DEV1

A Collaborative Virtual Production

Rochester, NY
Summer, 2021

Presented By
MAGIC Spell Studios at the Rochester Institute of Technology
Optic Sky Productions

Prepared By
Andrea Avendano Martinez
David Long
Mark Reisch
Emily Haldeman
Tim Stringer

Abstract

In the spring of 2021, MAGIC Spell Studios at the Rochester Institute of Technology partnered with Optic Sky Productions to produce a short teaser for RIT's Electric Vehicle Team (EVT). The project employed multiple virtual production technologies, such as LED walls, camera tracking, and real-time graphics rendering, with the goal of better understanding their use for both professional and academic purposes. It will also serve as a basis for the growing virtual production curriculum at RIT.

This report documents the production from start to finish and analyzes learnings and best practices around workflows associated with the various technologies utilized.

I. INTRODUCTION

Virtual Production

Virtual production is the integration of traditional live action filmmaking with real time graphics rendering. This umbrella term can apply to a variety of workflows with a wide range of applications taking place at any time within the production pipeline. Some major virtual production workflows include visualization through a game engine, motion capture of actors for animation, and on-set virtual production such as in-camera VFX.

Visualization

Visualization in virtual production encompasses a range of techniques that allow filmmakers to scout virtual locations, set up shots, and capture performances in camera all through traditional interactive display rendering or immersive virtual reality, with rendering usually driven by a game engine. Visualization includes pitchviz, previz, techviz, stuntviz, and postvis. At the early stages, pitchviz helps demonstrate the look and feel of the project before any major investment. Then, during pre production, previz provides an interactive experience where each shot can be planned from the camera angles, movement, lighting, and sound effects to approximate the look of the final sequences. For example, previsualization of sets in a virtual environment could be used to determine actor blocking before shooting. During production, techviz and stuntviz can be utilized to refine the precise technical and physical requirements for each shot as well as the feasibility and safety of stunt work. Lastly, postviz realizes the filmmakers' vision to the VFX team by using further visualization to represent any incomplete shots. For example, merging live action elements with temporary visual effects.

Motion Capture for 3D Animation

Another area of filmmaking that lends itself well to virtual production techniques is 3D animation. Performance animation can be facilitated through the use of real-time motion capture techniques that track real actor's movements to animate virtual characters. This could refer to either body movements or even more detailed facial tracking to generate emotional expressions. In terms of virtual production, the data can be collected using volumetric capture techniques, for example, which make use of an array of cameras around a target to digitize 3-dimensional space. Taking it a step further, these techniques could be used to create and animate photorealistic digital humans, or metahumans. Epic Game's Digital Humans project explores this as a building block for the future of gaming and even academic applications aside from filmmaking.

In-Camera VFX

Perhaps one of the most popular areas of virtual production, on-set VFX workflows make it possible for camera operators and crew to see and react to real-time, perspective-matching imagery. Two major examples of this are Simulcam or live compositing and LED wall cinematography, the latter of which was the main workflow used in our production and documented in this paper.

Simulcam compositing refers to the immediate juxtaposition of real and CG elements on screen achieved by using camera tracking data. This means the CGI scenes in a film can be directed as a traditional live action scene on set rather than in the post production phase, with live composite views available to the director, cinematographer, and VFX supervisor.

The concept of in-camera VFX in its essence has been available for decades, considering rear and front projection technologies used in the film era. As early as the 1930s, in-camera visual effects were achieved by combining foreground performances with pre-filmed backgrounds. So while the idea of projecting imagery behind actors to capture in-camera effects is not new, virtual production takes it a step further in that it offers fluidity and real-time synchronized parallax between the screen and the camera, powered by game engines and faster GPUs. Additionally, by tracking the movement of the camera, the real time sequences behind the actor are also able to be reframed and driven by real cinematic decisions. Each frame will regenerate based on the perspective of the camera so that everything looks real. This is an important advantage of virtual production over traditional VFX. The flexibility of adjusting the movement of a shot gives the director greater creative control and is closely aligned with what a traditional set would feel like. In recent years, game engines like Unreal or Unity have been used more frequently in filmmaking due to their powerful ability to generate realistic virtual environments with virtual cameras and lights as assets. In this way, game engines can be leveraged to create completely virtual film sets which could be directly used in a virtual production.

For this specific virtual production, we used Unreal Engine to drive our virtual set for an in-camera VFX workflow using a hybrid LED volume and rear projection screen. The software was at the core of the workflow as it translated the tracking data from the camera's position and focus on the real set to the virtual environment. It also generated all the assets used to create the virtual sets displayed during production. This meant that we could control the environment as we saw fit for each individual shot. For example, we could control the position of the sun around our talent to produce the most pleasing image; a level of creative power and flexibility that could only be achieved through this type of production. The benefits and challenges we faced throughout our virtual production will be explored in further detail throughout this document.

RIT MAGIC Spell Studios

MAGIC Spell Studios at the Rochester Institute of Technology is a state-of-the-art facility with a multi-disciplinary entrepreneurial approach to digital media learning, research and production. A community of faculty and student game designers, programmers, filmmakers and animators collaborate with industry partners to build entertainment and interactive experiences. Supported by a MegaGrant from Epic Games, faculty at RIT designed a virtual production curriculum in a way that granted students hands-on experience as its main goal.

This gave rise to RIT's first virtual production class offered in the spring of 2021 and made up of students of all disciplines from the College of Art and Design, including Film and Animation, 3D Digital Design, Game Design, and Motion Picture Science (MPS). Motion Picture Science is unique to RIT as it is the only program of its kind in the country, combining a film education with a rigorous engineering training. These students are fully equipped to understand all technical aspects of the industry and are at the forefront of developing technologies in cinema and entertainment.

By integrating engineering-driven MPS students with creative students from the School of Film and Animation, and design savvy 3D Digital Design and Game Design students, an opportunity was created for multidisciplinary collaboration and learning. Live-action film students had insightful knowledge regarding cinematography and lighting as well as practical skills associated with crafting a narrative and operating a film set. 3D Digital Design and Game Design students were more familiar with navigating Unreal and contributing assets in the digital space. MPS students brought technical and engineering skills around topics like display calibration, optics and camera physics. Including such a diverse group of learners with different training and skills and collaborating on a common goal meant mimicking a professional environment where all perspectives were welcomed and discussed. In this way, students could learn from each other in areas where they were less familiar while simultaneously strengthening their own expertise.

Optic Sky Productions

Optic Sky is a Rochester-based advertising and multimedia production company focused on short-form marketing production. Some of their many clients include Wegmans, Kodak, Maserati, and more. The company is also familiar with MAGIC's facilities as it was founded by RIT alumni, Aaron Gordon, and regularly partners with RIT and students. Naturally, Optic Sky was a perfect choice for a ground-level team available to explore virtual production technology with RIT. Given this, it made sense to seek out an on-campus story to highlight and promote as a commercial production, and we found that partner in RIT's successful student-run electric vehicle racing team, EVT. The EVT team is in the process of designing a new off-road electric bike, DEV1, for future competitions and suggested the commission of the promotional piece produced as RIT's first virtual production.

This opened the way for further collaboration with industry-leading entertainment companies such as The Third Floor (TTF), an award-winning visualization studio co-founded by RIT alumni Chris Edwards, and PRG, a global leader in entertainment and event technology solutions. TTF experts delivered guest lectures and directly trained faculty on the technologies while PRG was the project's hardware and equipment partner. For TTF and PRG, identifying new talent on campus to join their professional ranks was also a key priority of the collaboration.

II. BACKGROUND

Virtual Production Technology

Regarding this specific virtual production, we used an LED wall and rear projection screen for real-time rendering of a virtual set for in-camera VFX. We made use of a camera tracking system that would make it possible to take spatial data from the real camera and render the virtual camera's frustum in real time to achieve a photorealistic effect when viewing. The major components used in our production with the support of PRG are as follows:

- LED Walls
    - ROE Black Onyx 2 panels
    - Galaxia WinVision 9 panels
- LED Processors
    - (2) Brompton Tessera M2 LED Processors
    - (2) WinVisionAir 9 Galaxia LED Controllers
    - Barco Folsom ImagePro HD
- OptiTrack System
    - (12) S250e OptiTrack IR cameras
    - Calibration wand (CW-500)
    - Calibration square (CS-200)
    - Motive software
- Projector
    - Christie Boxer 4K20
    - Draper StageScreen (Silver) Rear Projection Screen
- Camera and Lenses
    - Arri Alexa Mini
    - Atlas Orion Anamorphic Lenses (80,50,32mm)
- Engine/Render Nodes, Workstations and Servers
    - (3) BOXX PCs with A6000 GPUs
    - (2) Disguise Media Servers
    - See Appendix for routing diagram
- Unreal Engine 4.26
    - (4) custom virtual environments
    - nDisplay project template

LED Walls

Our layout resembled a classic car process configuration on a soundstage. We made use of two different kinds of LED panels provided to us by PRG, background and overhead. Our main LED wall was made up of two 4 x 8 grid screens using the ROE Black Onyx 2 LED panels with a 2.85mm pixel pitch. This model was chosen for its light-weight, easy to mount design as well as cost-effectiveness. Each panel is 50cm x 50cm and weighs less than 12 lbs. The panels were erected on rolling platforms to permit ultimate flexibility around course instruction and the pilot production.

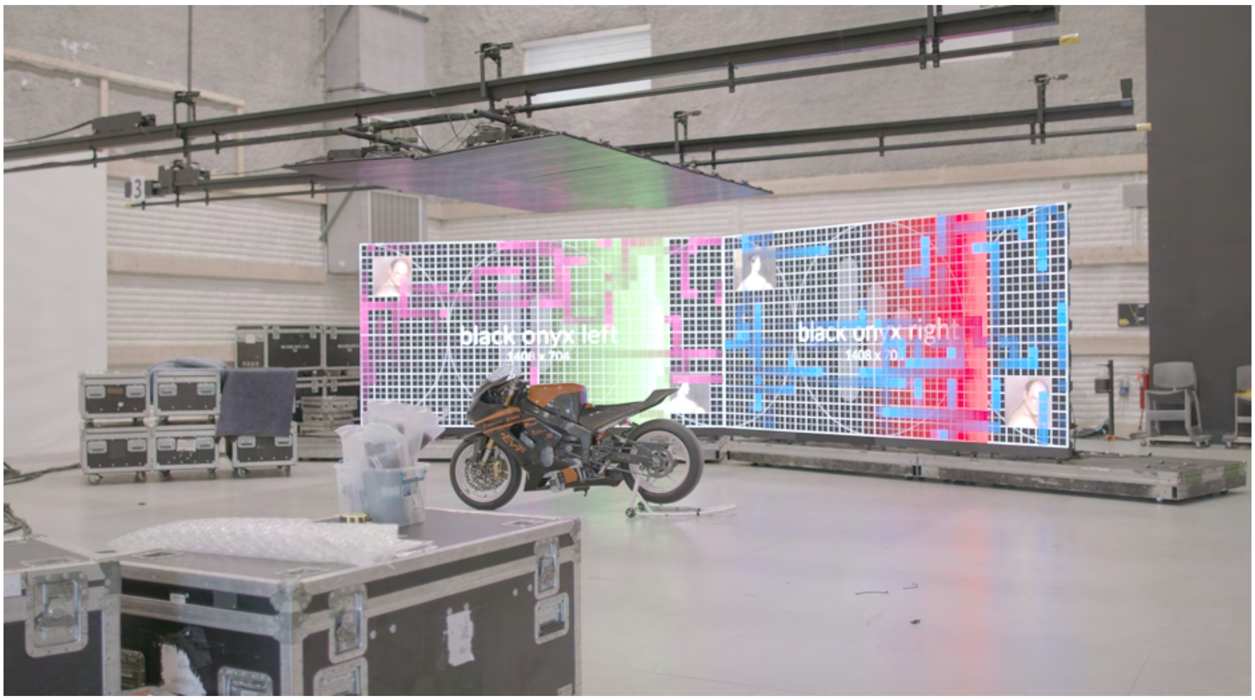


Figure 1: ROE Black Onyx 2 LED walls behind a prop EVT bike.

Considering each panel has a resolution of 192 x 192, our two screens resulted in a resolution of 1536 x 768 each, with bandwidth comparable to standard HD. These panels were predominantly used for environment art as well as supplemental actor lighting and were shot out of focus to prevent aliasing effects in camera.

Additionally, overhead panels used for environment lighting and reflections onto the actor and foreground props. These were Galaxia WinVision 9 with a 9.375mm pixel pitch and a resolution of 64 x 64. These panels are considered low resolution as their purpose is only to provide environment lighting as opposed to being directly in frame.

The MAGIC sound stage is 7,000 square feet and includes 14 individual electrified battens with DMX control that can be raised between 3ft and 35 ft off the stage floor. The Roe walls were typically stationary on the floor while the WinVision panels were rigged to the battens and suspended over the foreground actors.

LED Processors

Two Brompton processors were used to drive the LED walls, one for each of the two main walls. A Galaxia controller and Folsom ImagePro Processor were used to drive and control the overhead panels. The processors take the main signal from the routing matrix in our media racks and map it to the screens.

We used a separate laptop to run a software called Tessera Remote which is a UI that communicates directly with the Brompton processors to display an image on the LED walls. We used this to display test patterns or adjust the attributes of the display such as RGB intensity, brightness, gamma, and color temperature. This was especially useful when color calibrating the LED walls.

OptiTrack System

We chose to utilize an OptiTrack system for real time camera tracking based on its low latency performance. The system uses passive retro-reflective markers tracked by infrared cameras to extract spatial information from the set in real time. The OptiTrack software, Motive, computes the 3D coordinates of the markers by observing the 2D images captured from the synchronized cameras and associating each of their positions to known calibration markers through geometric triangulation. In other words, triangulation compares the 2D coordinates of a specific point in the images captured by all the cameras and estimates its location in the 3D space.

