (1) typical universities offer less courses in these disciplines, (2) the number and types of courses varies significantly between curricula types and university types (typical vs CG leadership). For more curricula to include an appropriate course covering these disciplines, more courses that mix CG, IP, and HCI should be proposed. Yet, typical universities might need guidance with respect to theoretical and practical concerns, since it is quite possible that none of the faculty members are CG specialists. Therefore, one way to help others to select, prepare, and give a good course, is to detail the many aspects that have to be considered:

(1) Aims
(2) Prerequisites
(3) Content
(4) Relationships with CG
(5) Textbooks
(6) Assignments

(7) Possible advanced courses

(8) Benefits for students and teachers

(9) Challenges and drawbacks

3 Computer Graphics and Image Processing Course

In this section, a 2D CG and IP course [14] is detailed with respect to the elements identified in the previous section. At the present time, only 3 percent of the courses on CG introduce it together with IP. It is proposed that combining the 2D aspects of CG with IP in a single course has many benefits. It is our hope that with detailed descriptions such as the one found here, alternative ways of introducing CG will be much easier to setup in every university and in many curricula.

3.1 Aims

The course provides students with knowledge useful for 2D content creation, for the acquisition of images, and for the reproduction of 2D content on different media. The students will get an understanding of the algorithms implemented in common software such as Photoshop® and CorelDRAW®. While understanding the algorithms, the student will be able to select the proper tools and parameters to create appropriate visual representation of information.

The course also presents the theory needed to understand how to appropriately capture and reproduce 2D content. The theory will enable the students to understand the limitations and constraints of capturing 2D content with digital cameras, by scanning from a printed copy, by scanning standard camera film, *etc.* The students will understand how their eyes perceive shapes and colors, and thus will enable them to understand the different costs and output quality of various media such as prints from inkjet or laser printers, computer screen, lithography, *etc.*

3.2 Prerequisites

Both CG and IP require students to have basic skills in calculus, linear algebra and trigonometry. It is important to note that bringing the two disciplines together does not put additional constraints on the target audience compared to the traditional 3D CG introductory course. This course on 2D CG and IP is an introductory course in the sense that it does not require any prior knowledge in these two disciplines. Finally, depending on the goal of the curriculum, little to moderate knowledge of programming will be required for the assignments.

3.3 Content

The 2D aspects of CG such as vector primitives, 2D curves, halftoning, 2D transformations, even though they are sometimes pointed out as outdated ways to introduce CG [15], are important in creating 2D content. Acquisition and reproduction of this content also requires knowledge of 2D CG and of the IP discipline.

The 2D CG and IP course presents particular topics of traditional CG and IP courses. Obviously it cannot cover all the material of the traditional courses, but it presents topics that form a coherent view of the two disciplines. Here is a brief list, summarizing the topics it covers:

perception human visual system, sensation of color

color models, transformations (brightness, contrast, gamma, histogram equalization), composition (alpha blending, mathematical and logical operations)

sampling pixels, quantization, aliasing, antialiasing

acquisition and reproduction digital camera, scanning, printing, displaying, halftoning

vector primitives ellipses, rectangles, polylines, curves, *etc.*

rasterization mid point, scan conversion, painter's algorithm

filling boundary and flood fills

image filtering filtering in the spatial domain (blur, sharpen, median filter, *etc.*) and in the frequency domain

transformations affine transformations, homogenous coordinates, composition

3.4 Relationships

This section presents the theoretical relationships between CG and IP. These relationships strengthen the logic of combining these two disciplines in a single course. Since the two disciplines are so closely related, the presented classification is a helpful tool for comparison, but not a strict classification.

3.4.1 Computer Graphics and Image Processing Shared Topics

Table 1 presents the basic relationships between CG and IP. Images are the [Tab. 1]
predominant relationship, and whereas CG is mainly used while (or before)

the images are created, IP is applied after an image is available to process. How the human eye perceives colors, how colors are represented in digital forms, and how vector graphics are sampled to be displayed on a monitor are few examples of aspects of these topics that are common to CG and IP.

3.4.2 Relationships Between Topics

Many topics are advanced topics in one of the CG or IP disciplines (Table 2). This section presents these advanced topics that have links to the other discipline, which is useful for the definition of advanced courses based on an introductory 2D CG and IP course. Shading and projection are basic topics in CG, but they relate to advanced IP topics of shape from shading and stereo vision. Contrast enhancement and image compression with wavelets are basic topics in IP, but they relate to advanced CG topics of High Dynamic Range Imagery (HDRI) contrast enhancement and multiresolution surface editing with wavelets. Other topics are advanced in both CG and IP, such as texture analysis and synthesis, and 3D reconstruction from images. [Tab. 2]

3.4.3 Computer Graphics Topics for a 2D Course

In every introductory CG course, many topics such as curves and transformations are first introduced in 2D and then extended to 3D. While these 2D topics may have little to do with IP, they fit well with the 2D context of images. Furthermore, 2D objects such as curves are often converted to pixels and images before they are displayed. Images can also be manipulated by 2D transformations, so this topic is also meaningful in the context of IP. Table 3 presents CG topics that are appropriate for a 2D CG and IP course and their extension to 3D graphics. It shows that bringing the same topic from 2D to 3D is not a trivial extension. In the 3D world, curves become surfaces, transformations become projections, *etc.* Presenting these topics in a 2D course conveys important knowledge while leaving many new challenges to the students when the topics are presented in the 3D space. This table is thus helpful in preparing an advanced course on 3D that would follow the 2D CG and IP course. [Tab. 3]

