

CASE STUDY

F16 TROPHY

About the author

I've always been very interested in science but also fiction. I've always thought that these two are linked, because understanding and visualizing our world requires a powerful sense of imagination. That's why I am passionate about imagining the impossible, and movies are the best candidates to stimulate that. Since 2009 I've tried to learn the basics of visual effects & 3D by myself, using tutorials in the web. I started with the 2D compositing of stock footage and small title designs thanks to the tutorials of the very talented Andrew Kramer. I also tried to use Blender for my VFX, allowing me to put 3D models impossible to capture in real life into my shots. At this stage this was just for fun, I made a couple of videos and started writing little stories in order to apply what I've learned in the tutorials during my spare time. It's in 2016 that I created my YouTube channel named "BEAM Studios" and I decided to be more and more serious about technical aspects of VFX and filmmaking. This was all about trying to reverse engineer epic sequences of big movies like Marvel Studios's Dr Strange or Star Wars. The real challenge at this point was to keep improving VFX quality (by increasing the working hours) and follow the aerospace engineering training at ISAE-ENSMA, a French university.

Context

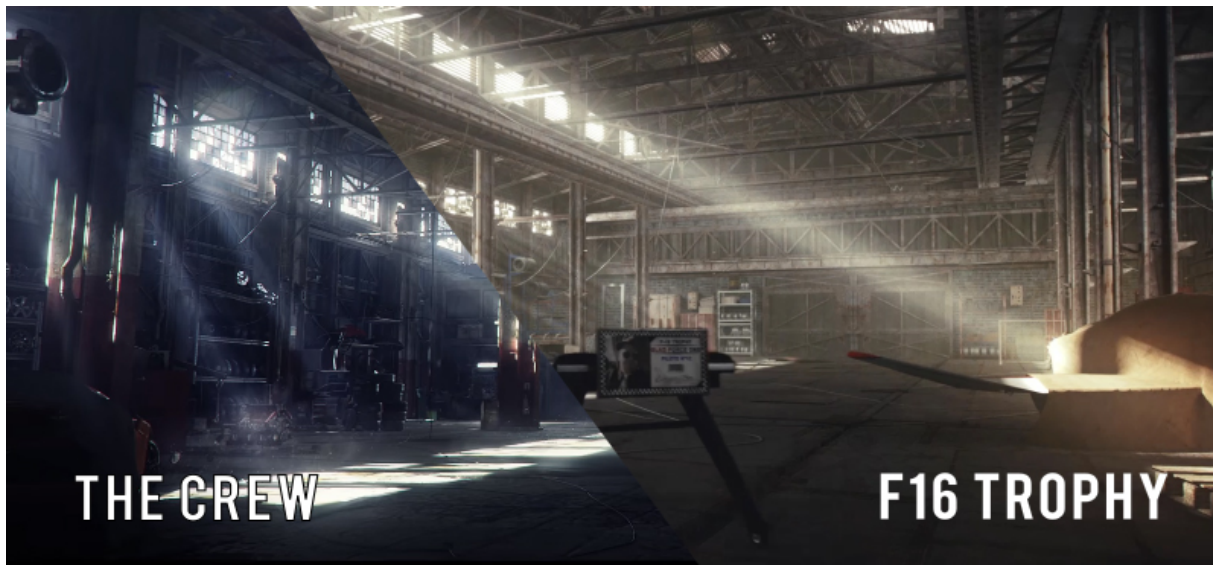
The graduation ceremony at my university is composed of a theatrical show. Short videos are projected during the performance to increase the immersion. I was responsible of the creation of the opening, that is usually a 1-minute 3D animation video. Every year a team of students (8 to 12) challenges themselves to improve the result, using Blender internal renderer. For 2018's ceremony I decided to go where nobody of my university has gone before, use a physically based renderer for an 8-minutes 3D animation short film.

Before starting the project

Ideas & inspirations

The first step was to find the theme of the video, in this case all the story was focused on mechanics & aeronautics. So there had to be aircraft and engineering stuff like spare parts, assemblies etc. The other challenge was to blend the latter elements with a story based on the year group (The one that is graduated). I won't develop this story as it is composed of private jokes that aren't relevant in this case study.

The main inspirational video that I found in the internet was a video game trailer “The CREW” developed by Ivory Tower and published by Ubisoft.



The full CG trailer was made by a French talented team at Unit Image. I instantly fell in love with the photorealistic look of the images and the animation perfectly merged with the music. As I was an intermediate user with Blender, I decided to learn 3D by adapting this trailer for my project.

Other movies helped me for my animations like Iron man and it's F22 raptor scene and Pearl harbor dogfights scenes. I also watched random fighter jet videos just to see how an aircraft move. The idea was to find exactly how to animate my 3D models so it's possible to visually feel the aerodynamic forces and instabilities.