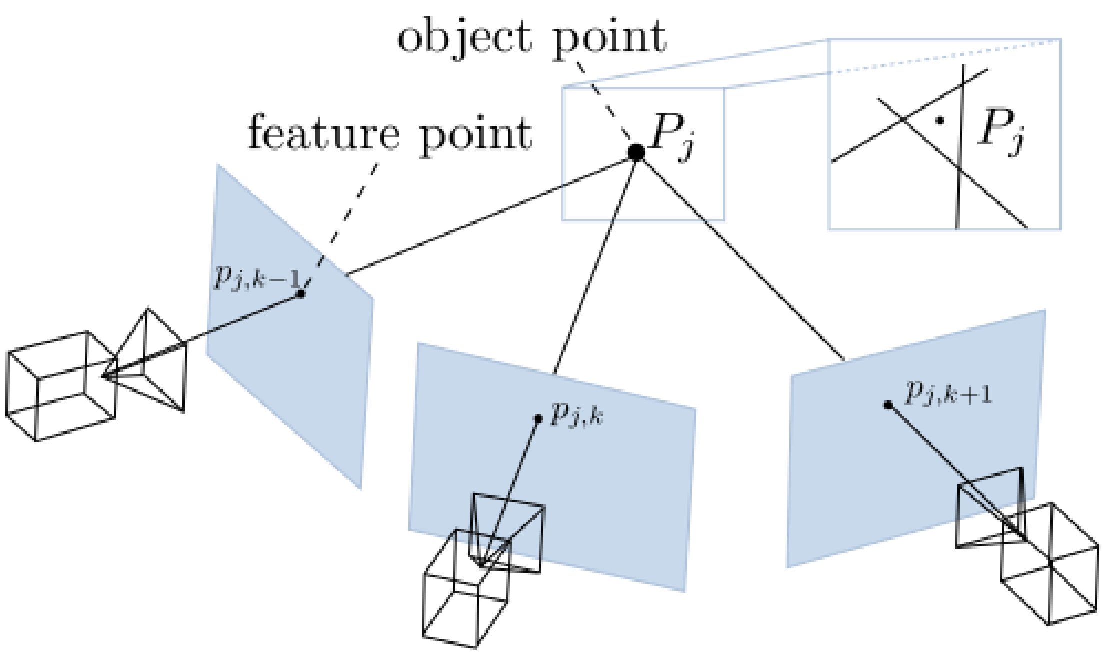


Figure 2: Multiple cameras computing 3D coordinates by triangulation. Image from: http://imagine.enpc.fr/~moulonp/openMVG/coreFeatures.html

For our purposes, a total of (12) S250e OptiTrack cameras were used around the volume which would estimate the position of four markers on our main camera. The cameras have two main components: a ring of 96 850nm IR LEDs and lenses with an 800nm IR bandpass filter. The IR LEDs give off infrared light which reflects off markers in the scene. This reflection is captured by the camera sensor after passing through the IR filter. Each camera is also assigned a specific index that correlates to their location. As a PoE (power over ethernet) device, each camera was connected to the main network switcher using ethernet. This not only provided them with power but was also used to transfer data to the server.

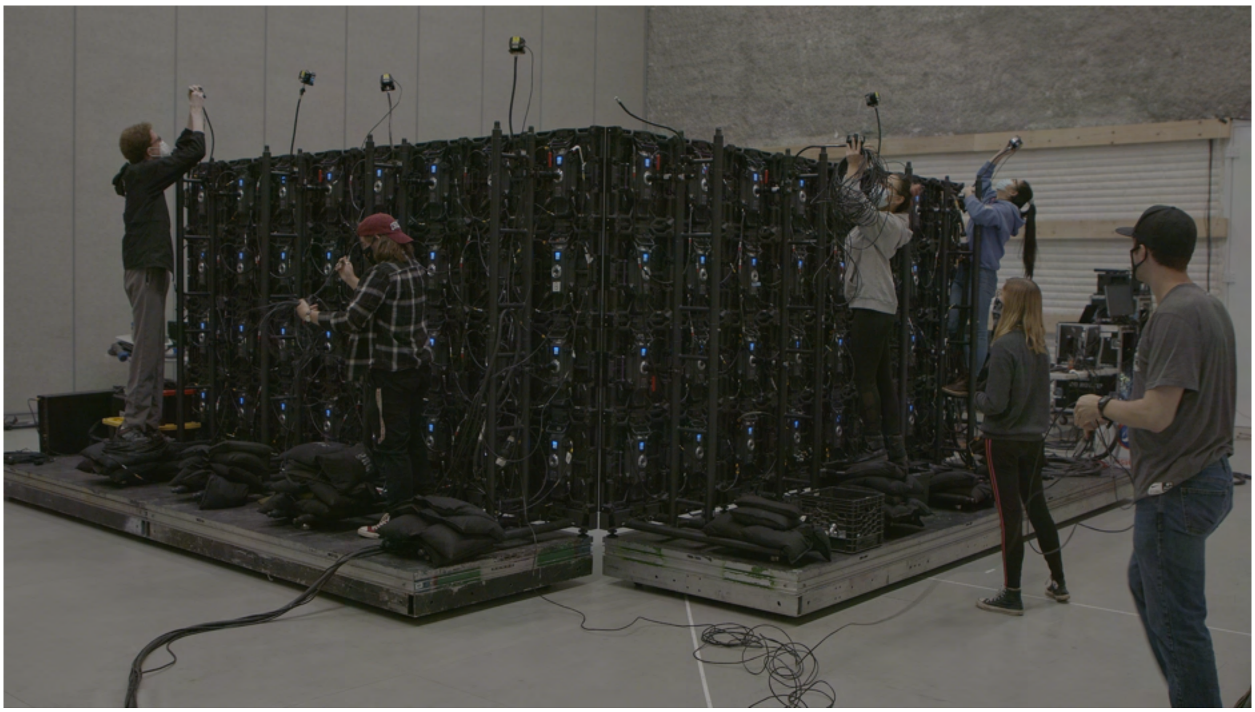


Figure 3: RIT students help mount OptiTrack cameras with the supervision of Optic Sky leads.

OptiTrack offers several plugins which makes it possible to stream real-time tracking data from Motive into Unreal Engine. We utilized the OptiTrack Streaming client plugin to create a OptiTrack Client Origin class in Unreal to communicate with Motive. Using this, we were able to create an OptiTrack Rigid Body component in Unreal to animate our virtual camera according to the movement of a rigid body in Motive, linked to our real camera. A rigid body is a collection of 3 or more markers that are interconnected to each other with the assumption that the spatial relationship among the attached markers remains unchanged in real space. Each rigid body has a Trackable ID value which corresponds to a matching Streaming ID in Motive. This will allow for the virtual camera to achieve correct orientation and enforce render parallax based on the perspective of the real camera.

Projection System

To support other custom filming configurations, a Christie Boxer 4K20 projector with a 30ft x 17ft Draper StageScreen rear projection screen was also experimented with for the production. The 4K resolution (4096 x 2160) was much higher than the LED walls which increased image quality and reduced aliasing artifacts for shots which required this. The projector has a maximum output of 20,000 lumens produced by 4 mercury lamps. It employs Texas Instruments' 3DLP (Digital Light Processing) technology. Each tiny mirror in the Digital Micromirror Device array acts as a single pixel in the projected image, making the native 4K resolution possible as well as a 1800:1 contrast. However, while it has an advantage in terms of resolution compared to the LED Wall, the bulb-driven projector fails to produce a comparably dark black point due to flare in the projection optics and ambient room reflections off of the projection screen. Also, the peak brightness produced on the full frame projection screen (100 nits) was less than the peak LED Wall brightness (800 nits), though the image could be zoomed to a smaller proportion of the screen area if greater brightness were needed. This, as well as colorimetric differences between the displays will be further investigated as part of our considerations when utilizing these technologies and how the final image could be affected.

Figure 4: Draper StageScreen displaying an Unreal Engine project driven by a PC.

Lastly, the virtual set was rendered to the projector using a separate BOXX PC with A6000 GPUs which had a separate instance of Unreal and nDisplay Listener running on it. This would make it possible to simultaneously view the virtual sets on the LED walls and projector.

Camera and Lenses

The production was shot on the Arri Alexa Mini with Atlas Orion Anamorphic lenses at 80mm, 50mm, and 32mm focal lengths. The Alexa Mini is a popular cinema camera among professional cinematographers and regularly used in high-end production. This camera is the top production kit available at RIT’s School of Film and Animation.

The Alexa Mini has a Super35 format sensor (28.25 x 18.17mm) with a pixel pitch of 8.25 microns. We shot in 24fps at 4.5K anamorphic and used the Prores 4444 format for storage. The anamorphic lenses were used to compensate for the short height of the LED walls since we wanted to make sure that the top edges were not visible in frame. It gave us a wide angle field of view despite the close distance to the subject so we were able to keep a short depth of field and keep the display out of focus to combat aliasing artifacts.

Workstations and Servers

For software and data management, we utilized two workstations with custom built PCs and two media racks with our network switches, LED processors, and Disguise servers. One PC was used to handle the Optitrack software and calibration while the other ran the Unreal project with the virtual set. Each rack had their own uninterruptible power supply (UPS) that gave us 5-10 minutes of battery backup in case of power failure. Refer to the Appendix for a signaling and routing diagram (Figure 64).

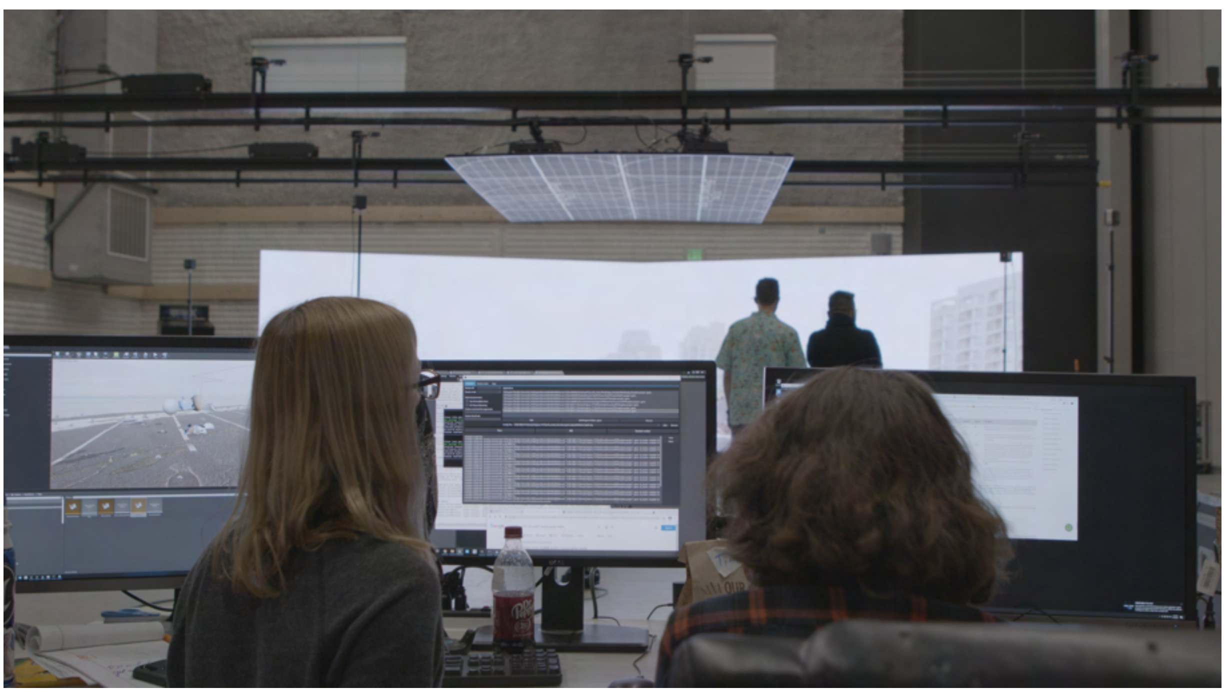


Figure 5: Workstation screen setup with Motive on the right monitor and Unreal Engine and nDisplay on the left and center monitors respectively.

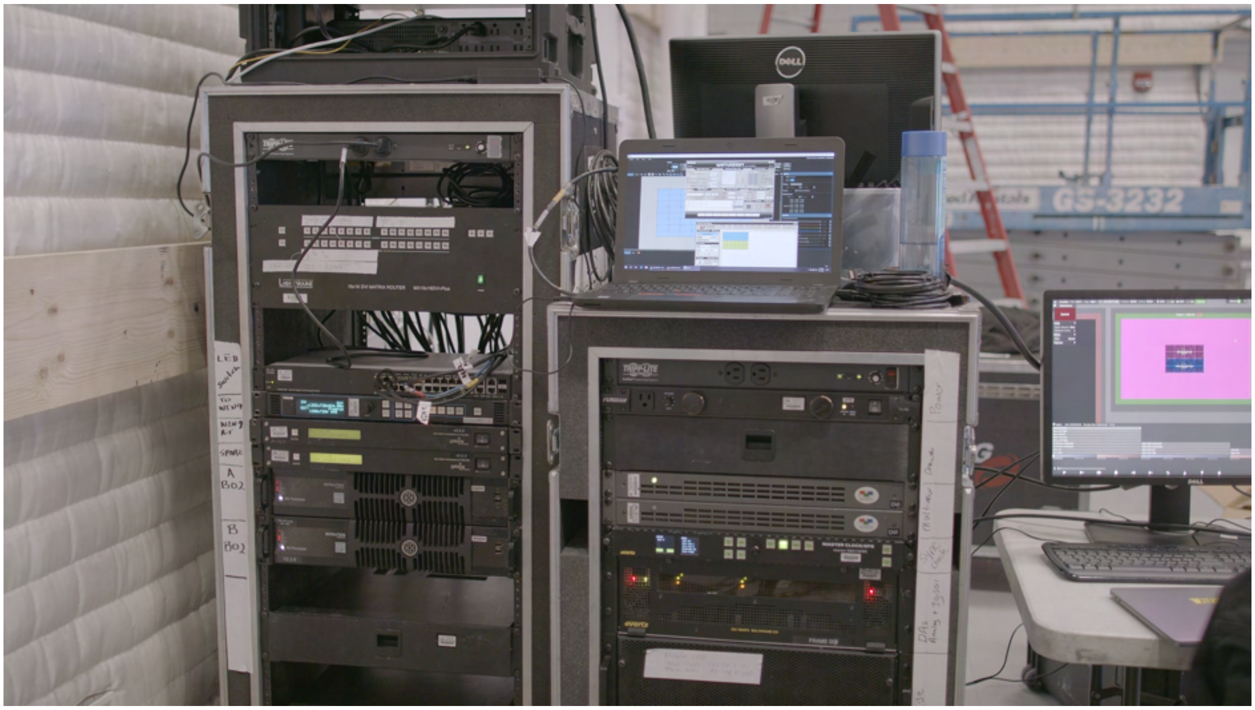


Figure 6: Our two server racks housing all data and software management hardware and laptop used to run Tessera Remote.

Unreal Engine 4.26

Unreal Engine version 4.26 by Epic Games was the game engine used to create and run the virtual set which was used both for previsualization and production. With virtual production emerging as a filmmaking trend, Unreal has been a popular solution with many plugins readily available to easily make your own project. In total we had four virtual worlds generated by our virtual art department (VAD): a racetrack, a forest, Pikes Peak, and a stage. The racetrack set we used was a custom environment with some modifications through assets found in the Unreal marketplace, such as trees and grass. The forest and Pikes Peak sets were also assets from the Unreal marketplace and modified to the required look and feel of the project with the help of RIT 3D digital design students. Lastly, the stage set was created in Maya as it was just a simple infinity backdrop. See section IV, page 47 for more details about our virtual art department.

We created our project from Unreal's built-in nDisplay template which automatically provided all the necessary settings and plugins to be able to render our project to multiple displays. While we were aware that there was an in-camera VFX template, we actually found that the nDisplay template provided us more flexibility to create our own mesh LED walls. We also utilized resources from user communities, providing examples on how to optimize the nDisplay plugin, which was the core process of rendering the image to the display.

III. PROJECT MANAGEMENT

Goals

The DEV1 EVT virtual production at RIT was executed against several key project goals.

1. **Understand VP technology and build a curriculum around it**

   One of the main goals of this virtual production is to further understand this technology and identify how to best learn and teach virtual production processes to RIT students in the School of Film and Animation, 3D Digital Graphics program, and the School of Interactive Games and Media. RIT has recently expanded its curriculum to include virtual production classes to introduce film, motion picture science and 3D graphics students to this emerging technology. This project is only the beginning of virtual productions at RIT.