3.5 Textbooks

Eventhough some textbooks cover CG and IP, we were not satisfied with any single textbook. Students could obviously buy two textbooks, one for each discipline, but this would become prohibitively expensive. Another good solution is to rely on course notes built from extracts of various textbooks. The proposed course notes use six textbooks [11,16–20] to cover the required topics as detailed in Table 4. This is an inexpensive solution and these notes respect the Canadian copyright law which our university must follow and which is similar to the "Fair Use" of the U.S. copyright law. This approach should thus be appropriate in many universities that have to follow a similar copyright law. [Tab. 4]

3.6 Assignments

Table 5 presents the assignment topics. These were selected because they provide a good balance between 2D CG and IP, as well as vector and raster graphics. This greatly reinforces the understanding of the links between these [Tab. 5] topics.

To further help the students in focussing on the theory while they are doing the assignments, a framework [21] that allows 2D CG and IP to be coded in a single application was developed. The framework does not present a working system, such as an image editing software, but gives a context in which algorithms are implemented with little programming effort and can be controlled with a readily working interface as can be seen in Figure 7. When object-oriented design can become a difficulty, example concrete classes are included in the system to facilitate the implementation of the classes relevant to an assignment. The framework design is thus focussed on ease of modification, not efficiency or rigorous object-oriented design. To facilitate its use by other teachers, this framework is developed in Java and is available under an *open source* licence on SourceForge.net. [Fig. 7]

3.7 Possible Advanced Courses

As we began to see it in Sections 3.4.2 and 3.4.3, the 2D CG and IP course is a good starting point for advanced courses. In an advanced course on IP, sophisticated IP topics could be introduced and the course could include vision and object recognition topics. A natural advanced course would be to pursue in the 3D space and rendering from 3D to 2D. Other advanced topics such as non-photorealistic rendering, animation, and advanced rendering techniques would also be appropriate choices.

3.8 Benefits

The previous sections presented many reasons why a 2D CG and IP course is an interesting approach. In this section we focus on the benefits of this type of course.

3.8.1 Benefits for Students

For students, the CG and IP disciplines fit well in terms of the applications they fulfill. While they are studying and in their future career, students in Computer Science will have to create 2D content in different forms such as technical talks and reports, data visualization, user interface, or promotional material. These types of 2D content use text but also heavily rely on 2D graphics and images. Acquisition of images from different sources and reproduction on different media impose constraints that become much easier with a background in 2D CG and IP. Knowing how 2D CG and IP algorithms work also enables the students to harvest the power of 2D CG and IP software and to be much more efficient at editing 2D content. The students get to be much more productive with the 2D features of common software such as Word® and PowerPoint® and they also understand more of the functionalities of software dedicated to 2D content such as Photoshop® and CorelDRAW®.

The students acquire knowledge that is useful in many applications that require the creation, manipulation, or reproduction of 2D content. Specific types of 2D content creation and manipulation are worth mentioning. Today, almost every company has its Web site. The 2D content of such pages poses both practical and theoretical constraints. Presenting data to the end user is important in almost every software. This data will almost inevitably be presented on a 2D display. Thus knowledge of how to create appropriate 2D images and vector graphics is an important asset. In computer games, most of the user interfaces and special effects are made in 2D. Fast computation, transformation, and composition of images is crucial in such applications. In these applications, basic knowledge of the 2D CG and IP course topics is essential to every programmer.

The course gives the students a good basis in both CG and IP. This enables them to interact with specialists in these two disciplines. Finally, the 2D CG and IP course is interesting for students, since it presents the theory that governs how they perceive the world and how digital images are approximate views of the real world.

3.8.2 Benefits for Teachers

For teachers, the theoretical relationships between the topics require quite similar background knowledge from the students. The close relationship also allows the two disciplines to be easy to present together. Since there is a significant overlap between the two disciplines, presenting them in a single course has the benefit of avoiding the duplication that would occur with two traditional courses, one on each discipline. The course provides an interesting platform on which to build advanced courses as seen in Sections 3.4.2, 3.4.3, and 3.7. Furthermore, it gives more time to present topics which, in an introductory course to CG, are typically presented in a rush, since introductory courses tend to focus more and more on the 3D techniques [22]. Such topics include the difference between vector and raster graphics and how vector graphics are rendered to raster representation, halftoning techniques, color models, and filling algorithms. These topics, even though they are important to CG, are not essential to cover in depth in a 3D course and are often barely touched.

3.9 Challenges and Drawbacks

A first practical challenge of such a course is that there is no appropriate textbook that covers both disciplines as seen in Section 3.5. The textbook by Watt and Policarpo [16] covers CG, IP, and vision, but does not present these topics in sufficient detail. In CG textbooks, low pass filtering is briefly presented and all other spatial filters are ignored. The presentation of the frequency domain, of image transformations, such as histogram equalization and median filtering, is also inexistent or too brief. In IP textbooks, halftoning methods, vector primitives, curves, conversion from vector graphics to raster graphics, geometric transformations, and filling are not presented. The proposed solution to use course notes built from five different textbooks works in practice, but the students have to adapt to the fact that the text is not completely cohesive.