First tests with Cycles

Before starting the project, I wanted to study the feasibility of making a photorealistic animation with Cycles. After 2 weeks of deep research on the internet I found Andrew Price (Founder of Blender Guru & Poliigon). His tutorials perfectly explained how to get a realistic look with architecture renders. The ideas I retained for this project are the following:

- Use PBR maps for 90% of the materials
- Enable filmic color management for the best dynamic range
- Pay attention to the scale of the objects relative to the scene
- Add realistic & interesting lighting
- Add random & realistic details to the ground (ground detail is a priority)
- Complexify the geometry (i.e add bevels, clean meshes, correct UV) during the modelling.

By applying these 6 steps and a lot of working hours I was able to get plausible results. The next challenge was to find a way to make an animation.

Modelling & texturing the warehouse:



Render time

Render time in Cycles basically mean getting rid of the noise. The latter is generally caused by convergence issues with light paths calculations and probabilistic global illumination. Again, Andrew Price made a very good video explaining how to render an image with less samples. What I retained: In Cycles a lot of parameters need to be adjusted to optimize the renders for animations. It can be indirect light bounces, volume step size, multiple importance surfaces or caustics etc.

But the substantial feature that completely revolutionize 3D animations in Blender is the denoiser. The denoiser allows the users to render a clean image with incredibly less samples hence less render time.

In my project the real challenge was interior scenes as the lighting is mainly composed of global illumination. For these environments I needed to clamp indirect light, keeping a realistic lighting but also less noise in the corners or walls.

With all these techniques I could render a single image in 25 minutes with an intel core i7. The result was very satisfying but even with this render time, a 2 min animation film with interior shots was rendered in 52 days nonstop. This render time is very uncomfortable, and the main issue is that it's impossible to work simultaneously.s

Render time solutions

Ray tracing renderers are the best tools when it comes to photorealism but require a lot of CPU power. After some research on the internet I found some solutions:

- Buy multiple CPU machines. This is maybe a good solution in the long run. But I had only 1 project to complete, this wasn't necessary.
- Buy one more graphic card. GPU rendering is a good way to drastically reduce render time, in this case from 25 min to 7 minutes per frame of interior shots. I had already one Nvidia GTX1070 Aero OC. But the Aero OC version causes Cycles to randomly crashes with the following error "Misaligned address". It was risky to buy another one as the developers still don't know where the problem come from.
- Use Sheep it, a free rendering service that allows users to mutually render frames in a worldwide scale. This is a very good solution for small projects and simple scenes. Unfortunately, I needed more features like the possibility to upload heavy smoke cache, download non-destructive open exr images, and upload big project files as all the PBR materials are packed. Sheep-it can't handle these projects for the moment.

Retained solution

The last solution was to pay an online rendering service that handles all the features that are necessary to get a professional workflow with photorealistic animation and also the possibility to make 3D visual effects. The first service that proposes all that for Blender is RenderStreet. I planned 6 months of rendering and production in the same time, so I didn't need my shots to be rendered quickly. I just wanted to render my shots when they were ready and work on the next one simultaneously, then repeat the process. Hence the monthly payment (more restrictive than the "pay per hour" solution) is adapted to this workflow because I don't need the full CPU/GPU power capacity that is more expensive. This solution is then compatible with projects that are planned way before the deadline. In this case 9 months for 8 minutes.

The monthly solution is available only for renders that are below 15 mins per frame in a 32-thread dual Xeon machine. This render time is easily reached with samples around 400 and denoiser on. The price of rendering was fixed to 300\$ instead of more than 1500\$ with the on-demand solution for 196 3D shots. Then it's possible to export full HD open exr image sequences, upload all possible caches (smoke, simulations, particles). I could benefit from the latest versions of Blender and I had access to a mapped ftp server to upload my Blender files using Windows explorer. So, I decided to choose RenderStreet as a rendering solution for this ambitious project. And it worked.

Production

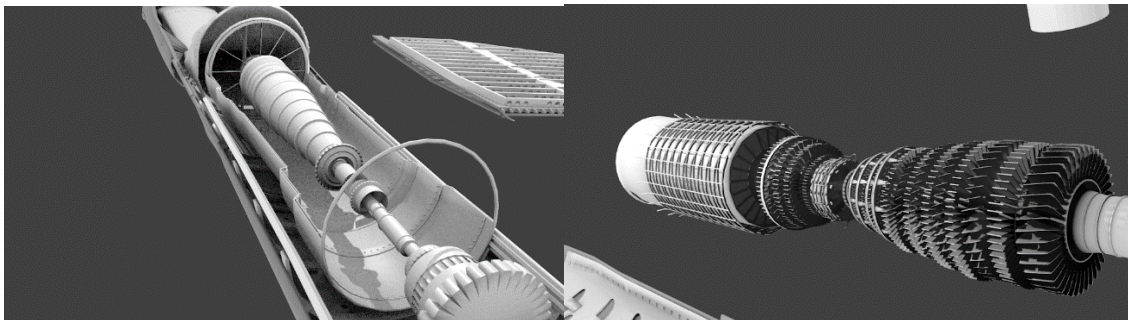
Script

A 3D animation this long needs to be planned from the beginning to the end. So, the first step was to write a global script with the different ideas then a detailed script that list all the 3D shots. Each shot written was imagined, without any precise timing as it's impossible to visualize that by head. This step is crucial in the sense that it determines what is going to happen in the video and what are the environments & objects that will be used for sure.