   Bringing virtual production technology directly into RIT's campus will provide students with hands-on experience and will grant them a great advantage as contributors to the future of filmmaking.

2. **Create a VP workflow solution at RIT**

   The experience gained through this project will allow RIT to create an optimized virtual production workflow that can be documented for future use. Students can use the documentation to create their own virtual productions which ensures a greater understanding and awareness of the technology amongst the student body.

3. **Lead and produce a pilot VP**

   In producing their first virtual production, Optic Sky Productions will have a better understanding of a virtual production workflow and will be better equipped to plan and produce future productions of this style for potential new clients.

4. **Promote RIT's Electric Vehicle Team**

   The final product will be used to promote the RIT Electric Vehicle Team's new electric dirt bike, the DEV1. The video will spread awareness of this student-run project and hopefully gather more sponsors and funding for their work.

5. **Assess difficulties in shooting a VP**

   Lastly, since this is the first virtual production produced by Optic Sky and RIT students, the experience will be key in assessing both the difficulties and benefits of this style of filmmaking. With this knowledge, the team will be able to make more informed decisions about when to use these techniques and how to use them if needed. Specifically, this production aimed to tackle three main challenges: shooting in multiple environments, creating different lighting scenarios, and working with fast-action handheld camera movements. Some of the questions to be considered in the pre and post production include:

- Can virtual production techniques, independent from traditional VFX, achieve photoreal “final pixel” content? Does real-time rendered content appear photoreal on camera?
- What are concerns and solutions in color management when dealing with two different display technologies?
- How can lighting be optimized to believably merge the virtual and physical worlds together on camera?
- What workflow challenges arise from integrating new technologies together? How can the system be refined to be usable and adaptable on a production set and for students in a film school?

These will be revisited more in depth in sections IV and V.

Timeline

Our schedule comprised 11-weeks to complete “DEV1,” starting on March 22, 2021. Most of the 11 weeks included acquiring, calibrating, and testing new virtual production equipment, in addition to traditional pre-production processes such as casting and art design. The mentality in a virtual production style workflow focuses heavily on previsualization and a “”fix it in pre-” attitude. In contrast with traditional production, which is a very linear and structured process, virtual production allows for a more collaborative and fluid filmmaking style. Previsualization, or previz as it is often called, eliminates the uncertainty surrounding what a shot would look like after compositing in post. Since the imagery is closer to “final pixel”, filmmakers can make creative decisions much earlier in production, when the entire team is present.

We shot over a period of 2 days on RIT’s campus at MAGIC Spell Studios. The post production phase then lasted approximately 4 weeks after shooting. This included editing, color grading, and additional motion blur passes needed in VFX. The final deliverable was presented to stakeholders on June 4, 2021.

Budget and Projections

Partnerships

Equipment: PRG, Expressway Cinema Rentals
Crew: Optic Sky, RIT students
Location: RIT campus

This project was made possible by the many partnerships that contributed resources, equipment, knowledge, and time. RIT's megagrant, provided by Epic Games, allowed the school to begin implementing a virtual production curriculum. PRG primarily provided all the hardware such as the LED walls and media servers while Expressway Cine rentals provided the lenses and extra camera accessories. The crew consisted of Optic Sky department heads as well as RIT students enrolled in RIT's first virtual production courses. This production was primarily a research collaboration funded by the contributing partners. While we kept track of production expenses to support our research investigation, many aspects of our cost model would be considered inconsistent with a standalone production. This is due to extra time and resources devoted to instructional delivery, training, and equipment experimentation built into the classes themselves. Because this is not representative of a typical commercial endeavor, we will not provide specific analysis or conclusions.

Resource Comparison of VP to Greenscreen/Traditional VFX

In setting up the virtual production, we did note our budget focused the majority of expense to preproduction and previs tasks. As a contrast, if we were to have implemented a traditional greenscreen workflow, we would have likely devoted most of our expenses to post-production work. In terms of our production, the entirety of the shots relied on some sort of virtual background effects. In this case, if we had taken a greenscreen approach, we would have spent a large amount of time in post compositing a pre-rendered image behind the actor. Not only would this limit the live creative freedom that a real-time virtual set provides, it would also impact the creative control of the filmmaker and DP versus a VFX artist. Our goal in all production decisions was to support the cinematic authenticity of moving a camera around a set and reacting to exactly what we were shooting.

Department Leads for DEV1

Our crew consisted of members of Optic Sky as head of departments while most of the assistant and support positions were filled by RIT students.

- Optic Sky
    - Producer: Christopher Hussar
    - Director: Sullivan Slentz
    - Co Director: Tim Stringer
    - Director of Photography: Aaron Gordon
    - VP supervisor: Emily Halderman
- RIT Faculty/Staff Support
    - David Long

- Mark Reisch
- Flip Phillips
- College of Art and Design IT Support Team

IV. PRE PRODUCTION & WORKFLOW ENGINEERING

Equipment Testing and Calibration

To be able to develop a new production-ready workflow, we first had to understand how each element in our system functioned. During hardware setup, IT support was used to test different network configurations for the full system. Then, nDisplay profiles and VAD workflows through Unreal Engine were refined and optimized, and we learned how to properly calibrate the OptiTrack system to reduce camera tracking errors and latency. Additionally, we had to consider how to tie all these individual components together in a way that both provided predictable results and was flexible enough to meet the changing needs of a production set. These issues were further addressed in our integration testing. Lastly, we considered how the final image rendered by our workflow was impacted by the type of display technology we used, as well as how to mitigate the visual differences in each shot presented by switching between shooting against the LED walls and the projector.

Because of RIT's emphasis on technology and engineering education for film workflows through the motion picture science program, we employed students with that background to run more complex technical assessments of optics, aliasing, and color management. Specifically, we performed a series of tests in order to characterize the colorimetric behavior of the LED walls as well as the spatial artifacting associated with filming a subsampled display with a digital sensor. A deeper summary of this work follows.

Figure 7: Concept map of the main components in our workflow and their individual attributes.

OptiTrack Calibration and Startup

Calibrating the OptiTrack system was a simple, but necessary step in our workflow. OptiTrack has an extensive documentation on their website which we followed and found very useful while troubleshooting.

First, we ensured that all the cameras were properly mounted and facing the center of the volume. The cameras must remain in the same fixed position to avoid constant recalibration. They also have to cover the entire volume so that they are capable of detecting a marker anywhere on the set. Since the IR cameras rely on reflective material to detect a marker, any reflections that do not want to be detected must be masked. For example, athletic shoes, or reflections off the surface of the bike we were shooting could be candidates for masking. The Masking tool in Motive applies red masks over the unwanted reflections seen from the 2D camera view, and all of the pixels in the masked regions are entirely filtered out.

Figure 8: A Motive panel shows each camera’s view of the detected markers.

Wanding was the core process that sampled calibration data into Motive. We used the CW-500 calibration wand consisting of 3 markers. The wand was waved in front of the cameras repeatedly, allowing all cameras to capture several sample frames of the markers’ position. Motive then computed their respective position and orientation in the 3D space. We generally collected at least 2000 samples for each calibration.

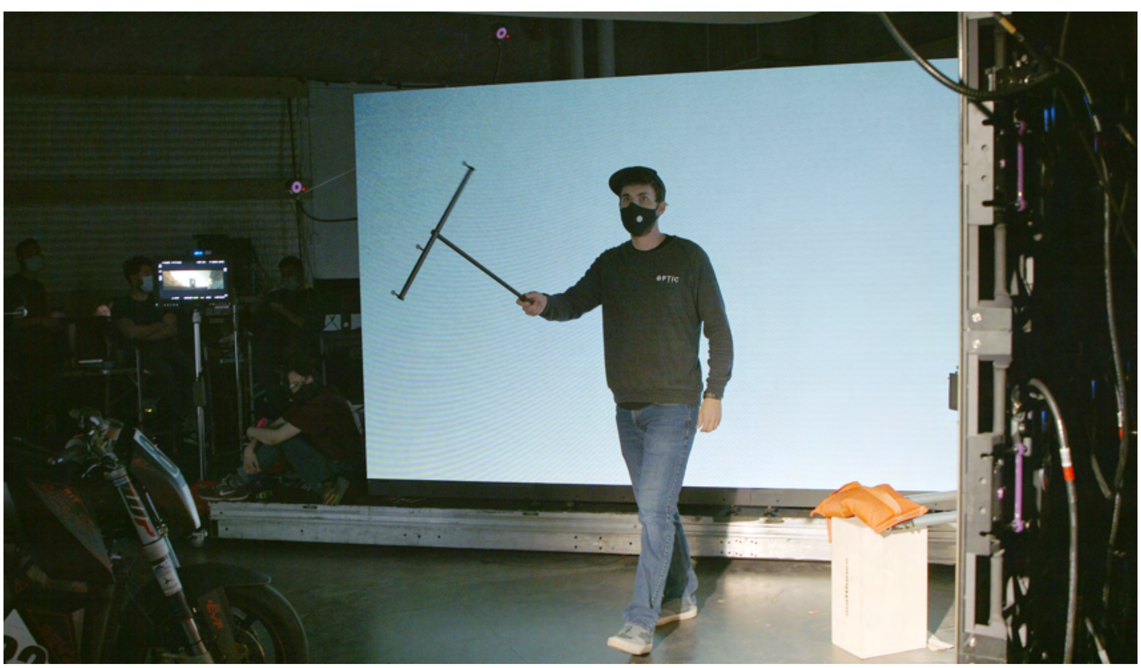


Figure 9: Co-Director performing the wanding process to calibrate the OptiTrack system.

Lastly, we used a CS-200 Calibration Square to set the ground plane of the volume. The position and orientation of the calibration square was referenced for setting the 3-dimensional coordinate system and origin in Unreal.

We had no major difficulties in setting up the tracking software since it is very heavily documented on the manufacturers website. It includes documentation for the plug-ins as well that we would use to stream the captured tracking data from Motive into Unreal.

LED Wall Characterization and Calibration

We performed a variety of tests to better understand the nature of the LED panels and characterize their behavior. These tests included observing the relationship between moire patterns and aliasing as a function of distance from the subject, focal length, and aperture. We also characterized the walls' chromaticity, luminance, EOTF, and spectral energy as a function of viewing angle and ramp intensity. We further assessed uniformity of color behavior as a function of spatial location across the full wall segments.

- **Aliasing Reduction Test**
  Aliasing, also known as a moire pattern in virtual production, is a visible interference pattern caused by the convolution between the different sampling frequencies of the camera sensor and the display. In our case, the LED walls not only conveyed the sampling frequency of the rendered image, but also the frequency of their physical pixel grid structure. However, there are ways in which we can reduce and eliminate the effects of this phenomenon through understanding the geometric optics of a camera. As you deviate from the ideal focus plane in a lens, the frequency response of the optical system begins to degrade, resulting in what we perceive as blur. This frequency response can be defined via the modulation transfer function (MTF) of the imaging system, where the highest spatial frequencies are correlated to sharp and highly resolved details in an image, especially in edges. Sharp detail is lost when the higher MTF frequency response is degraded as scene objects locate outside the focus plane. In photography, this phenomena manifests as the more familiar concept of depth-of-field. As objects in a scene are located further from the plane of optimal focus, they exhibit a blurring often used to aesthetic effect. We can use this to our advantage in filming against the LED walls to prevent aliasing caused by the physical LED structure. By eliminating high frequencies in the filmed walls, they cease to interfere with the sampling frequencies of the camera sensor.

  As a review, depth of field in photography is defined as the distance between the nearest and the farthest objects in an image that are in acceptably sharp focus. It can be manipulated mainly through 3 different parameters: distance from the subject, focal length of the lens, and lens aperture. There are other impacts from sensor size that we addressed by using two different cameras in our analysis. We devised this test based on working parameters for our film set to demonstrate what role these attributes play in manifesting different interference patterns on our display, as well as to hypothesize and prove if the optimal combination of parameters provided us with an image with the least visible aliasing.

Figure 10: RIT students focusing Arri Alexa Mini on a subject for aliasing analysis with a test background image on the LED walls.

In our tests, there were 16 permutations of the variables in total, in four main categories: camera type, distance from the subject, focal length, and f/number.

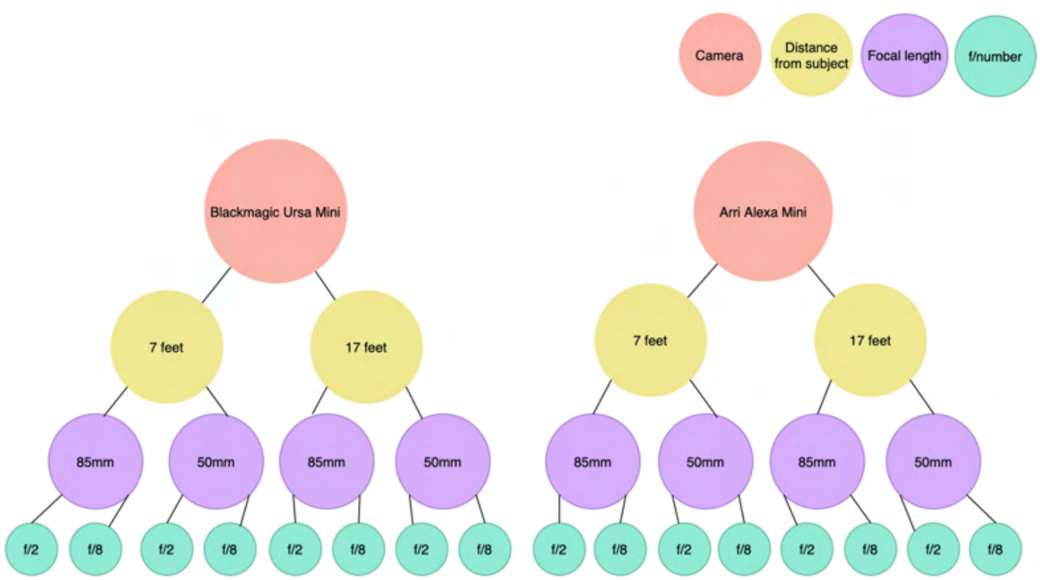


Figure 11: Permutation diagram of the different variables observed.

We tested an Arri Alexa Mini versus a Blackmagic Ursa Mini 4.6k to observe the difference in moire patterns produced by two different sensor sizes. The Ursa has 11.9 Megapixels in its 25.34 x 14.25mm format, meaning it has a pixel size of around 5 microns. The Arri Alexa has 8 micron pixels in its larger 28.25 x 18.17 mm size sensor. Since we used the same lenses and working distances for both cameras, which have slightly different fields-of-view due to sensor dimension, a different sensor configuration effectively means the Alexa has a fewer number of pixels to image a larger number of pixels off the wall, in other words, a different frequency sampling. This will have an effect on the manifestation of the aliasing in the two cameras that we will see in our results. Each camera was tested with a 50mm and an 85mm lens at 7 and 17 feet distance from the subject. For each of these, shots were recorded at different apertures of f/2 and f/8.