The topics covered in the course, even though there is no doubt they are related, can seem too independent. Raster and vector graphics, like 2D CG and IP, may seem unrelated opposites. This type of dichotomy between related topics is inevitable and is present in traditional 3D CG introductory courses. If the teachers are aware of this difficulty, they can point out and explain the links between these topics. An approach to bring the topics closer together uses assignment topics balanced between 2D CG and IP, as well as vector and raster graphics.

In the context of having a 2D CG and IP course as an introductory course followed by a 3D CG course, the most problematic aspect is that the 2D CG and IP course is a prerequisite for the 3D CG course. Thus students cannot directly follow the 3D course without going through IP. Creating 2D content is so important and relies so heavily on 2D CG and IP that, in the context of many curricula, it should outbalance this problem. Another potential problem is that some topics, such as transformations, curves and primitives presented in Section 3.4.3, are split between two courses. Even though the topics were already presented in the 2D context, the teacher of the 3D course will have to refresh the students memory before extending the 2D concepts to 3D. However, this problem is common in any course that further develops material learned in a prerequisite course.

4 Computer Graphics in Different Curricula

A 2D CG and IP course is one approach to introduce CG, but many others have their places depending on the particular context. The ÉTS Engineering School is a nice environment to study different ways CG can be introduced, since three curricula have different introductory CG courses.

The 2D CG and IP course presented in the previous section is almost identical to the GTI410 course of the Information Technology engineering curriculum. It presents a practical academic example where a 2D CG and IP course is an important tool for pedagogy. The Information Technology engineering students gain abilities in the five disciplines of engineering, Computer Science, CG, networking, and business. In this context, the GTI410 course fills the most important part of the multimedia content creation knowledge required by the CG, networking, and business disciplines. A 2D CG and IP course brings a definitive advantage to Information Technology, which represents 14 percent of the curricula from our dataset.

In the Electrical Engineering curriculum, the ELE615 introductory CG course is a mixture of CG and HCI. The future electrical engineers need the basic knowledge of interface design and the CG topics serve different needs such as a challenging application of computer programming, application of real-time system, and an introduction to visualisation aspects.

The Software Engineering curriculum presents many theoretical aspects of software processes and methodologies. In this curriculum, the LOG750 introductory CG course is a traditional 3D CG course. The course is an exciting and challenging application of the various skills learned in the curriculum such as HCI, object-oriented design, data structures, testing, and mathematics. Image synthesis and visualization are other interesting assets for the future software engineers.

5 Conclusion

This paper first presents some important relationships between CG, IP, and HCI. These disciplines have many topics in common and some of these are sometimes introduced in the same course. Unfortunately, most courses put a strong emphasis on one discipline and only briefly present topics from the others.

Computer Science-like curricula were surveyed and analysed to try to understand if there is a place for more courses that merge CG, IP, and HCI together.

The survey and analysis captured the variations between:

- Curricula types
- CG, IP, and HCI disciplines
- Introductory and advanced courses
- Typical universities and universities in the leadership of the CG discipline

The analysis first confirms that the relationship between CG, IP, and HCI is also apparent in the way curricula are design and how courses in these disciplines often occur in the same curriculum. The average number of courses in these three disciplines suggests that there is room for many courses that combine two disciplines together. Yet, it is observed that very few such courses appear in the various curricula in the presented survey.

To increase the ease with which courses tailored to particular curricula can be put in practice, it is suggested that curricula should be carefuly described with respect to many aspects. These aspects will ease the selection of courses that best meet the requirements and constraints, will help in putting the course in practice, and suggest possible advanced courses. This methodology is put in practice in describing a course on 2D CG and IP. During their studies and in their future career, typical computer scientists are more likely to develop 2D content than 3D content. Therefore, such a course should be particularly helpful in many curricula where the knowledge required to acquire, edit, and reproduce 2D content is important. To help teachers in putting such a course in practice, the course content is described and suggestions regarding textbooks and assignments are made.

5.1 Future Work

As was mentioned, a long-term goal is to have a collection of typical introductory and advanced courses that can be investigated when a course is to be created or evolved to a different focus. The 2D CG and IP course is quite well defined and analysed, but it would be interesting to look at other disciplines that could be mixed with CG. Disciplines such as HCI, CAGD, and art share some topics with CG. It would be interesting to study if the relationships between CG and such disciplines are as tight as the relationships between CG and IP.

Many interesting results came from the analysis of the survey of CG, IP, and HCI courses. Many other important observations could be made from the interactions between the different types of courses, curricula, and universities. It would be of great interest for the CG discipline to augment the dataset with more universities and more criteria. This would lead to a better understanding of how this discipline is presented to the students who will lead its direction in the upcoming years.

A Computer Graphics Courses Dataset

The dataset used in this paper was developed by gathering information about Computer Science-like curricula from five countries. In each curriculum, information was gathered on courses in the CG discipline and the related disciplines of IP and HCI. The process of collecting data relied on the descriptions of curricula and courses found on the Web. Using data from paper publications of the universities could provide similar data, but would prevent others from quickly examining our dataset in more details. Since the data is of interest to people in Computer Science, it was also decided to restrict ourselves to universities from English-speaking countries. This should be adequate since people in Computer Science often rely on English textbooks, scientific journals, and conferences.

