MOOCs in Computer Graphics

J.-J. Bourdin[1]

[1]LIASD Université Paris 8, France
jj@ai.univ-paris8.fr

Abstract

Since their first presentations, MOOCs have been advertised as the future of higher education as well as a solution against the increase of tuition fees. This panel is introduced by an experiment of MOOCs in Computer Graphics. Our purpose is to give some material to encourage the audience to discuss the opportunity of a common material for teaching Computer Graphics.

Categories and Subject Descriptors (according to ACM CCS): K.3.0 [Computer and Education]: General—K.3.1.3 [Computing Milieux]: Computers and Education—Computer Uses in Education Distance Learning K.3.2.1 [Computing Milieux]: Computers and Informations Science Education—Computers Science Education

1. Introduction

"They came; they conquered very little; and now they face substantially diminished prospects." Robert Zemsky [Zem14].

MOOCs (Massive Open On-line Courses) have been widely presented as the future of higher education [EDU13]. As the supply of tenured faculty is decreasing [Rut14] and as the cost of higher education is increasing, almost anything can be considered as a solution as long as it increases the productivity of the teaching staff. With a few MOOCs enrolling more than 100 000 students, there seemed to be such an opportunity.

The first MOOCs have been presented as early as 2008. After seven years it may be time to have an overview of what MOOCs offer, what they don't offer, the problems encountered and their found solutions if any. Our purpose is to focus on MOOCs for Computer Graphics to help discuss what common material may be elaborated for our community.

Hereafter we present our study on the MOOCs in Computer Graphics and some discussion on the problems encountered.

2. Personal experiments

We started to work on MOOCs during winter 2015 using two different approaches: 1- try and make try by our students MOOCs in Computer Graphics; 2- make an academic work, i.e. read the papers on MOOCs and make a survey of the most relevant results.

The last method led us to a great disappointment. Most of the papers focus on technical aspects (see [Thi14] for example) or on ways to decipher the large amount of data [YMF15]. We found almost nothing about what MOOCs are good at or what they do really achieve. Only a few of these papers put MOOCs in perspective, Zemsky [Zem14], Thille [Thi14] or Ruth [Rut14] have been helpful with our understanding of what goals MOOCs achieve. But the most helpful paper has been Burge's [Bur15], where the author presents her experiment of MOOCs, having tried them herself. This was an encouragement for our main approach. Therefore we will start with the presentation of our experiments of MOOCs in Computer Graphics.

We wanted to try MOOCs in Computer Graphics and were surprised to find only three of them:

- Foundations of Computer Graphics [Ram14] on edX is a course that presents the concepts of 3D graphics with OpenGL and GLSL programming.
- Interactive 3D Graphics [Hai14] on Udacity is based on three.js [thr16].
- Interactive Computer Graphics [Iga14] on Coursera introduces interactive tools with their algorithms.

These courses are presented by a short video of the author, as in class with slideshow and then focus on problems and ways to solve them. Sometimes more videos are presented to explain details.

To study these MOOCs our idea was that teacher and students should try them. During the spring session of 2015 students in the Computer Graphics course (third year students of CS curriculum) were proposed to choose between writing a program or study a MOOC in computer Graphics and write a report on what they had learned to validate the course. The main idea was to encourage students to study a MOOC. As the projects represent a large amount of workload, it was supposed that students would prefer the study of a MOOC. It started that way with 8 students of the 23 choosing a MOOC and starting (or so they said) to work on it. Of these 8 students, only one completed one of the MOOCs. 4 of them stopped

DOI: 10.2312/eged.20161028

and took another assignement, a project, and the last 3 students did not complete the lessons and wrote a short report about the difficulties encountered.

In the following we present briefly one of these courses: Interactive 3D Graphics as it is the only one that was achieved by one of our students.

2.1. The syllabus of Interactive 3D Graphics

This course is based on three.js [thr16] and presents the fundamentals of Computer Graphics.