Prior to our experiment, we expected that the least visible aliasing would be achieved by manipulating the parameters to create the shallowest depth of field. The variables can be related to depth of field by equations 1 and 2 below.

$$h = \frac{f^2}{c * f/\#} = \frac{fa}{c}$$

(1)

Hyperfocal distance is the focus distance giving just acceptable sharpness when a lens is focused at infinity, *a* is the aperture diameter, *f* is the focal length, *f/#* is f-number and *c* is the diameter of acceptable circle of confusion on the sensor.

$$DOF = \frac{2d^2}{h} \quad \textit{(for } d \ll h\textit{)} \qquad (2)$$

Depth of field, where *d* is the distance to the subject from the camera in scenarios where focus is significantly closer than the hyperfocal distance

Considering these equations, we see that DOF will be inversely proportional to the square of focal length, directly proportional to aperture f/# and proportional to the squared distance from subject and aperture, given only one variable is changed at a time. From this, we hypothesized that the shallowest DOF would be created by opening the aperture (lower f/#), using a longer focal length, and setting a shorter distance to the subject. Additionally, knowing that the circle of confusion is proportional to sensor size due to magnification, we expect cameras with a larger sensor size will produce a shallower DOF, assuming field of view and distance from the subject is kept constant (though this isn't optically achievable unless the focal length is varied with sensor size).

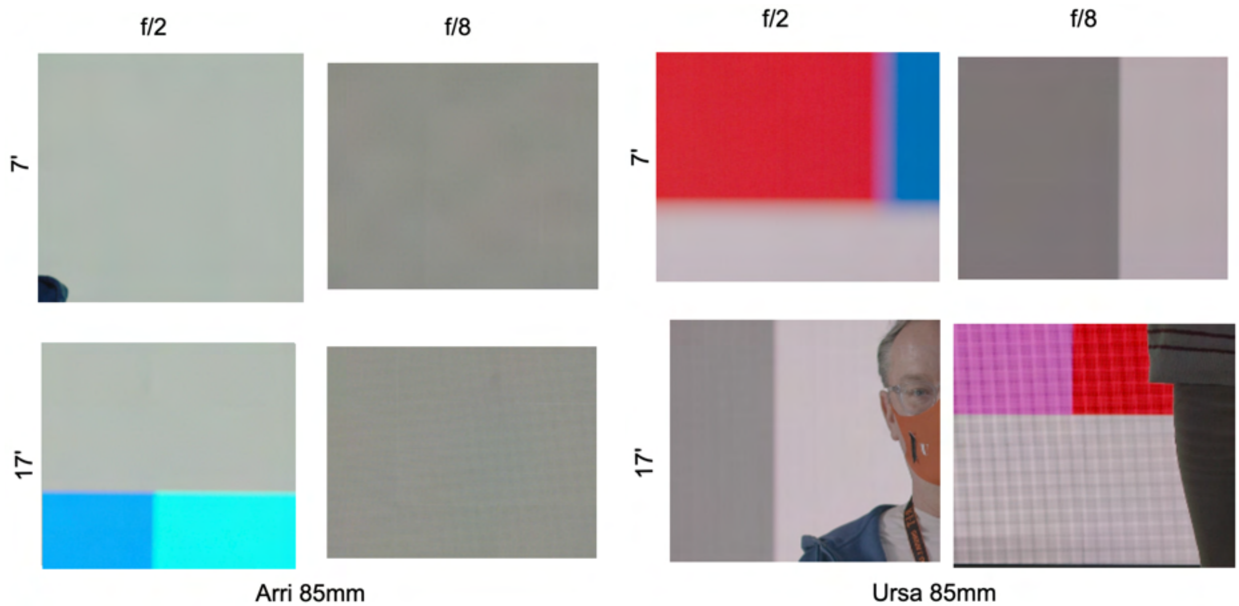


Figure 12: Visual comparison chart of the manifestation of aliasing between the Arri Alexa and Blackmagic Ursa at a focal length of 85mm.

Overall, the combination of optical parameters with the least aliasing we gathered from our results was the Alexa with an 85mm lens at f/2, 7 feet from the subject. This was exactly what we were expecting from theory. However, it was interesting to see the differences in how the interference pattern manifested between the two sensor types as well as the frequencies that were preserved in different scenarios and what to expect when photographing against a large LED background. In our results, the aliasing manifested spatially on the Ursa and more chromatically in the Alexa. Looking at the 17 ft, f/8 condition on both cameras at 85mm, the Ursa seems to show a higher contrast interference pattern while the Alexa footage has a more subtle, but chromatic result.

Figure 13: Closer look at the aliasing on the Arri Alexa (left) vs the Blackmagic Ursa (right).

We hypothesized this could be due to the demosaicing process of the Alexa, which differs from the Ursa. Demosaicing is the interpolation of color data inside the camera used to create a full color image from a color filter array with RGB samples. Not much thorough documentation currently exists about the aliasing on virtual production displays as a function of camera sensors, and we hope that bringing more attention to these differences enables professionals to respect how camera choice can impact final aliasing quality for in-camera VFX.

It should also be said, of course, that aesthetic intentions will most likely dictate the shooting parameters and optical configurations explored in an actual production shot. Our intent in this study was to understand the working limitations imposed by aliasing artifacts and to aid our creative teams in refining solutions to co-optimize the final image quality with creative intent.

- **Spectra, Luminance, and Chromaticity Tests**
  We also wanted to understand the LED wall's colorimetric behavior in terms of spectral emission profile, electro-optical transfer function (EOTF), luminance output, and colorimetry. For these experiments, we measured the radiometric output as a function of viewing angle, ramp intensity, and spatial location. All of our measurements were taken with a PR655 spectrometer.

Figure 14: An RIT student and faculty member take colorimetric measurements using the PR655 spectrometer.

First, to understand how the viewing angle affected the image, we took measurements off one side of the wall from 30 to 150 degrees in 20 degree steps. In this case, 90 degrees would be perpendicular to the wall and would be treated as the optimal or ‘normal’ camera angle. We repeated these for the native red, green, and blue primaries as defined by the Tessera Remote UI, as well as for a white frame.

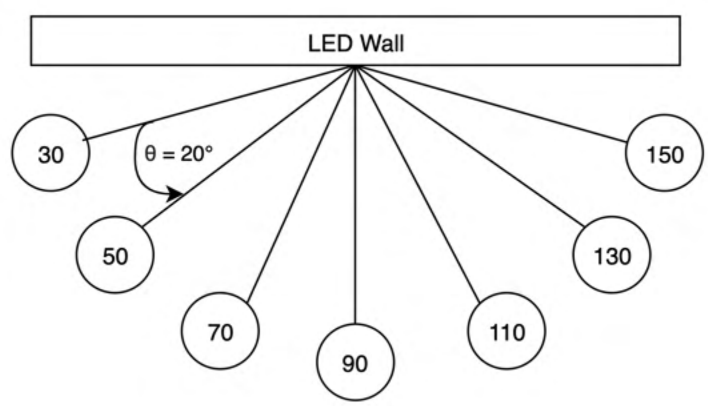


Figure 15: Overhead diagram of viewing angles measured.

From our measurements, we concluded that changing viewing angle mostly affects the chromaticity of the white point and overall luminance of the wall rather than spectral energy. In the case of luminance, we observed that the brightness of the display falls off considerably at wider viewing angles. This was somewhat expected as display screens are typically not perfect lambertian, or diffuse, surfaces. The brightest value was recorded at the 90 degree angle, the surface normal. We can see this in figure 16, where the normalized luminance of the LED wall decreases as the viewing angle deviates from the surface normal.

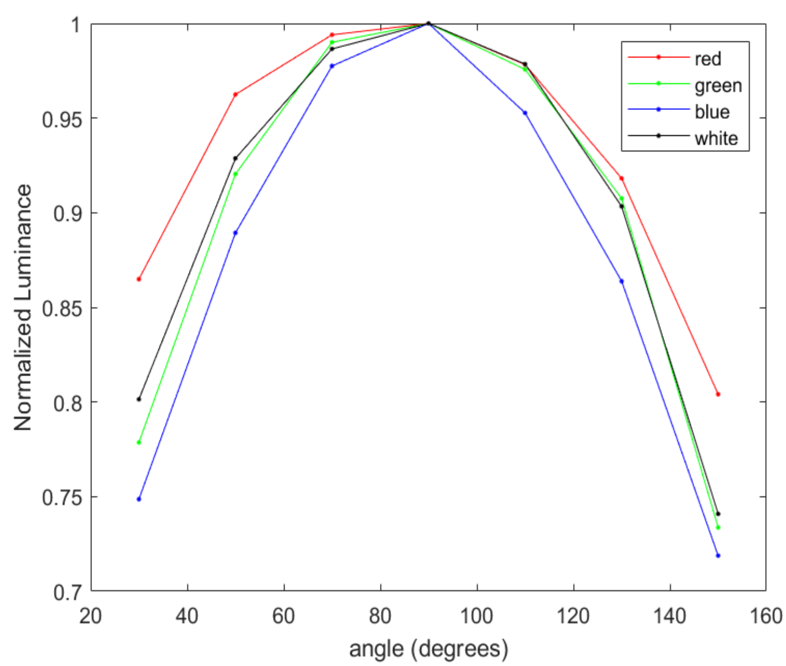


Figure 16: Normalized luminance as a function of viewing angle on the LED wall.

Looking at the most extreme viewing angles of 30 and 150 degrees, the blue primary's luminance falls off to 0.75 for a camera viewing at 30 degrees and approximately 0.72 at 150 degrees. This equates to just less than half a stop of exposure loss in this record for these extreme filming angles, which is likely to have a notable impact on exposure metering and color reproduction.

This ultimately means that when treating a display as a scene for virtual production, different camera angles and movements will be impacted by its non-lambertian behavior. For example, if you are intending to closely replicate the real world with a game engine, each of your assets or objects will have their own radiometric properties and interactions with a light source as a function of illumination and reflection angle (see *bidirectional reflectance distribution function*). However, once you render the virtual world to the LED wall, the display's own physics come into play as an addition to the rendered behavior of your scene. So in shooting at wider camera angles from the normal, the reflections of objects in the scene may reduce more dramatically than expected due to this interaction.

In terms of colorimetry and spectral emission, there was not an obvious trend in changes for the individual color primaries (beyond the power losses overall described above). However, we observed that there is a strong correlation for the white point chromaticity as a function of deviation from the optimal viewing angle. This could be predicted by the lack of uniformity in angular losses in the red, green, and blue series.

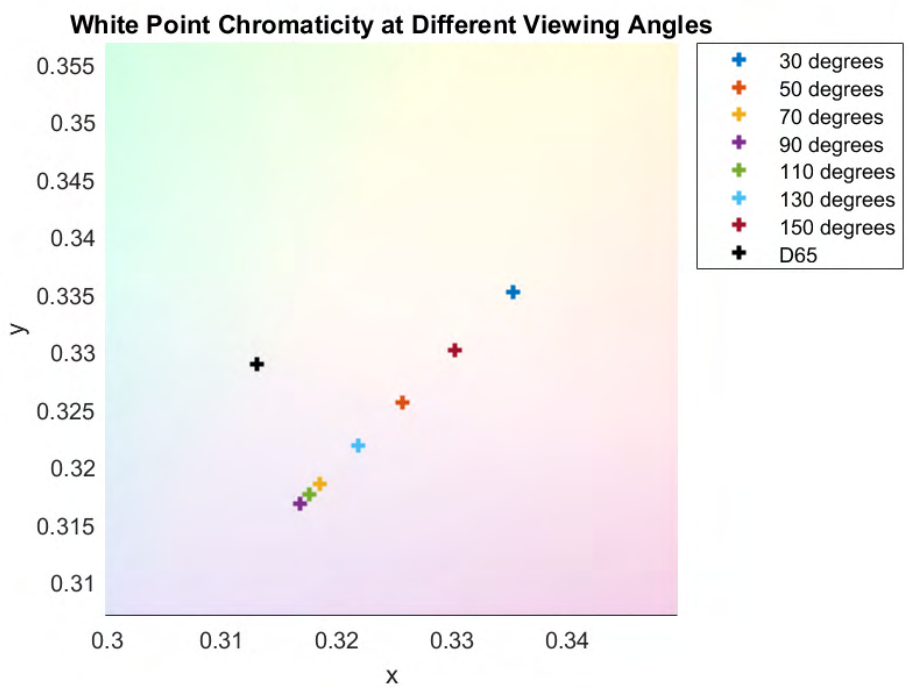


Figure 17: Chromaticity of the white point as a function of different viewing angles.

As seen from figure 17, the white point shifts in hue across the angular detection series such that the correlated color temperature of the white point is warmer at wider angles. We can quantify the degree of hue difference from the optimal viewing angle by performing a deltaE comparison. DeltaE is a measure of change in human visual perception of two given colors.

	White point luminance and color differences based on measurement angle					
Angle from left side of wall	30	50	70	90	130	150
Normalized luminance	0.80	0.93	0.99	1.00	0.90	0.74
deltaE from normal	10.6	4.7	1.0	0.0	4.5	12.1

Table 1: Colorimetric behavior of the white point at different viewing angles of the LED wall.

From Table 1, it can be seen that the largest perceived color difference of the white point at 60 degrees from the optimal viewing angle is a deltaE value of 12.1. Generally, a deltaE value of less than 2 is indicative of non-perceptible color differences. Again, our purpose for characterizing our LED display in this way was to understand its behavior as a scene and consider these colorimetric behaviors when planning out camera angles and movements. While we cannot change the display's inherent physics, this knowledge allowed us to make more informed, engineering-driven decisions about our cinematography and to anticipate the darker and warmer image reproduction as the camera was dollied to more extreme angles in select shots.

Next, we observed the behavior of the colorimetry as a function of the intensity value driven to the LED wall. We took measurements across an 8-bit drive encoding of 32 to 256 with 32 step increments for the color primaries; the white point was driven across an intensity range of 16 to 256 with 16 step values. In this experiment, our main purpose was to calculate the EOTF (sometimes simplified to a power-law function known as 'gamma') of our display. Historically, IEC, SMPTE and ITU standards have enforced a 2.2 (sRGB) or 2.4 (ITU Rec.1886) gamma for displays (though the actual expected behavior is notably more complex than a simple power law). However, in virtual production, the purpose of our display is to act as a scene being captured by a camera, not as a standard display, therefore we would actually need to know the EOTF of the display so as to properly prepare images for rendering in a net radiometrically-linear fashion.

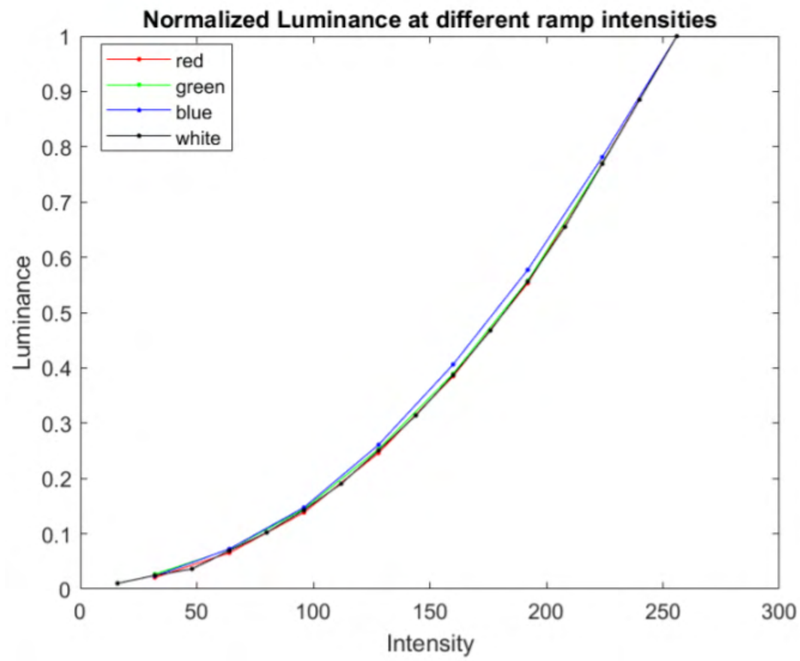


Figure 18: Normalized luminance of color primaries and white point of the LED wall as a function of ramp intensity.