The specific attributes of our dataset are detailed in this section. Some of these attributes present criteria that do not allow a strict classification, but that are useful to get some insight in how and where CG, IP, and HCI are taught. At the early stages of building this dataset, universities were selected based on a subjective appreciation of their leadership in CG and also on queries on Web search engines for Computer Science curricula. The selection then became much more rigorous as objective criteria were introduced. To identify the universities that have a strong leadership in the CG discipline, the universities involved in three or more papers presented at the 2004 annual ACM SIGGRAPH conference were selected. From the list, few universities had to be discarded because no reliable information could be found on the Web or the information was not in English. Thus eleven universities were classified as CG leadership universities (Stanford, Washington, MIT, Illinois, Carnegie Mellon, New York, British Columbia, Columbia, Princeton, California at Berkeley, Toronto).

It is also important to investigate how CG is taught at the typical universities. To gather another group of eleven universities, but this time by randomly selecting them, the Yahoo!® education directory of Colleges and Universities was used. It provides listings of universities for each country out of which we could randomly pick. To have a meaningful representation with respect to the population of each country, we used the population as a weight and rounded to the upper integer. This resulted in a distribution of universities per country as follows: Australia (1), Canada (1), New Zealand (1), United Kingdom (2), United States of America (6). To randomly select the precise universities based on the lists of Yahoo!® , the lists were first shuffled and then the entries were looked at to find the first one with a Computer Science-like curriculum described in appropriate details.

Computer Science-like curricula were selected in each university. These were classified in four curricula types: (1) Computer Engineering, (2) Computer Science, (3) Information Technology, and (4) Software Engineering. The process of selecting the curriculum type for each curriculum was based on the title of the curriculum and if this was insufficient to decide, the course list and description were analysed and the course was classified to the best of our interpretation. Inside each curriculum, the courses related to CG, IP, and HCI were selected. Course domains were identified to allow the classification of courses as: CG, IP, HCI, and CAD, as well as hybrid courses of CG & HCI and CG & IP. Other courses clearly related to CG could not be classified in these domains and were added to the "CG Related" category. As for the curriculum types, this classification is not strict and was done based on the course description and at the best of our interpretation. Each course also has two properties: "Traditional" and "Introductory". The traditional property corresponds to courses that follow well known textbooks such as Foley et al. [11] for 3D CG or Gonzalez and Woods [19] for IP. This classification is not completely strict, but is still quite objective. Finally, the introductory property represents the fact that the course does not require a prerequisite course on the same topic. For example, the standard 3D CG course based on textbooks such as Foley et al. [11] typically has prerequisites in mathematics and programming, but none in CG. This criterion is strict as long as the information found on the Web is accurate.

The Web resources related to this dataset can be found here:

- The current version of our dataset
  http://www.logti.etsmtl.ca/profs/paquette/cgcourses/
- Yahoo!® education directory
  http://dir.yahoo.com/Education/

B Acknowledments

We wish to thank J. Forcier and T. Richer for their work on the database application. We also want to thank C. Barré-Brisebois, J. F. Barras, F. S. Bois, and M. El Ghaouat for their work on the J2DCG framework.

References

[1] S. Cunningham, Re-inventing the introductory computer graphics course: Providing tools for a wider audience, in: J. C. Teixeira, W. Hansmann, M. McGrath (Eds.), Graphics and Visualization Education Workshop (GVE '99) Proceedings, 1999, pp. 45–50.

[2] M. Ohlson, The role and position of graphics in computer science education,

in: Proceedings of the seventeenth SIGCSE technical symposium on Computer science education, 1986, pp. 232–237.

[3] R. Wolfe, Bringing the introductory computer graphics course into the 21st century, Computers & Graphics 24 (1) (2000) 151–155.

[4] L. E. Hitchner, H. A. Sowizral, Adapting computer graphics curricula to changes in graphics, Computers & Graphics 24 (2) (2000) 283–288.

[5] L. Hitchner, S. Cunningham, S. Grissom, R. Wolfe, Computer graphics: The introductory course grows up, in: Proceedings of the thirtieth SIGCSE technical symposium on Computer science education, 1999, pp. 341–342.

[6] R. Wolfe, A syllabus survey: Examining the state of current practice in introductory computer graphics courses, Computer Graphics 33 (1) (1999) 32–33.

[7] J. Brown, R. Burton, S. Cunningham, M. Ohlson, Varieties of computer graphics courses in computer science, in: Proceedings of the nineteenth SIGCSE technical symposium on Computer science education, 1988, p. 313.

[8] M. M. Larrondo-Petrie, J. Bresenham, C. Laxer, J. Lansdown, G. S. Owen, Approaches to teaching introductory computer graphics, in: Proceedings of the 21st annual conference on Computer graphics and interactive techniques, ACM Press, 1994, pp. 479–480.

[9] S. Grissom, J. Bresenham, B. Kubitz, G. S. Owen, D. Schweitzer, Approaches to teaching computer graphics, in: Proceedings of the twenty-sixth SIGCSE technical symposium on Computer science education, 1995, pp. 382–383.

[10] S. Cunningham, Recommendations of the computer science curriculum group of interest, Eurographics/SIGGRAPH Workshop on Computer Graphics Education 02 (CGE 02) (2002).