1. Introduction
   Motivation and a trip down the graphics pipeline, laying out the fundamental processes. Some fundamentals are explained, FPS, CPU cycles, camera and frustum culling, Z-buffer, ...
2. Points, Vectors, and Meshes
   The basics of 3D primitive geometry definition, points, vectors, meshes, left-handed coordinates system...
3. Colors and Materials
   Color representation, material computations, ambient and specular lights, transparency and order of displaying objects.
4. Transforms
   Translation, scale, rotation and how to properly combine all these. This lesson is important to apprehend the matrices used for these transforms.
5. Matrices
   Transform representation by matrices and how to fully control them. The basic transformation matrices are presented, identity, translation, rotation, scaling...The transforms of matrices are given, such as transpose and inverse.
6. Lights
   Directional and point light sources. Different ways to compute the light in a scene are given. Even the usage of shadow mapping with three.js is given.
7. Cameras
   How the camera is defined, how matrices are used to compute the projections: perspective and orthographic.
8. Textures and Reflections
   Color and opacity textures, along with reflection and normal mapping. The lesson starts without filtering then presents the nearest and linear filters.
9. Shader Programming
   An introduction to programming vertex and fragment shaders. A large number of videos are presented to explain the different possibilities.
10. Interaction and Animation
    How to select and make objects move. Until then almost everything was static, movement and interactivity are therefore presented with videos to show particles animation and collisions.

2.2. Workload for an average student

These MOOCs explicitly give a workload in term of hours per week and a number of weeks. For example 6 hours per week during 10 weeks for Interactive 3D Graphics [Hai14]. It is difficult to understand how these values are computed. Our best student, with a good background on programming and computer graphics needed that amount of time. The others students felt overloaded and did not end their courses. It is a usual problem since the workload might be over or under estimated by the author. For example Burge [Bur15] presents a situation where "One MOOC estimated it would involve 3-5 hours of work a week. It turned out that the MOOC had 3-5 hours of video to watch every week, plus readings, plus an essay due that needed to link the video and readings (with citations), plus peer grading multiple essays from classmates."

Another problem occurs: it is difficult to feel compelled to work on a MOOC with sufficient regularity. And the reasons for procrastination are easily overwhelming. Therefore with insufficient regular work the notions maybe be forgotten and the student has to come back to previous lessons before going on. Enlightened by our own experiments we feel authorized to write that the main argument given by some MOOCs "set your own pace" [Iga14] is a false advertising.

3. Problems and solutions

MOOCs have been presented as a solution to the problem of increasing tuition fees [Rut14]. Stephen Ruth writes: "At first glance MOOCs would appear to be the ideal vehicle for reducing tuition, since they can potentially increase the number of students attending the same course by several orders of magnitude." It may be one of the main reasons to the sudden interest on MOOCs starting in 2012. But do MOOCs do the job? The only problem that seems to be solved is the consistency of the programming platform. MOOCs are available and it is easy to work on them whatever the platform used. In this section we focus on different problems raised by using MOOCs.

3.1. Cost

There is, at first the problem of cost. The cost of a MOOC is estimated to 40 000-50 000€. That is not something to ignore. But to be visible, a MOOCs has to be presented on a main provider, such as Coursera, Udacity and edX. In order to work these organizations will need to gain profit from this activity. Therefore they migrate from "free" MOOCs to paid courses. Or the institution will pay an extra cost as noted by Ruth [Rut14]: "For example, if an institution wants to prepare its own MOOC using edX – a well-funded partnership between Harvard and MIT – as an expert consultant, there is a base rate of $250,000 per course with an additional $50,000 for each time the course is offered again."

If these sums are paid for ten thousand of students it seems to be really cheap.

3.2. So many students!

How to assist so many students? How to evaluate their work? How many of them will succeed?
The instructor availability, for one, is a real problem. For example Warren [WRGW14] presents a MOOC where 19 000 students received a Statement of Accomplishment on two sessions (2012 and 2013). The paper is not clear on the question of how many students enrolled in the course. In the 2012 experiment, three staff members were devoted to answer students' help requests and both of

them worked approximately 15 hours per week for each of the nine week course.