As expected, the normalized luminance of the three primaries and white increase as an exponential function as the intensity increases. There is only a slight difference in the blue primary where it seems to have a slightly steeper slope in the middle range of intensities, but otherwise follows the trend. The display gamma was then calculated by taking the log of each normalized axis and finding the slope of the resulting function.

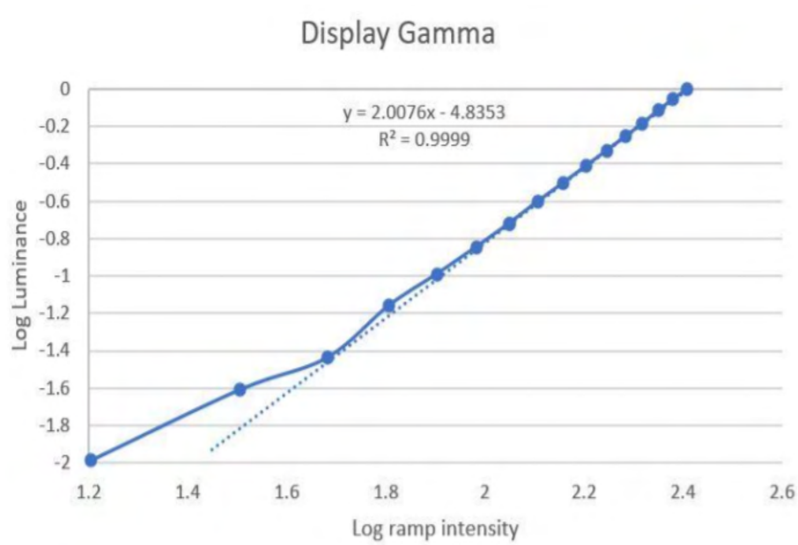


Figure 19: Calculated display gamma from the fitted linear regression of the logarithm of axes in Figure 18.

We found that the slope of the line of best fit, omitting the darkest four data points, was close to 2.0 in each record. The reason for omitting the first data points is that the measurement is more prone to error in integrating the lesser amount of light. At brighter intensities it is easy to see a very linear trend, so the fit was adjusted accordingly.

What this all means for “scene to screen” image rendering intent (opto-optic transfer function or OOTF) from the virtual set to the LED walls is that we must consider both the opto-electronic transfer function (OETF) or how the image to be displayed is encoded, and the electro-optical transfer function (EOTF), how the display decodes that image back to light. For LED walls in virtual production, we know that we cannot use the standard nonlinear video OOTF as defined for ITU Rec.709 cameras and Rec.1886 displays because it was created considering human psychophysical phenomena such as the Stevens effect which dictates that the contrast of an image will appear to decrease with lower image luminance (and most viewed images are radiometrically less bright than the scene light they are created from). To combat this, as well as similar effects like dark surround adaptation and imaging system flare, the Rec.709 OETF encodes signals with a gamma greater than the inverse of the Rec.1886 decoding EOTF to boost the contrast of the displayed image. If we were to use this rendering intent in our workflow, we would be boosting the contrast of the virtual world even before we captured it with our camera. Treating the LED wall as equivalent scene light, we would want to follow a linear OOTF where there is a perfect compensating ratio between the encoding gamma and the decoding gamma. For example, our walls with a 2.0 gamma should be paired with an image rendering that encodes scene light with a 1/2.0 gamma. At the same time, considering the LED walls are still a light emitting display technology that is subject to flare, a goal for our rendering intent could also be to mitigate the effects of ambient spill on the displayed image. These are themes that can be further explored and discussed in terms of developing a standard rendering intent for display technologies as scenes in virtual production workflows.

Lastly, the colorimetry was measured at 9 different locations on the LED wall in order to determine the uniformity of our display. We were curious to know and quantify the range of difference in luminance and chromaticity between the different regions of panels. Similar to our viewing angle experiment, the importance of uniformity lies in that there should be no perceptual variability in how the image looks when shooting at different locations on the screen. This was somewhat true in our luminance measurements, where they were normalized relative to the center of the screen. Figure 20 depicts the luminance corresponding to the different measurement locations on the display.

0.98	1.02	0.97
0.97	1.00	0.99
0.94	0.98	0.95

Figure 20: Visual representation of the variation in luminance values at different measurement locations on the LED wall.

We observed that the bottom left and right side of the LED wall were dimmer than most of the other locations, albeit the difference was no more than 6%. The chromaticity of the white point was considerably more affected by change in location as seen from figure 21 below.

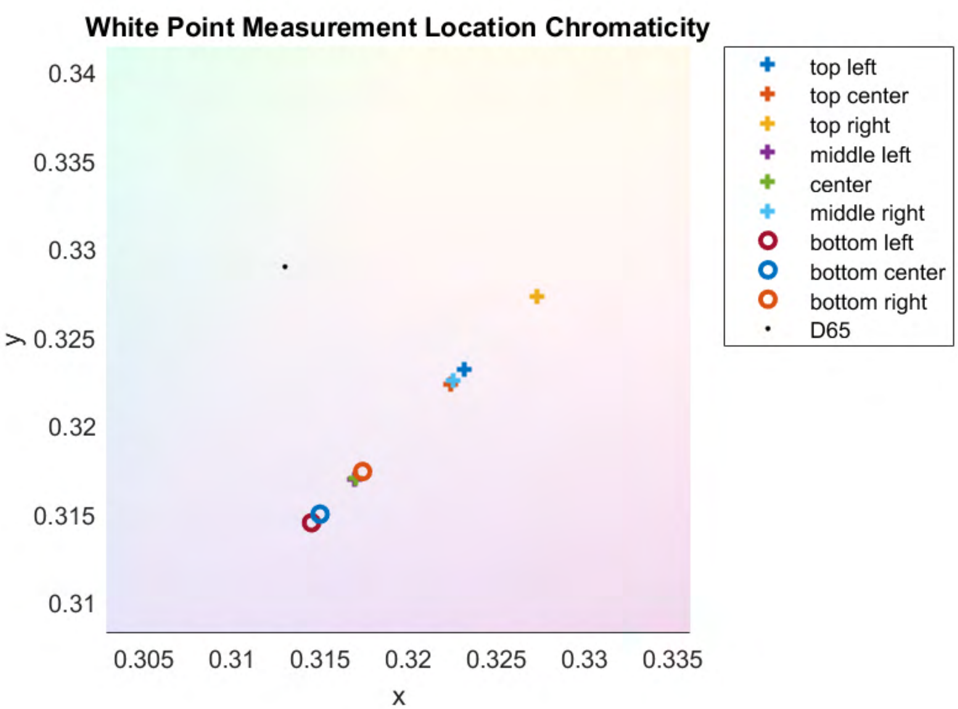


Figure 21: White point chromaticity as a function of measurement location on LED wall.

The top of the display leans towards a warmer correlated color temperature while the bottom is biased towards cooler correlated color temperatures. Specifically, the bottom left is perceived as the coolest white while the top right seems like the warmest white. To quantify the perceptual color difference between the two extremes, we calculated the deltaE to be 6.89. Again, we would ideally want the difference to be less than 2 deltaE units, however since each LED wall is made up of individual panels, there exists an inherent variability between them all. Just like the colorimetric variation with viewing angle, variation with location could affect how the final image is presented. A strict tile-by-tile calibration strategy should be employed when maximum colorimetric uniformity is needed. This would be most important fro uniform and low frequency scene elements where color contouring is easier to detect in camera.

In terms of individual colors, we noticed the green primary had the most variability in its chromaticity at different locations, while the red and blue stayed mostly constant. A more comprehensive table of the deltaE color differences calculated from these observations is available in the Appendix for further review.

- **Color Temperature Calibration**
  During our calibration process, we discussed what color temperature and white point the LED walls should be set to. There were two main areas of thought: one from a display-centric premise and the other from an image capture premise. If we treated the LED walls as a traditional display technology, we would calibrate to the CIE's illuminant D65 white point for viewing, according to Rec. 709 standard convention created by the International Telecommunication Union (ITU). D65 corresponds roughly to the average white that is seen as "neutral" to the human eye in an emissive electronic display, so it is usually used to calibrate TVs, monitors, and other viewed devices. However, we had to consider that we are not directly viewing the display, but rather the camera is. In our case, the LED walls were being treated as the "scene" being captured by the camera. Understanding this, we would set the white point to D55 which is commonly used as daylight for cinematic sets.

  Since D55 and D65 are not radically different, it wouldn't make a drastic impact to prefer one over the other, as long as the camera's white balance and additional on-set lighting is set to match the wall's output. That being said, we ultimately decided to calibrate our white point to D55 to conform to traditional photographic conventions which were defined by several research groups, including Kodak, at the dawn of color photography. They found that the average spectral distribution of typical daylight had a correlated color temperature of approximately 5500K and early photographic films designed for daylight use were white-balanced to this expectation. All major artificial motion picture daylight technology has subsequently conformed to this convention. We felt this was a more appropriate choice considering the application of our LED wall as the subject of our photography rather than a display for direct viewing.

- **Observer (Camera) Metamerism**
  Another concern regarding color management in LED wall workflows is the issue of observer metamerism, or, in our case, camera metamerism. This phenomenon refers to differences in color appearance when viewed by cameras with dissimilar spectral

sensitivities to the color matching functions of the human visual system. When we attempt to use a display, such as an LED wall, as our primary stimuli, the way in which it creates color has a direct impact on the feasibility of creating a metameric match between what we as humans see and what a camera sees. Figure 22 shows the spectral power distribution of the three red, green, and blue primaries of the LED walls.

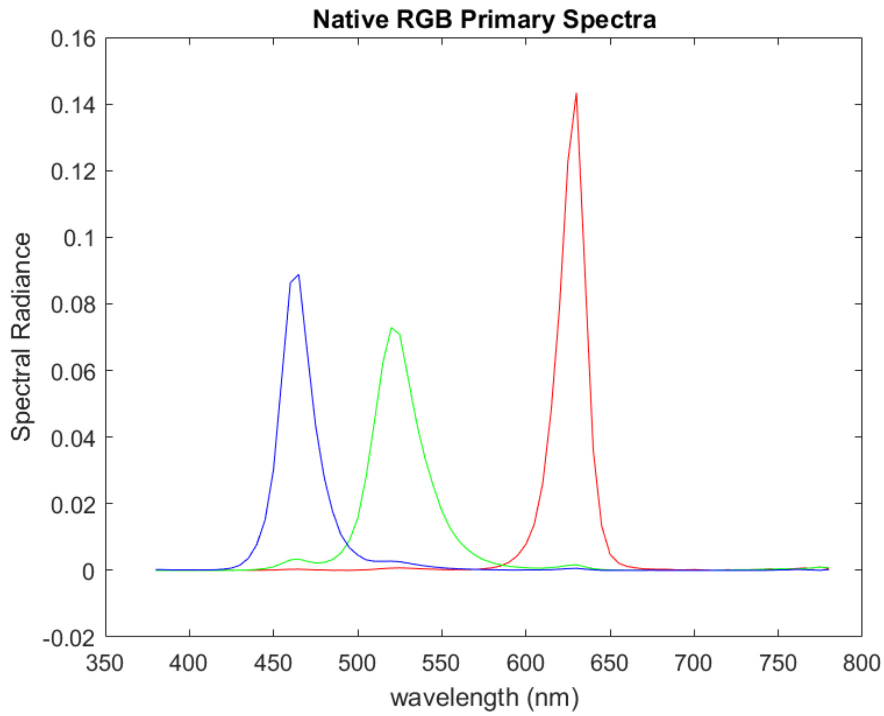


Figure 22: Spectral power distribution of the color primaries of the LED wall.

The narrow emission for each of the color primaries actually exacerbates the variability of exposure integration across dissimilar spectral response functions. This means that even if a colorimetric match is established between an object in the real world and an image of the same object on the display, it does not guarantee a metameric match when captured on camera. Considering that a goal of in-camera VFX is to create imagery as close to “final pixel” as possible during production, this presents a significant drawback if not properly addressed. Future work regarding this issue is to develop a correction matrix to potentially reduce the effects of observer metamerism. This would create a real-time, on-set solution akin to the other benefits of virtual production.

Integration Testing

- **Frustum Calibration**
  The next step after acquiring the real-time tracking data from motive into Unreal was to determine if the virtual camera’s rendered frustum displayed on the LED wall matched the real camera’s perspective and movement around the stage. Since the LED walls present content in a 2D plane as a display, they must create the illusion of depth from the 3D virtual environment based on some starting origin and camera perspective.

In general there are two “frustums” in an in-camera VFX setup. First, the inner frustum is a dynamic picture-in-picture render on the LED wall. It represents the optical perspective of a real camera as a function of focal length and focus distance as it is tracked moving about the scene. The image displayed in the inner frustum will be rendered based on real camera tracking data and optical predictions. This creates a parallax effect that simulates shooting in a 3D environment but with scene light projecting from a flat 2D screen, however, it will only be achieved if the inner frustum is calibrated correctly from the camera tracking data and optical parameters.

Second, the outer frustum is defined as the rendered content outside of the camera’s field of view, but restricted to the area of the LED walls, given a defined observation origin on set. The outer frustum is effectively a “window” into the 3D world from the perspective of an observer standing at the defined origin. The main purpose of the outer frustum is to remain static while providing ambient light and reflections from the environment. In a physical set, this would already be provided by the real physical lighting and environment objects, but for a virtual world they need to be added in to achieve a realistic look. This also mimics how lights and object reflections simply exist and do not change with the moving camera in the real world, hence the need for the outer frustum to be static. For each shot set up, the outer frustum can be changed by moving the perspective to the desired location within the Unreal Engine environment.

Figure 23: Co-Director testing the inner frustum render with a tracked camera.

Our main issue was calibrating the inner frustum in a way that perfectly matched the real camera’s perspective. Difficulties arose in that we had to relaunch nDisplay, our display rendering program, every time we wanted to make a change in Unreal. We did not have any real-time feedback if we wanted to adjust any parameters in the game engine. This led us to a trial-and-error methodology where we characterized how Unreal reads the data from Motive by manually adjusting the tracking parameters and observing how it affected the result. A specific issue we ran into was that the values from Motive were not to scale with the values in Unreal regarding the camera's location. We were able to fix this by manually multiplying the values in Unreal by a correction factor which made it easier to see any errors in tracking. This is less than ideal, however, as we would have preferred a geometrically relevant calibration process to ensure sightlines and perspective were absolutely correct in every configuration.