[11] J. Foley, A. van Dam, S. Feiner, J. Hughes, Computer Graphics Principles and Practice, 2nd Edition, Addison-Wesley, 1990.

[12] R. Parent, Computer Animation: Algorithms and Techniques, Morgan Kaufmann, 2001.

[13] P. Dutre, P. Bekaert, K. Bala, Advanced Global Illumination, AK Peters, 2003.

[14] E. Paquette, CoGIP: a course on 2D computer graphics and image processing, in: ACM SIGGRAPH 2004 Educators Program, 2004.

[15] S. Cunningham, Report of the 1999 Eurographics/SIGGRAPH workshop on graphics and visualization education, Graphics and Visualization Education Workshop (GVE '99) Proceedings (1999).

[16] A. Watt, F. Policarpo, The Computer Image, Addison-Wesley, 1998.

[17] F. Hill, Computer Graphics Using OpenGL, 2nd Edition, Pearson Education, 2001.

[18] N. Efford, Digital Image Processing: A Practical Introduction Using Java, Addison-Wesley, 2000.

[19] R. Gonzalez, R. Woods, Digital Image Processing, 2nd Edition, Prentice Hall, 2002.

[20] D. Hearn, M. Baker, Computer Graphics with OpenGL, 3rd Edition, Prentice Hall, 2004.

[21] J2DCG, Java 2D computer graphics and imaging framework, http://j2dcg.sourceforge.net/ (2003).

[22] D. Bouvier, From pixels to scene graphs: The evolution of computer graphics courses, in: Proceedings of the Educator Program, ACM SIGGRAPH 2001, 2001.

C Vitae

Eric Paquette is a professor at the École de technologie supérieure engineering school in the Software & IT Engineering department. He is also a member of the LESIA research laboratory at École de technologie supérieure. His research interests include realistic image synthesis, visualization, and Computer Graphics education. Professor Paquette received his B.Sc. in Computer Science from Université de Sherbrooke, Canada, and both his M.Sc. and Ph.D. degrees in Computer Science from the Université de Montréal, Canada. He is a member of the IEEE Computer Society and of ACM SIGGRAPH. He is also on the board of the ACM SIGGRAPH Montreal Chapter.

Figure Captions

Figure 1

Statistical correlation between CG, IP, and HCI.

Figure 2

Introductory courses related to CG, IP, and HCI.

Figure 3

Average number of CG, IP, and HCI courses in different curricula.

Figure 4

Proportions of introductory and advanced CG courses per curricula type.

Figure 5

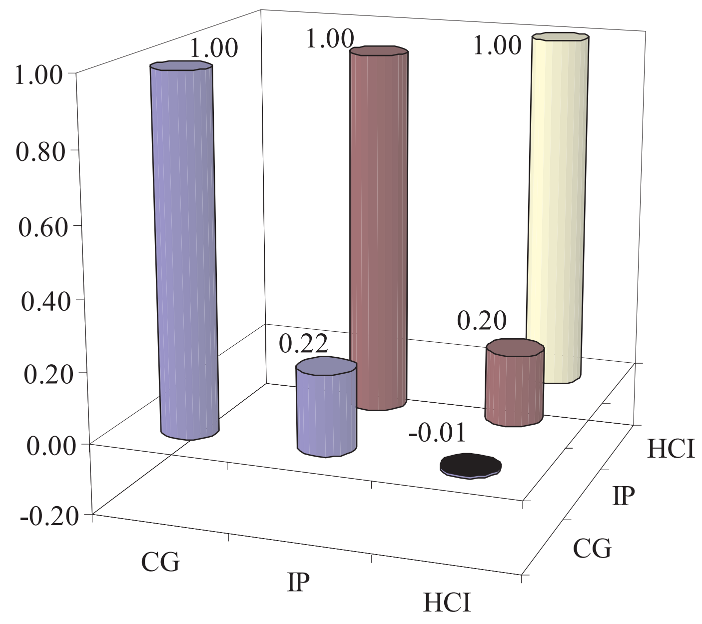
Number of courses per curriculum at typical and CG leadership universities.

Figure 6

Proportions of introductory and advanced CG courses.

Figure 7

The user interface of the j2dcg framework developed for the 2D CG and IP course.


1.00
1.00
1.00
0.22
0.20
-0.01
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
CG
IP
HCI
HCI
IP
CG

[Fig. 1]