First pass of 3D modelling – Main objects & environments

After I completed the detailed script, I started to model the environments. In this video the environment is generally a room, an exterior scene near a building or a landscape. The idea is to create the environments that I imagined in the detailed script and try to model them according to camera angles I had in mind. It's very important to know when to stop modeling medium details of a room as the camera angles aren't set yet.

For instance, I was completely sure about the use of the warehouse. I modelled it entirely and I started to add pipes, vents, big objects and furniture just to start setting up the scale of the scene. Then I modelled and textured objects that are the center of attention. All mech parts of the F16, highly detailed F16 wheels, the turbofan engine, F16's rig, various aircraft, a robot, a car and it's rig etc.



The reason I modelled these objects and partial environments is that I needed them to set up the animatic. As I was alone I didn't want to create low poly models and environments for the animatic then start from the beginning with highly detailed objects.

Animatic

So, I made an animatic of the video, this is a viewport visualization of animations, timing and continuity of the shots. It helps to verify the synchronization of the shots with the music. Above all, the animatic allows to know exactly the total number of shots and the length of the video, improving drastically the planning's quality.

I took the objects I modelled, and I put them into my scenes. I started to add cameras in different angles according to what I had in mind in the detailed script. After setting-up some keyframes I was ready to export the shots and verify continuity.

The OpenGL renderer in Blender allows to export very quickly the viewport in AVI JPEG format. Every shot is then placed in a video editing software like Adobe Premiere. The idea is to move between the two software and add corrections or improvements until the result is satisfying. Thanks to that I was able to confront myself to issues that I couldn't anticipate by head. This could be animation issues, feasibility of the shots or global feeling of camera movements. The advantage of this workflow is that when the animation looks good, I only need to focus on adding details, texturing and rendering. And I'm sure that the result will work without losing my time rendering non-compatible shots. This process took 4 months.

Last pass of 3D modelling

After the animatic was complete I went back to the beginning of the video and I added details in the field of view of the camera. These objects aren't interesting individually but they help to set-up the scale of the scene and the photorealistic look of the image. This could be papers, rocks, a cell phone, imperfections and so on. One should pay attention of the polycount as well as the memory. Smart duplication of small objects is a very good way to add details, but it can crash the computer if it's not properly done (Alt+D conserves memory except Shift+D).

Lighting

I activated the filmic color management for the highest dynamic range, in that way I could turn up light energy coming from the outside without burning the image.

Every scene needs a realistic lighting. One of the biggest challenges was the warehouse.

One way to light the scene was to use a sun lamp (parallel lights) entering by an array of small windows. Photons bounce over the ground and hit every corner of the room. The issue with indirect lighting, is that Cycles doesn't like big interior spaces with small entries for the light, it generates an incredible amount of noise.

Example without clamping indirect lighting (denoised):



Clamping indirect lighting (i.e reducing energy that comes from indirect light) tend to preserve the final render, however it's less physically accurate. To compensate that I needed to add area lights that approximatively recreate global illumination.

Export

As I said I had access to a mapped ftp drive in my windows explorer. I just had to click and drag my blender files. I found that this way of uploading files is more stable than directly via the web browser. The latter showed some errors during upload as the internet connection was quite poor. But even with this drawback I was able to upload and download my 196 3D shots not to mention all the VFX (smoke, clouds). Smoke cache can be easily uploaded as long as the relative paths are conserved between files.

Final corrections

The last step is to use a compositing software (After effects or Nuke) to add final color corrections and bring life to the final render. This is here where the multiple passes are combined (volumetrics, Zpass, smoke elements, explosions). Most of the VFX are made outside of Blender, as the latter is a 3D software.

Conclusion

It seems that the denoiser feature completely revolutionize the animations in Blender, allowing the artists to render frames below 1000 samples without noise. In addition to that, thanks to RenderStreet it's possible to handle ambitious projects with a complete set of tools & features that preserves the rendering workflow. It's regularly updated to the latest version of Blender, and it works 99% of the time.

However, there are some Blender tools that need to be improved in the future. The first thing would be the camera's rig. It would be very useful to propose directly a two-node camera with a point of interest without being constantly trying to play with null objects and non-intuitive constraints. The second point would be to improve the smoke simulator, even if it is really good, some situations forces the users to use more complex simulations software such as FumeFX (which is very expensive as it comes with 3DsMax, but this is the price of physically based fluid dynamics algorithms). The denoiser should be slightly improved to avoid artifacts around bright and sharp areas.

The last versions of Blender allow a lot of people to have access to CGI for free and the results promise a good future for this magic software.