Another issue arose regarding the smoothness of the virtual camera's movement. Initially, we thought the best way to move around the virtual environment was to move the origin. However, when we did this we found out that the frustum appeared like it was jittering or struggling to catch up with the movements of the real camera. We weren't able to determine the root cause of this problem, or whether it was an issue with our processing power, so we implemented a workaround that would move the world around the origin instead. This seemed to solve the problem fairly well, though additional investigation is still required.

This particular struggle taught us that implementing workarounds and being creative was the best and fastest way to keep to the schedule.

- **Genlock and Sync**
  Genlock, short for generator locking, uses a reference signal from a signal generator to synchronize other video sources together. The importance of genlock and framelock when it comes to a virtual production is even more important because our virtual environment is rendered to screens that have a refresh rate of their own, distinct from the camera's frame rate. In real locations, the physics of light leaving an object and entering the camera is continuous and analog in nature. Without genlock, there could be ghosting between frames meaning that the camera captures the instance between two display wall frames, causing a double-exposure artifact in the image. Further, limitations in real-time motion blur rendering mean that most virtual frames are displayed sharp, only exacerbating errors when double exposure artifacts arise. In the time allotted for production, we were ultimately not able to achieve a full scale genlock system, though robust solutions do exist. We did gain a fair amount of valuable knowledge concerning how we would be able to apply this in a future virtual production at RIT. While we were able to achieve genlock on our main workstation PCs and Unreal using a master time clock on our rack, the main issue was with our Arri Alexa. The camera we used did not have a genlock input, and will need to be retrofit to allow for that. Alternatively, we experimented with using a TinyLockit box synced to the master time clock to feed the timecode into the camera, as well as taking framecode from the camera, but were unable to get acceptable results.

- **Focus Mapping and Double Defocusing**
  In addition to camera tracking, we wanted to have a physical follow focus on our real camera that would pull focus on the virtual camera and environment renderings at the same time. To do this, we had to map the specific degree of rotation on the follow focus to each focus position on the lens. We attached markers onto the follow focus and assigned it to be a rigid body so that the OptiTrack system would be able to estimate the degree of rotation as it was turned. Since the OptiTrack data is transferred to Unreal Engine using Motive, this data was easily accessible and a blueprint could be made to attribute it to the virtual camera's focus. The blueprint specifically worked by pulling focus on the virtual camera based on the degree of rotation of the rigid body. This made it possible for the camera operator to pull focus as if they were in a traditional set while keeping the realism of the virtual set at the same time.

Figure 24: Virtual follow focus rig with OptiTrack markers.

In our workflow, we were essentially working with two cameras, a real one on set and a virtual camera in Unreal. Ideally, these would be programmed to move and act as one, however a consideration we needed to make was how to realistically match the focus on the real camera which our choice of lenses and the focus on the virtual camera that would affect the environment renders. Double defocusing is a term that describes the blur on the virtual background rendered through the virtual camera physics simulation on top of the real camera's depth of field enforced by optical decisions on set. In our case, we set the focal distance on the virtual camera in our Unreal project and rendered this to the LED walls, then shot those walls with a camera and lens system that had a further blur depending on its own physical depth of field and the actual proximity of the walls to the camera. This could pose a problem in that the background can become inaccurately blurred versus what would be expected in a real location set. Not only is this disconcerting for the DP, it also prevents them control of what the final image will look like and changes their artistic intent. Our goal was to achieve as realistic a look as possible which means that we had to adjust the depth of field parameters in the virtual camera on a per shot basis, especially when we changed lenses. In the future we hope to experiment more with different types of lenses and characterize how Unreal's virtual camera responds to these changes. We would be further benefited if there weren't aliasing concerns and we could actually keep the LED walls more in focus in the real camera's depth-of-field.

Co-Director: *"There is just no 'here is the way to do it' right now and everybody is trying different systems, LED walls, processors, you name it. It's kind of the wild west right now as far as how to get started."*

- **Multiple Display Optimization**
  We had the opportunity to use both an LED wall and a large rear projection screen in our virtual production setup. Having two display technologies in our production helped our flexibility in terms of camera angles and coping with sudden technical difficulties. We could switch from one display to another if an issue rendered one display unusable for a period of time. Specifically, the large projector allowed us to capture a low camera angle that the LED wall could not have been used for. On the other hand, the LED wall ceiling panels were perfect for achieving environment reflections on the actor's clothing and visor which would have been less optically dynamic with a projection screen. However, the largest downside we faced in combining these two technologies was that the rendered image would appear different due to their distinct color physics, flare & contrast, and spectral emission profiles as rendered by the spectral responsivity of the Arri camera sensor. Considering this, we would have to compensate for it in post production with additional color grading to match the shots.

Rendering

- **nDisplay Workflow**

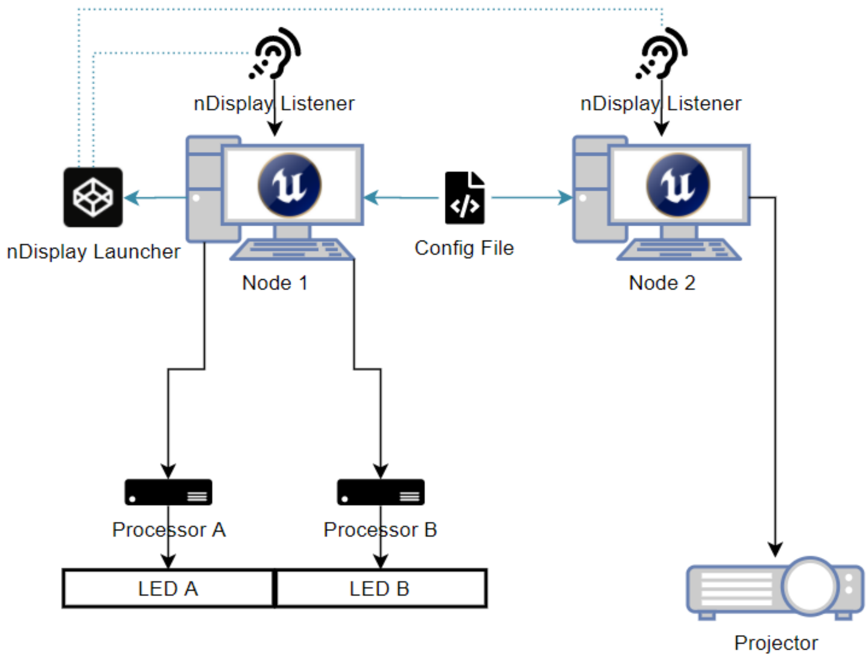


Figure 25: nDisplay signal flow diagram to the LED walls and projector.

The process to display the virtual set from Unreal Engine to the LED walls or projector utilized nDisplay. nDisplay is a built-in program in Unreal Engine that renders the virtual set on multiple synchronized display devices. This can be done from a singular master computer or in a network with several computers called cluster nodes. Each of these cluster nodes drives one or more displays. For our workflow, we had two PCs drive our two different displays, one for the LED wall and one for the projector. We ran multiple instances of Unreal on them and directed each to render to the different screens. nDisplay works with four main components: nDisplay plug-in in Unreal, nDisplay Launcher application file, nDisplay Listener application file, and a shared configuration file. The plugin that works inside Unreal is the basis for communication and synchronization of information between devices in the cluster. It makes sure the display device renders the correct frustum of the virtual set. The nDisplay Launcher and Listener application files work hand in hand to send and receive requests on each computer. The Launcher is the main management application while the Listener processes the Launcher's requests on the local computer. Lastly, the configuration file contains all the settings nDisplay needs to seamlessly render the virtual world on all display screens.

To get started with nDisplay, we first created a mesh of our LED walls in Unreal to represent where the screens are in relation to the origin in the tracking system. Then, we needed to edit the configuration file inside nDisplay to what our specific setup required.

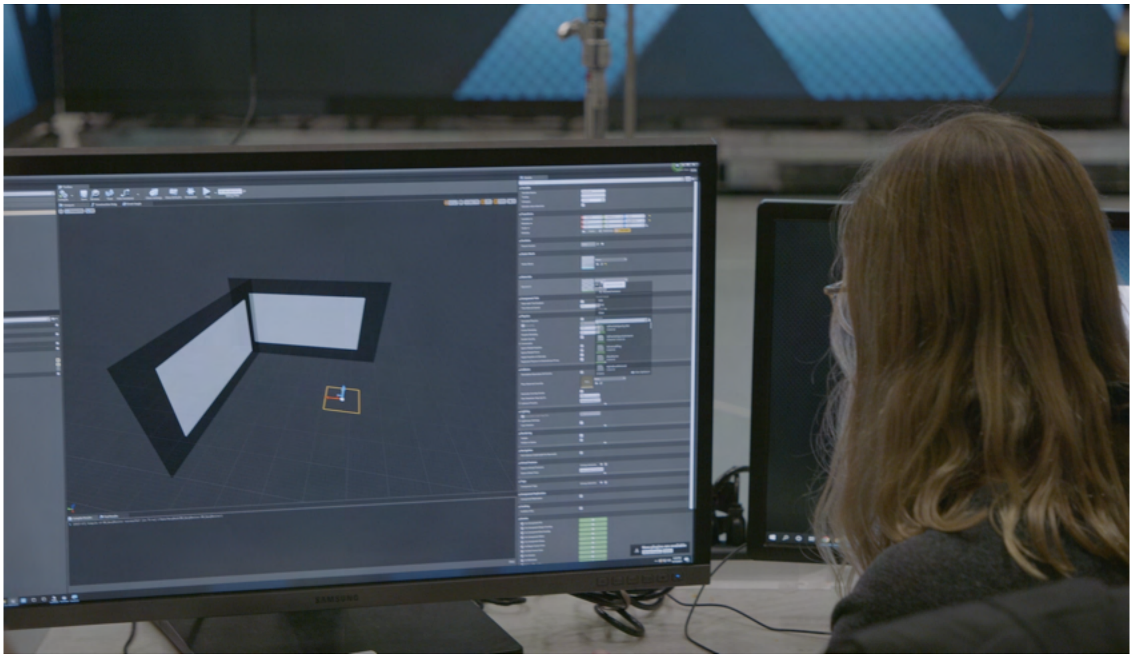


Figure 26: VP supervisor creating custom masking material in Unreal for our LED walls.

For the two LED walls, we had two viewport configurations. Each of which defines a rectangular area of Unreal that nDisplay should fill with a rendered view of the virtual set. These areas were predetermined by our meshes which had been scaled in Unreal to match the specific dimensions of the physical walls. Once the configuration file was set up, it was run through the nDisplay Launcher application. The nDisplay Launcher sends a message to the nDisplay Listener on each cluster node in the configuration file, instructing it to launch the packaged Project. Finally, the nDisplay Listener on each computer will launch Unreal and begin rendering the two viewport configurations. The viewport is representative of the outer frustum which will fill the entire LED screen and provide environment lighting.

Figure 27: Two renders of the Unreal project simultaneously displayed on both LED walls and the projector using nDisplay.

While Unreal Engine has thorough documentation about nDisplay, we had several issues with the integration of all the components in our system rather than the nDisplay application itself. For example, if the aspect ratio wasn't specified correctly in the nDisplay config file, then our LED walls would be completely black. This was a difficult problem to troubleshoot because we did not have specific feedback to debug the system. In the end, we wanted to refine the workflow to a point where it would be usable on set, meaning that it needed to be able to continuously display the virtual set along with any new changes we decided to make. For example, if the director decided to move an object in the virtual world, the system had to be reliable enough so we could do so with confidence.

Co-Director: *"You know, I think we quickly got past the hurdle of 'alright let's get things moving and up on the screen'. It was then, 'okay, how do we refine this to a point where it's usable on set'. We need things to be 1) controllable and 2) realistic, and that's the hard part, getting it ready for production."*

Previsualization in Unreal

As mentioned before, previsualization of shots using game engine technology is one of the greatest advantages of a virtual production. Since game engine technology is capable of simulating real-world physics, all assets created in the software will closely replicate the real world. In our case, we had a 3D model of one of our prop bikes as well as a character for the rider. Using this, we were able to visually experiment with different camera angles, movement, and lighting scenarios in all of our shots prior to production. This allowed us to make creative decisions about shots and what would look best in pre production as opposed to taking up that time on set.

Stage Setup Diagrams and Shot Lists

In addition to traditional shot lists, we had to consider the placement of the LED walls and projector as they would not only affect how our environment would look but also the lighting on set. We had 4 main stage configurations for this production which involved tilting the LED walls at different angles and switching between shooting against the walls or the projector. Each stage setup was chosen with an individual shot in mind.

- **Stage Setup 1**

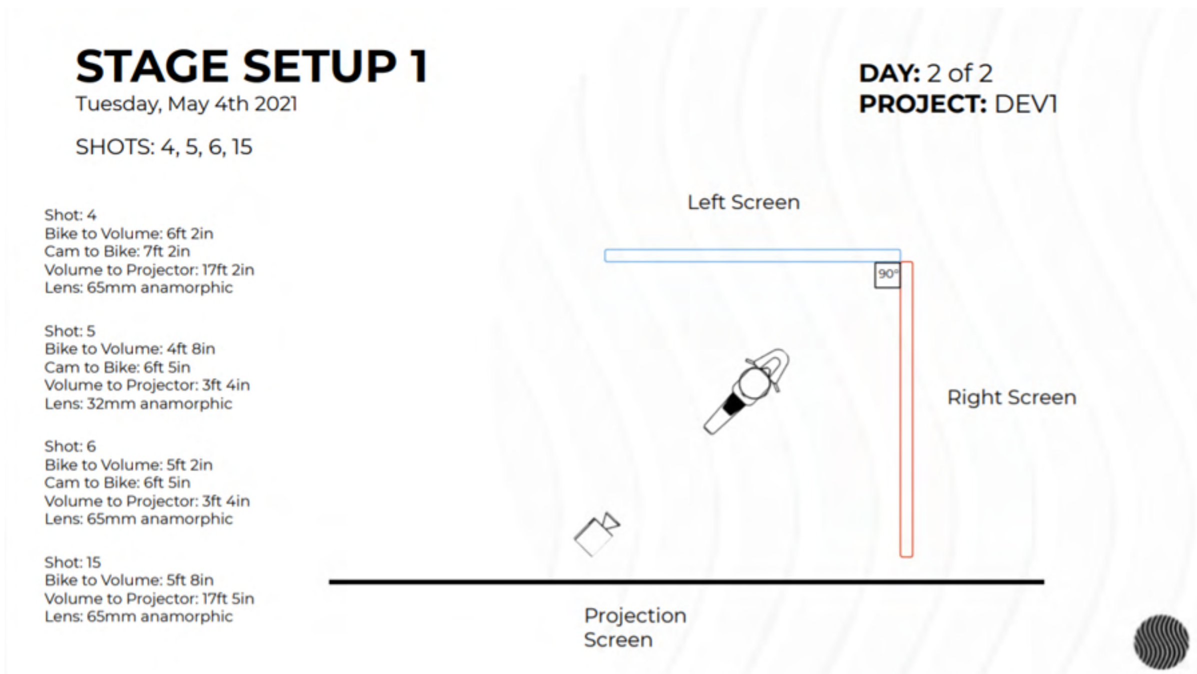


Figure 28: Overhead view of stage setup 1, shooting against LED walls at 90 degrees.