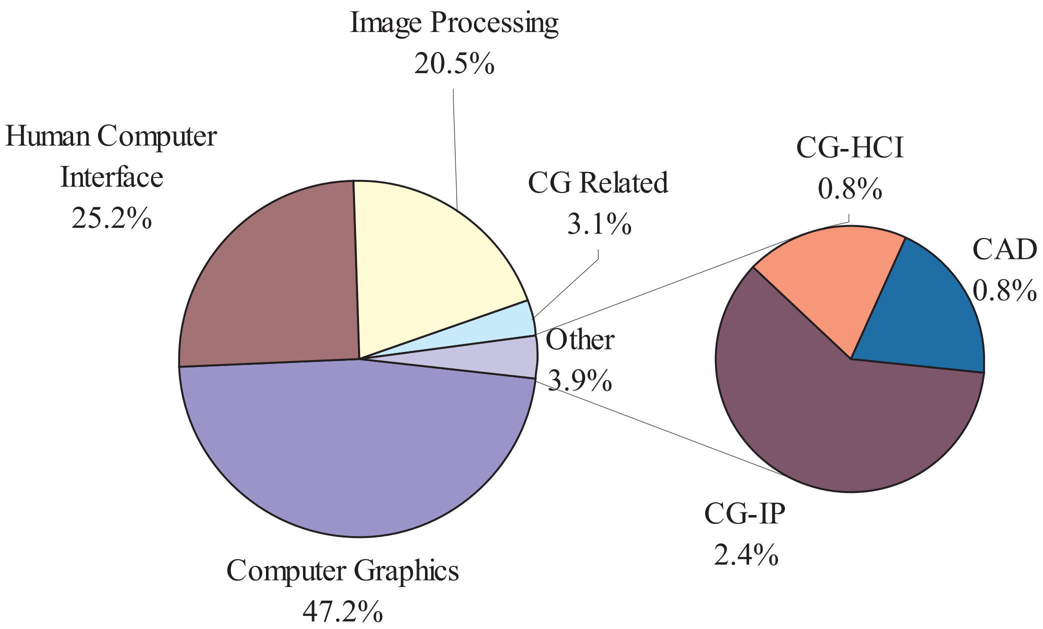
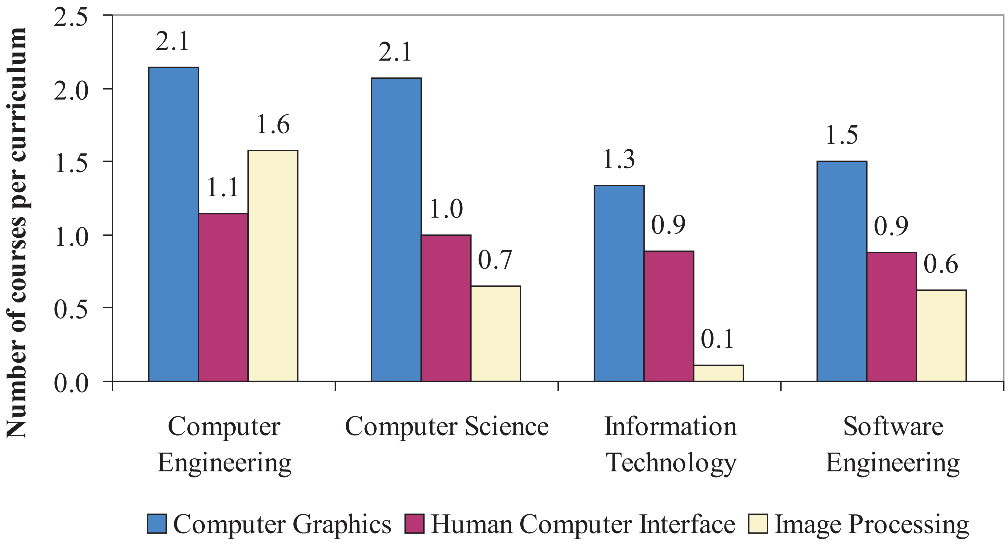


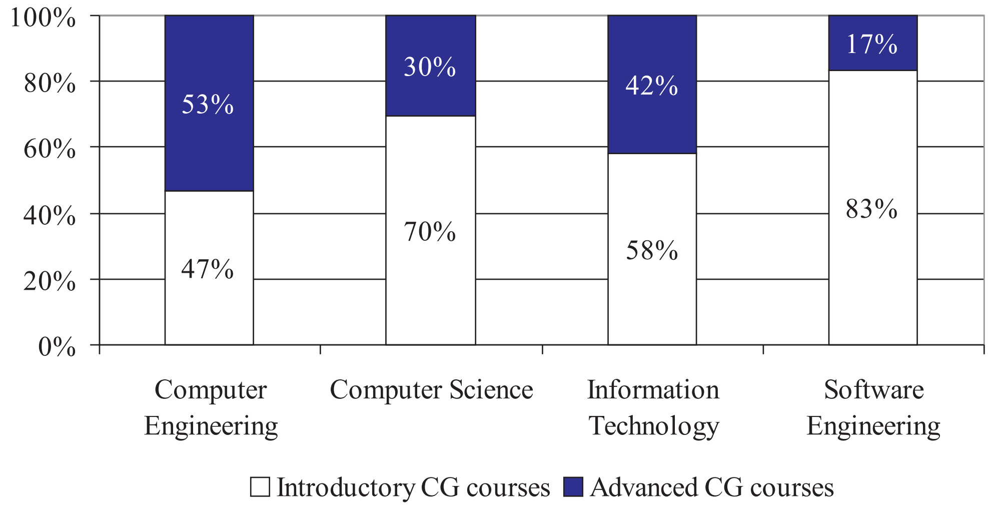
Image Processing
20.5%
Human Computer
Interface
25.2%
CG Related
3.1%
Other
3.9%
Computer Graphics
47.2%
CG-HCI
0.8%
CAD
0.8%
CG-IP
2.4%

[Fig. 2]