The most interesting part of the paper is that students have been advised to peer reviews and peer help. This solution can not be presented as perennial, it may work while some students devotes themselves to help the other students but one can not know what has been learned in this case. For example we have all seen students giving the solution to a specific problem to their comrades while the helped ones still could not devise a solution to another very similar problem. Worse than that, in this case, the student who was helped may feel unable to succeed. In this case the teacher is the only way to ensure the understanding of the lesson. As noted by Burge [Bur15] randomly assigned peers may lead to very random evaluations.

This combines with the simplicity of the evaluation to explain the great number of success in this experiment. The assessment came in two parts: a mini-project and quizzes. The authors state that: "Our social approach to these quizzes was to allow multiple attempts at the quiz to encourage mastery of the material as well as to give students the opportunity to discuss the quiz problems in the class forums." Although this has not been designed as a way to increase the good results they may come as a logical consequence. If a student has been helped by peers on the mini-project and can attempt the quiz until he succeeds, it is surprising that some of them fail.

3.3. Low ratio of success

The low average of success on MOOCs is one of the main concern, it has been evaluated to less than 7% (Ruth states less than 10% [Rut14] while Burges quotes 6.8% [Bur15]). The reasons of these low rates are multiple and extensively discussed, for example by Anderson et al. [AHKL14]. One of them is that most of the students enrolled in a MOOC are not full time students. A lot of them already have a knowledge and/or a practice of the content of the course. Their goal can not be what is expected of our regular students. Some other students enroll in a MOOC while attending a class on a close subject in their institution. In this case the MOOC is seen as an insight for a regular course. These students do not need to finish the MOOC. There are other explanations, but why does no one point at the obvious? It is known and has been known for ages, since Socrates, that one motivation for studying comes with the desire to please the teacher. When the teacher is not there, this motivation fails.

There is also the problem of the credits given. In most cases students do not get credits for their work on MOOCs even if they get a Statement of Accomplishment. To encourage students, badges are given (see for example [PEM14]) at every stage.

3.4. Specific problems

There is, for us, another issue: our field evolves very quickly therefore a MOOC in Computer Graphics as well as your own classes, has to be updated every other year. That means an increase of costs on one hand but also a large amount of work to build a new MOOC every other year. It is not sure that an institution will accept to spend that much money and workload.

In their paper, Piccioni et al. [PEM14] present a very different experiment. They used a MOOC platform to organize a SPOC (Small Private Online Course). With a total of 327 students, this course can not be presented as a MOOC, and it was not the purpose. Their idea was to use the platform as support and complement to a residential course. In this case the results are interesting with students very positive on its use: "The MOOC was helpful to understand the theory." or "The MOOC, which was introduced as a secondary learning instance, complements the lecture perfectly and should absolutely be continued and advanced in the next year." But one of the reasons why this course with SPOC material was popular is the introduction of games as a way to teach programming but as we already noticed at EUROGRAPHICS [AVOW*12] games are a vector of choice to encourage students to program.

4. Conclusion

While our study is not conclusive about MOOCs, it is nonetheless convincing about reasons not to implement a regular MOOC for Computer Graphics.

If the wishes are to get a large number of students, one needs to build a MOOC with somebody well known by the public, not only by specialists of the field. For example, in Computer Graphics, enrollments could be high for a MOOC built by Edwin Catmull but what would be its purpose?

If the goal is to build a course that enables all our institutions and staff a clean slate to organize a course in Computer Graphics then the main problem will be to choose the language (C, C++, Java, Javascript...), the API (OpenGL/GLSL, DirectX, OpenGL ES...) because we do not work similarly.

But having a shared material for our courses in Computer Graphics is a goal to achieve and the experiment by Piccioni et al. [PEM14] should lead us to a proper way to build a MOOC as a complement to our own courses. We do hope a team of Computer Graphics teachers from EUROGRAPHICS will start such a useful project.