For this setup, the LED walls were set perpendicular to each other at 90 degrees and placed beside the projection screen. This would give us flexibility to either shoot against the left LED screen or the projector while the right LED screen provided the environment lighting and reflections.

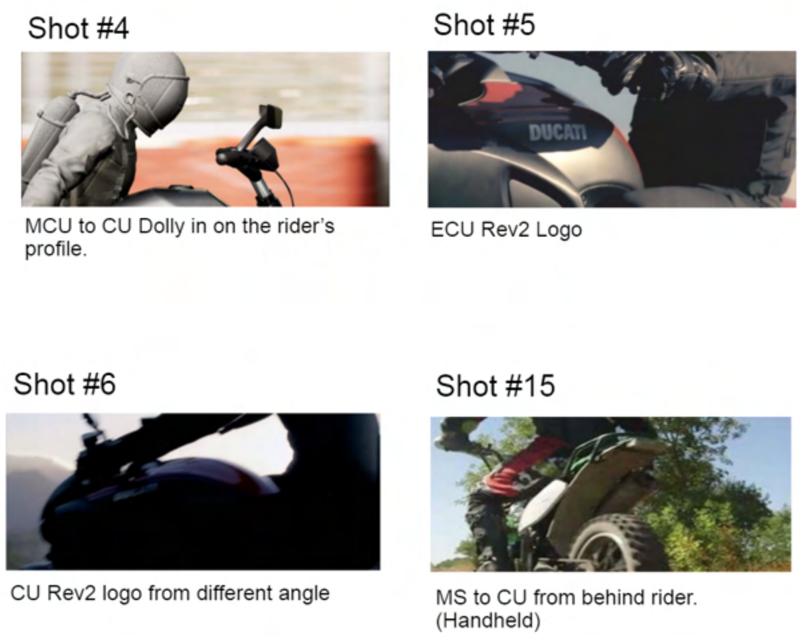


Figure 29: Setup 1 shots prevized in Unreal.

- **Stage Setup 2**

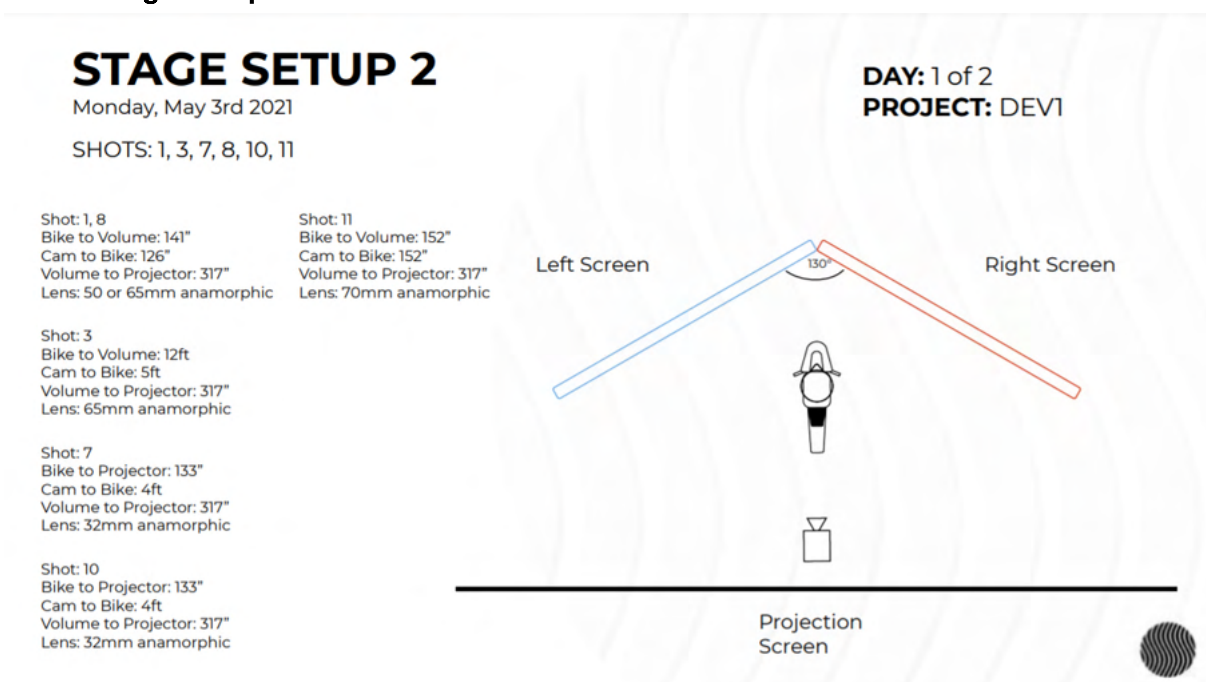


Figure 30: Overhead view of stage setup 2, shooting against the LED walls at 130 degrees.

The majority of our shots were planned to have the LED walls at a 130 degree angle between them. The reason for this was because we wanted to shoot dynamic closeups that incorporated some sort of camera movement like a dolly or push in. The slight angle between the walls allowed us to shoot closeups in a way that we would never be shooting at an angle over 60 degrees from the perpendicular. This limits the effect of luminance fall off as discussed in our earlier experiments.

Shot #1

MCU from behind Rev1. We dolly right following the rider as he mounts the bike

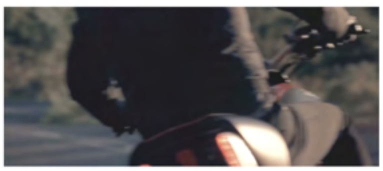
Shot #3

MWS we dolly around the talent as they ride the Rev2. (Slider)

Shot #7

CU on front of bike. From the rider's POV we see a time-lapse like zoom through

Shot #8



Pan across back of bike as the rider takes a turn. (Slight Dutch)

Shot #10

ECU dolly out from behind the bike as the road rips past us

Shot #11

MCU handheld shots of details of the Rev2

Figure 31: Setup 2 shots prevized in Unreal.

- **Stage Setup 3**

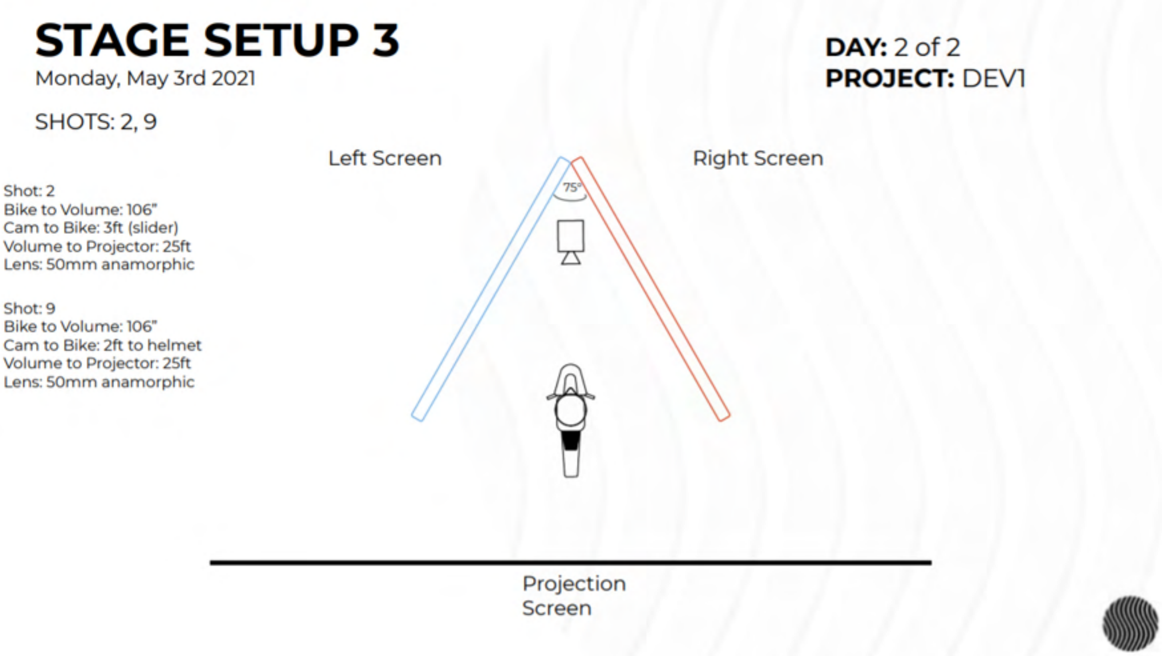


Figure 32: Overhead view of stage setup 3, shooting against LED walls at 75 degrees.

We planned for two of our shots to rely completely on the projection screen for the main background. This was because we wanted to utilize both the LED walls for environment lighting from behind the camera.

Figure 33: Setup 3 shots prevized in Unreal.

Specifically on shot 9, we wanted the reflection of the environment on the biker's visor. To do this we took advantage of our two displays to mimic a curved screen on the LED walls that would give the aesthetic we were looking for and wrap around the front of the helmet.

- **Stage Setup 4**

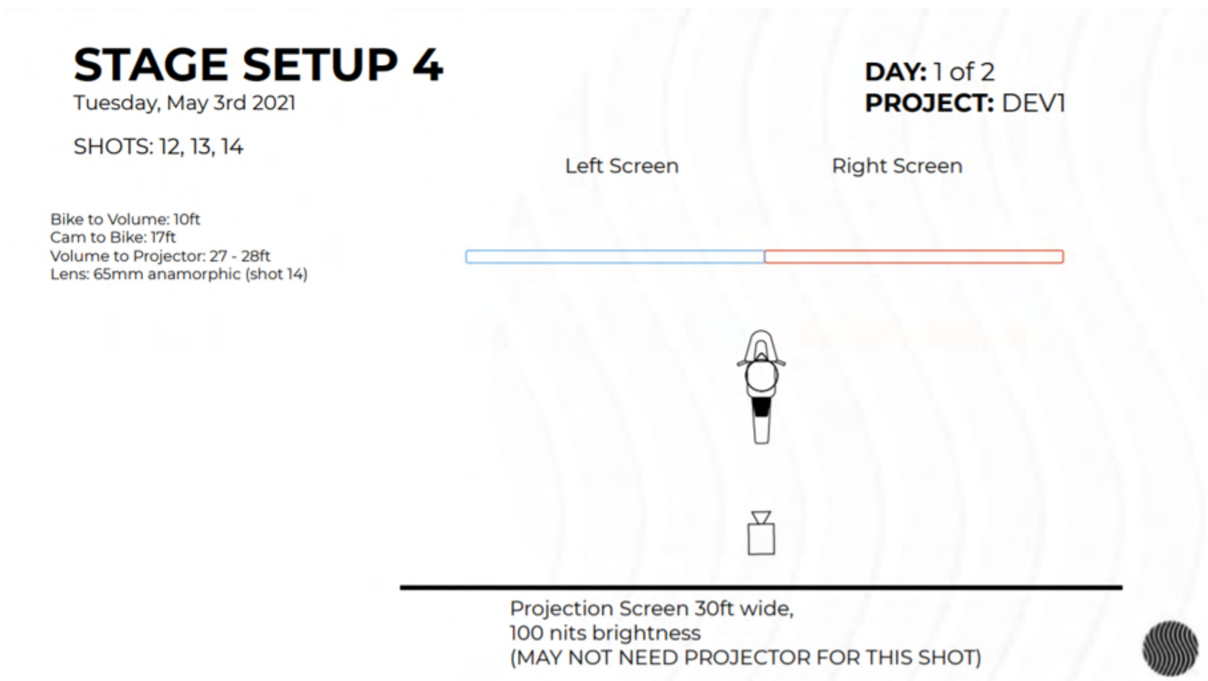


Figure 34: Overhead view of stage setup 4, shooting against flat, adjacent LED walls.

For our last setup, we shot with the walls adjacent to each other. This was to take advantage of their full width in setting up a wide angle shot. Specifically, for shot 14, we had the prototype DEV1 bike displayed as a graphic on the LED walls while two other real bikes surrounded it on either side. The projector here would serve as additional lighting , if needed, but we could achieve the same effect using studio lights since the background was so simple.

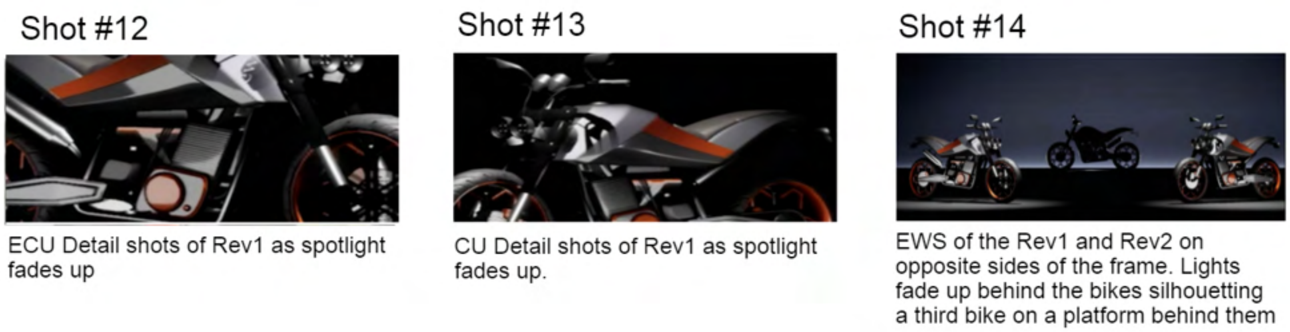


Figure 35: Setup 4 shots prevized in Unreal.

Lighting Tests and Blocking

For on-set lighting we used 4 Arri L-7Cs around the volume as well as 2 large Arri S-120C skypanels closer to the center of the set. The Arri skypanels were chosen due to their ability to be controlled remotely with a DMX board as well as their diffuse, uniform lighting. The lights were all set to a 5500K color temperature. Additionally, we had a large white diffusion silk on the overhead lights to soften them as much as possible.

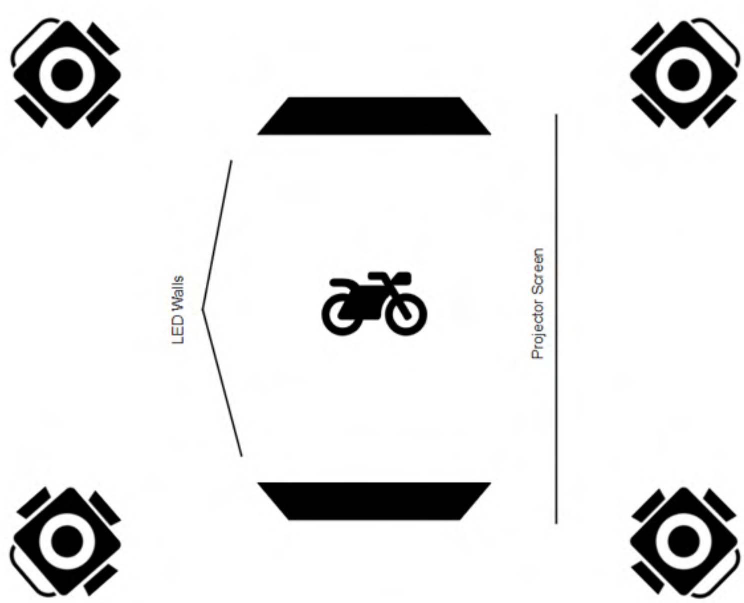


Figure 36: Overhead lighting layout on set.