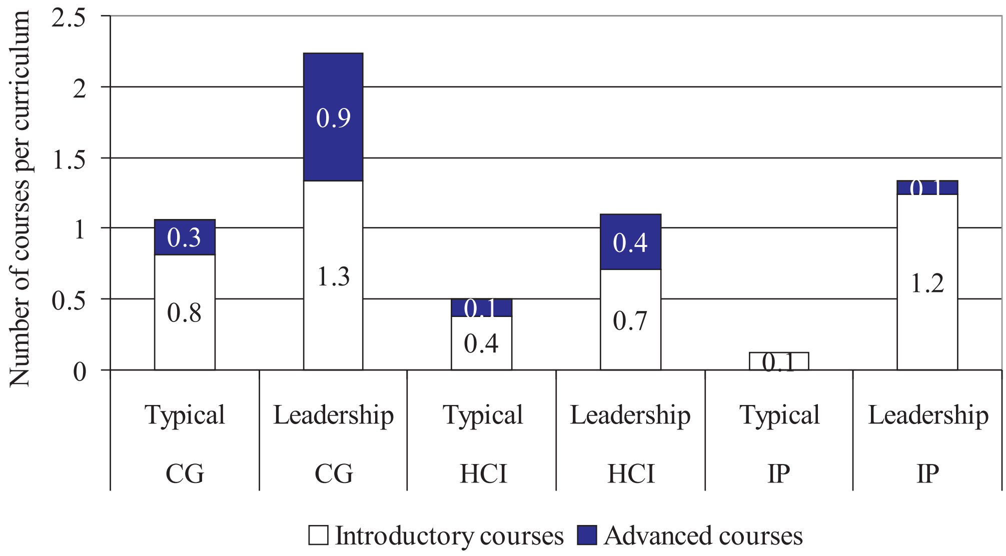


Number of courses per curriculum
2.5
2.0
1.5
1.0
0.5
0.0
2.1
1.1
1.6
2.1
1.0
0.7
1.3
0.9
0.1
1.5
0.9
0.6
Computer Engineering
Computer Science
Information Technology
Software Engineering
Computer Graphics
Human Computer Interface
Image Processing

[Fig. 3]

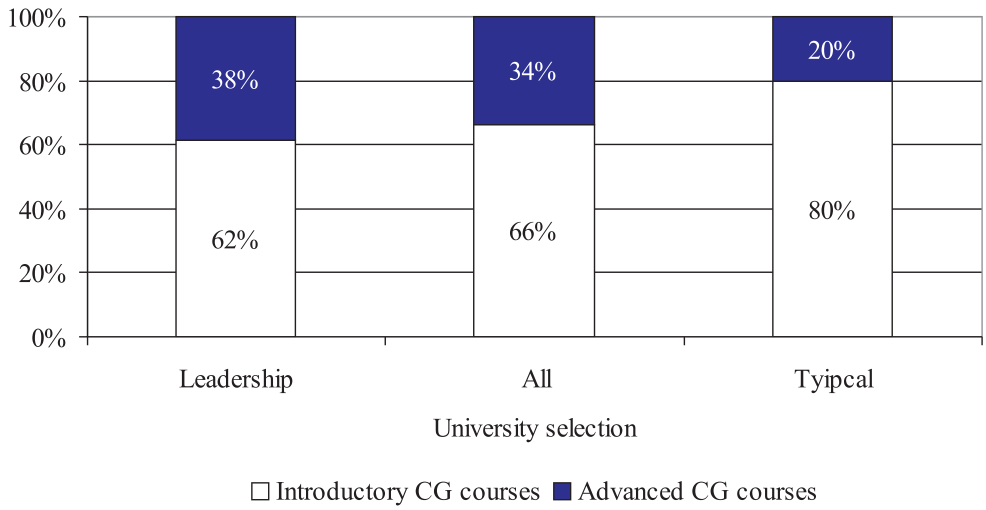

100%
80%
60%
40%
20%
0%
53%
47%
30%
70%
42%
58%
17%
83%
Computer Engineering
Computer Science
Information Technology
Software Engineering
Introductory CG courses
Advanced CG courses

[Fig. 4]