Acknowledgments

This work owes much to the efforts of the students of the computer graphics course, we thank them and extend these thanks to the organizers of EG Education program, B. Sousa Santos and J.M. Dischler and the reviewers of this paper and Eric Haines and Ed Angel for their encouraging comments.

References

[AHKL14] Anderson A., Huttenlocher D., Kleinberg J., Leskovec J.: Engaging with massive online courses. In *Proceedings of the 23rd International Conference on World Wide Web* (New York, NY, USA, 2014), WWW '14, ACM, pp. 687–698. URL: http://doi.acm.org/10.1145/2566486.2568042, doi:10.1145/2566486.2568042. 3

[AVOW*12] Adamo-Villani N., Oania M., Whittinghill D., Brown J., Cooper S.: Building a Serious Game to Teach Secure Coding in Introductory Programming Courses. In *Eurographics 2012 - Education Papers* (2012), Gallo G., Santos B. S., (Eds.), The Eurographics Association. doi:10.2312/conf/EG2012/education/033-040. 3

[Bur15] BURGE J.: Insights into teaching and learning: Reflections on MOOC experiences. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education* (New York, NY, USA, 2015), SIGCSE '15, ACM, pp. 600–603. URL: http://doi.acm.org/10.1145/2676723.2677243, doi:10.1145/2676723.2677243. 1, 2, 3

[EDU13] EDUCAUSE LEARNING INITIATIVE:. *ELI Online Spring Focus Session* (2013), Stephen Downes. URL: http://www.educause.edu/events-search/275109. 1

[Hai14] HAINES E.: Interactive 3d graphics, creating virtual worlds, 2014. URL: https://www.udacity.com/courses/cs291. 1, 2

[Iga14] IGARASHI T.: Interactive computer graphics. Coursera, 2014. https://www.coursera.org/learn/interactive-computer-graphics. 1, 2

[PEM14] PICCIONI M., ESTLER C., MEYER B.: Spoc-supported introduction to programming. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education* (New York, NY, USA, 2014), ITiCSE '14, ACM, pp. 3–8. URL: http://doi.acm.org/10.1145/2591708.2591759, doi:10.1145/2591708.2591759. 3

[Ram14] RAMAMOORTHI R.: Foundations of computer graphics. edX, 2014. https://courses.edx.org/courses/BerkeleyX/CS_184.1x/3T2014/. 1

[Rut14] RUTH S.: Can MOOCs help reduce college tuition?: MOOCs and technology to advance learning and learning research (ubiquity symposium). *Ubiquity 2014*, July (July 2014), 3:1–3:10. URL: http://doi.acm.org/10.1145/2591685, doi:10.1145/2591685. 1, 2, 3

[Thi14] THILLE C.: Opening statement: MOOCs and technology to advance learning and learning research (ubiquity symposium). *Ubiquity 2014*, April (Apr. 2014), 1:1–1:7. URL: http://doi.acm.org/10.1145/2601337, doi:10.1145/2601337. 1

[thr16] THREEJS.ORG: three.js, 01 2016. URL: http://threejs.org. 1, 2

[WRGW14] WARREN J., RIXNER S., GREINER J., WONG S.: Facilitating human interaction in an online programming course. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (New York, NY, USA, 2014), SIGCSE '14, ACM, pp. 665–670. URL: http://doi.acm.org/10.1145/2538862.2538893, doi:10.1145/2538862.2538893. 2

[YMF15] YIN H., MOGHADAM J., FOX A.: Clustering student programming assignments to multiply instructor leverage. In *Proceedings of the Second (2015) ACM Conference on Learning @ Scale* (New York, NY, USA, 2015), L@S '15, ACM, pp. 367–372. URL: http://doi.acm.org/10.1145/2724660.2728695, doi:10.1145/2724660.2728695. 1

[Zem14] ZEMSKY R.: With a MOOC MOOC here and a MOOC MOOC there, here a MOOC, there a MOOC, everywhere a MOOC MOOC. *Journal of General Education 63*, 4 (2014), 237–247. 1