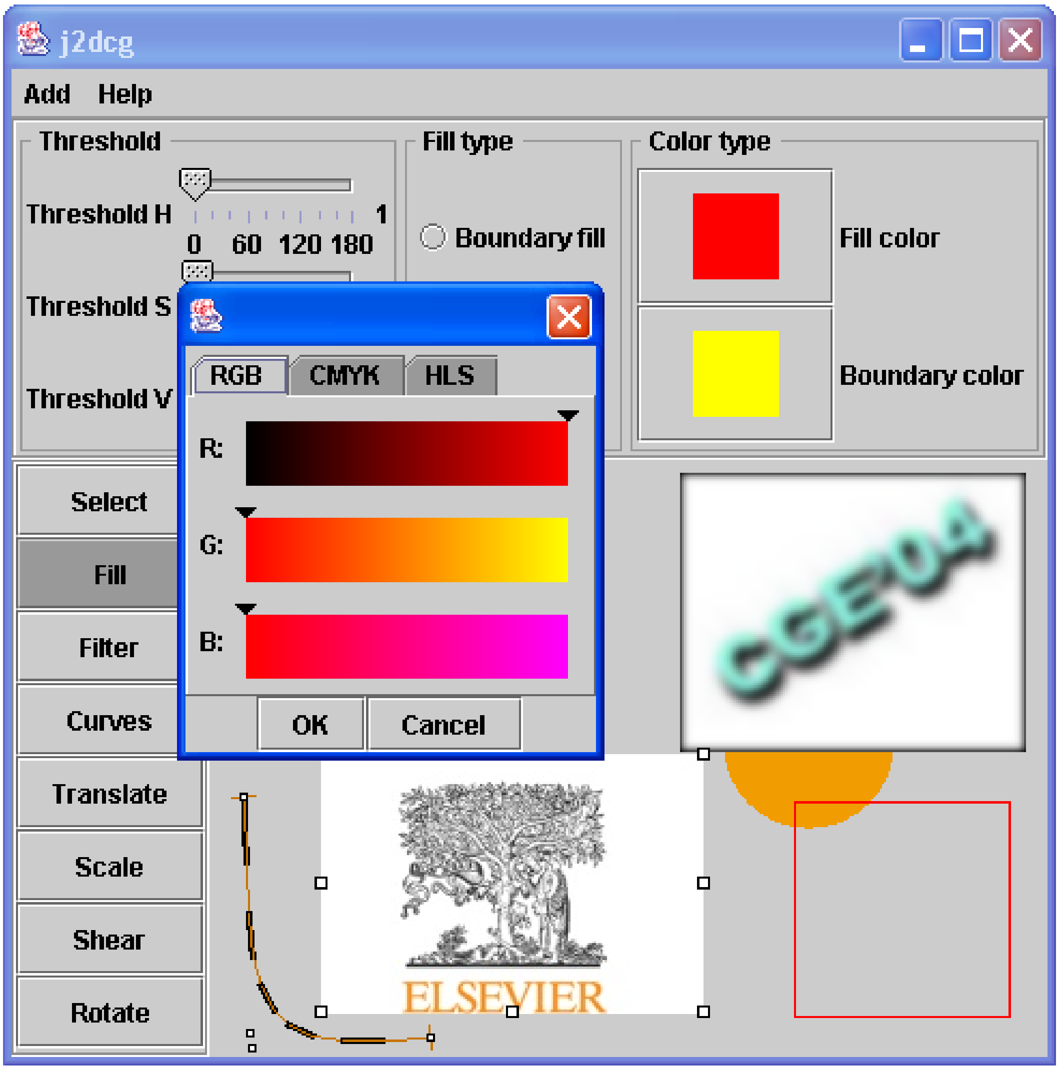


Number of courses per curriculum
2.5
2
1.5
1
0.5
0
0.3
0.8
0.9
1.3
0.1
0.4
0.4
0.7
0.1
0.1
1.2
Typical CG
Leadership CG
Typical HCI
Leadership HCI
Typical IP
Leadership IP
Introductory courses
Advanced courses

[Fig. 5]


100%
80%
60%
40%
20%
0%
38%
34%
20%
62%
66%
80%
Leadership
All
Tyipcal
University selection
Introductory CG courses
Advanced CG courses

[Fig. 6]


j2dcg
Add
Help
Threshold
Threshold H
0
60
120
180
1
Threshold S
Threshold V
Fill type
Boundary fill
Color type
Fill color
Boundary color
RGB
CMYK
HLS
R:
G:
B:
OK
Cancel
Select
Fill
Filter
Curves
Translate
Scale
Shear
Rotate
CGE'04
ELSEVIER

[Fig. 7]

Table Captions

Table 1

Relationships between topics shared by CG and IP.

Table 2

Relationships between CG and IP topics that are advanced topics in one of the two disciplines. The advanced topics are highlighted in italics.

Table 3

Topics of 2D CG and their extension to 3D.

Table 4

Textbooks used to build the 2D CG and IP course notes.

Table 5

Suggested assignments their balance between CG/IP and Vector/Raster.

Computer Graphics	**Shared / Related**	**Image Processing**
synthesis	image	processing
rendering	volume	filtering, thresholding
	perception	
	color	
	sampling	
antialiasing		aliasing
supersampling		filtering
	texture filtering	
texture mapping	forward-reverse mapping	image 2D transformation
animation	morphing	image

[Tab. 1]

Computer Graphics	**Shared / Related**	**Image Processing**
shading		*shape from shading*
image synthesis	projection	*stereo*
HDRI mapping		contrast
multiresolution surfaces	wavelets	image compression
synthesis	texture	*analysis*
	3D reconstruction	

[Tab. 2]

2D Computer Graphics	**Topics**	**3D Computer Graphics**
2D	curves	3D & surfaces
2D	transformations	3D & projection
2D	animation	3D, more degrees of freedom
2D vector	primitives	polygon, sphere, torus, *etc.*
DDA, mid point	vector to raster	scan-line
contour / interior		wireframe / smooth shading

[Tab. 3]

Textbooks	**Topics**
Watt & Policarpo [16, chap. 25]	perception, color
Hill [17, chap. 10]	color, sampling, vector primitives, reproduction, rasterization, filling
Efford [18, chap. 7]	image filtering (spatial domain)
Gonzalez & Woods [19, chap. 4]	image filtering (frequency domain)
Foley et al. [11, chap. 11]	curves
Hearn & Baker [20, chap. 5]	transformations

[Tab. 4]

Topic	**CG/IP**	**Vector/Raster**
Color models	both	both
Filling	CG and some IP	raster
Filtering	IP	raster
Curves	CG	vector
Transformations	CG and some IP	vector and some raster

[Tab. 5]