Figure 37: Crew helps mount the diffusion silk for the Arri S-120Cs.

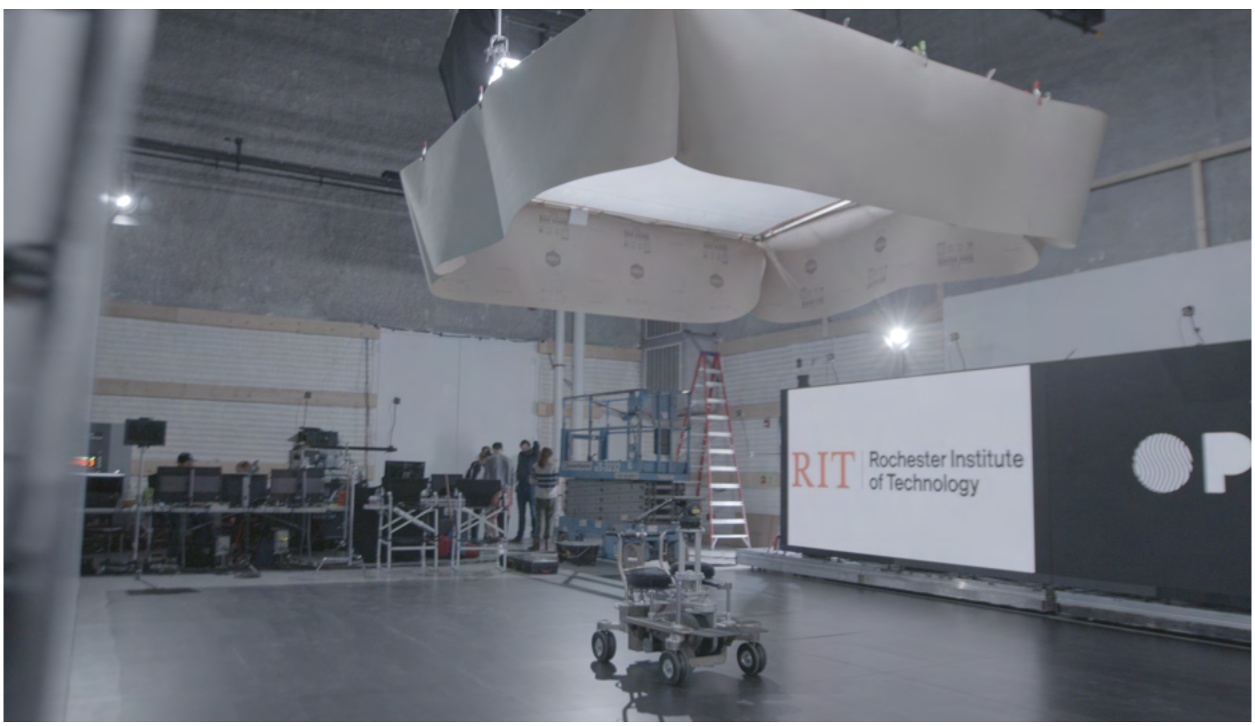


Figure 38: Final lighting setup with additional curtains to direct the lighting onto the center of the set.

We used a GrandMA2 Lite DMX board in order to be able to remotely control the lighting as well as group different lights together to easily switch between settings.

Figure 39: Crew adjusts the settings on the GrandMA DMX board.

As an example of how scenes differed as much as they do on a regular set, we top lit with a softbox for the intro race track scene to match the misty effect of the gray day, but used harsh light for our canyon scene because it was golden hour. The biggest difference is using the walls for practical light effects, but even then we found ourselves mixing the dynamic effect of the walls with moving our film lights themselves during the shots. On this set, both our film lights and walls were LED, fully controllable, real time adjustable, and linkable, making for a streamlined experience.

Additionally, we were able to play with the virtual lighting in the virtual sets. For example, we could control the exposure on each individual texture and object. In terms of post-production, this means that we would eliminate the need for complex custom color grading for specific objects in the background. This is also a great advantage for complete creative control for the DP and director to create their virtual world exactly how they want it to look while on set.

Collaboration between artists and engineers is crucial for virtual production. From lensing and lighting choices to match environments, both physical and digital must go hand in hand to sell the effect. Again, this adds more time for prepping in terms of creating a lighting chart with placement for the walls and exact measurements. In the end, the goal is to get as close to the final product as possible before post-production. Virtual production is inherently reliant on solid pre-production, pulling a lot of responsibility away from traditional post.

Virtual Art Department

Sets

A goal of this production was to utilize a variety of virtual sets with their own unique challenges. Even if we planned for our final product to be no more than a minute long, we wanted to ensure a high-energy, dynamic feel to the piece. We used 4 virtual environments in total in our production.

- **Racetrack**

Figure 40: In-game view of the racetrack set in Unreal.

The racetrack setting provided the ideal scenario for a virtual production. The environment was hazy and lit by soft ambient light with no harsh shadows. We planned for slow and smooth camera movements such as dolly shots to best complement the look. This is a style that most other virtual productions stick to because adding motion blur in post is not as crucial with slow, limited movement. However, we wanted to elevate from this and incorporated an animated stop light shot that brought some dynamism in.

- **Pikes Peak**

Figure 41: In-game view of the Pikes Peak set in Unreal.

This set was primarily designed to test the boundaries of harsh lighting at sunset as well as to see if we could achieve the foreground vs background selling point. One of the advantages of virtual production is being able to control almost everything about your environment, so for us it meant we could shoot at sunset for the entire day. We could also reposition the sun wherever we wanted. This made it significantly easier to truly strive for perfection, which is why we decided to tackle what would be extremely difficult in a real location. Additionally, unlike a real set, there were no foreground elements like rocks we could place in front of the bike since the LED wall is a flat background and was representing motion for many of these shots. This meant we would have to plan each shot differently even if the location was the same. It was another challenge we wanted to tackle and see if we could still make it look realistic even if the environmental elements were restricted to being in the background at all times.

- **Forest**

Figure 42: In-game view of the forest set in Unreal.

This environment incorporated the toughest camera moves of the whole production as well as darkly lit shadows, making it a challenge to shoot. The background would be dynamic meaning that it would be animated to appear like the bike was zooming past at high speed. It gave us an additional lighting challenge to think about, in addition to figuring out a way to incorporate realistic motion blur.

- **Stage**

Figure 43: In-game view of the stage set in Unreal.

Our last world was simply a black backdrop with a CG version of the DEV1 dirt bike being promoted in the piece. The main purpose of this scene was to try to blend the real and virtual elements seamlessly as two real bikes would be positioned on either side of the virtual one on screen. Our goal was to blur the lines between real and virtual.

DP: *“There were a few key elements for this production that I think were really important. And one was versatility. We wanted multiple environments because we wanted to not just show one stock environment. The two pinnacles of this production were multiple environments that showed different lighting scenarios.*

Talent and Props

The existing REV1 and REV2 bikes, plus an additional dirt bike were provided by RIT’s Electric Vehicle Team. These were used as props for one of the last shots as well as the prop bikes for the rider. The rider, Matt Hotaling, is a design lead for EVT. His experience in riding helped enhance his performance as the bike would need to be still for the production. Some additional effects used to make the shots look realistic were a leaf blower and whole wheat flour to act as wind and dirt as well as crew shaking the bike out of frame to add a little jitter to the bike as one would expect on a real track. A custom stage was also built for the bike so it was elevated above the ground. This would make it possible for the wheels to turn while the bike remained still. It helped the talent’s performance and enhanced the realism of his movements.

Figure 44: Prop bike provided by EVT.

Figure 45: Prop dirt bike used as a stand-in for upcoming DEV1.

Figure 46: Close up of the REV2 prop bike and rider gear.

V. PRODUCTION

Final Production Workflow and Diagrams

General Workflow

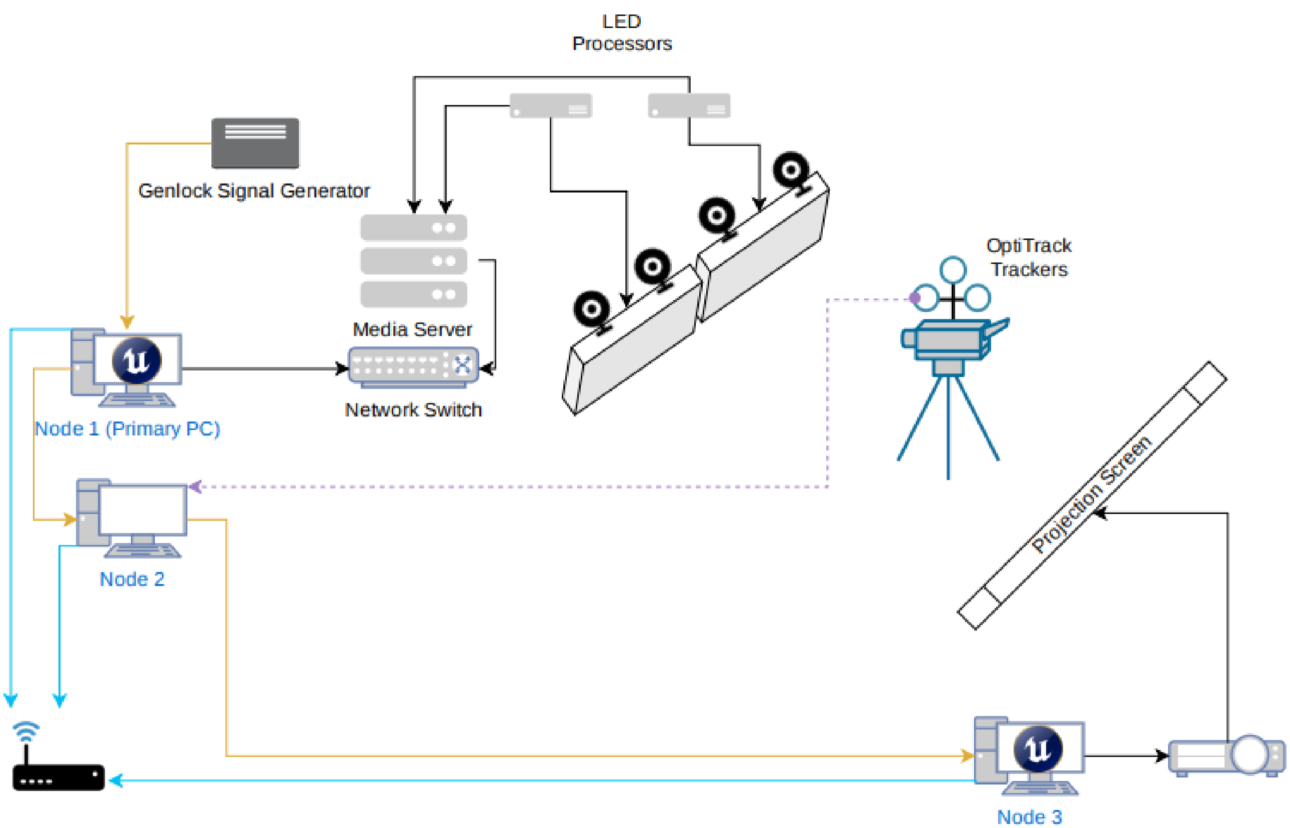


Figure 47: General signal flow diagram of final workflow to render an image from primary PCs to the LED walls and projector. Yellow lines indicate sync between all the PCs while blue indicates internet connections.

Overhead View of Actual Set

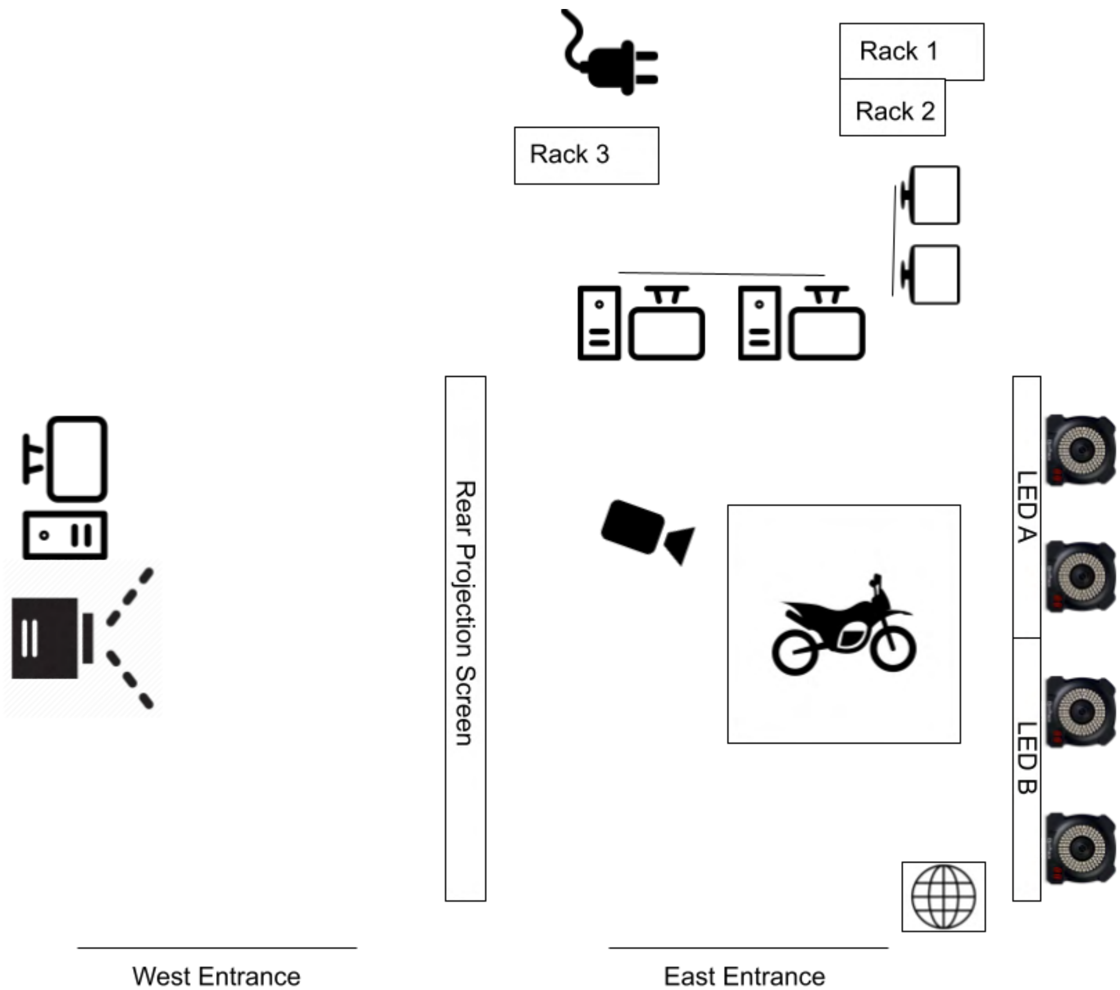


Figure 48: An overhead view of the set layout in the MAGIC Spell Studios soundstage.

Production Stills

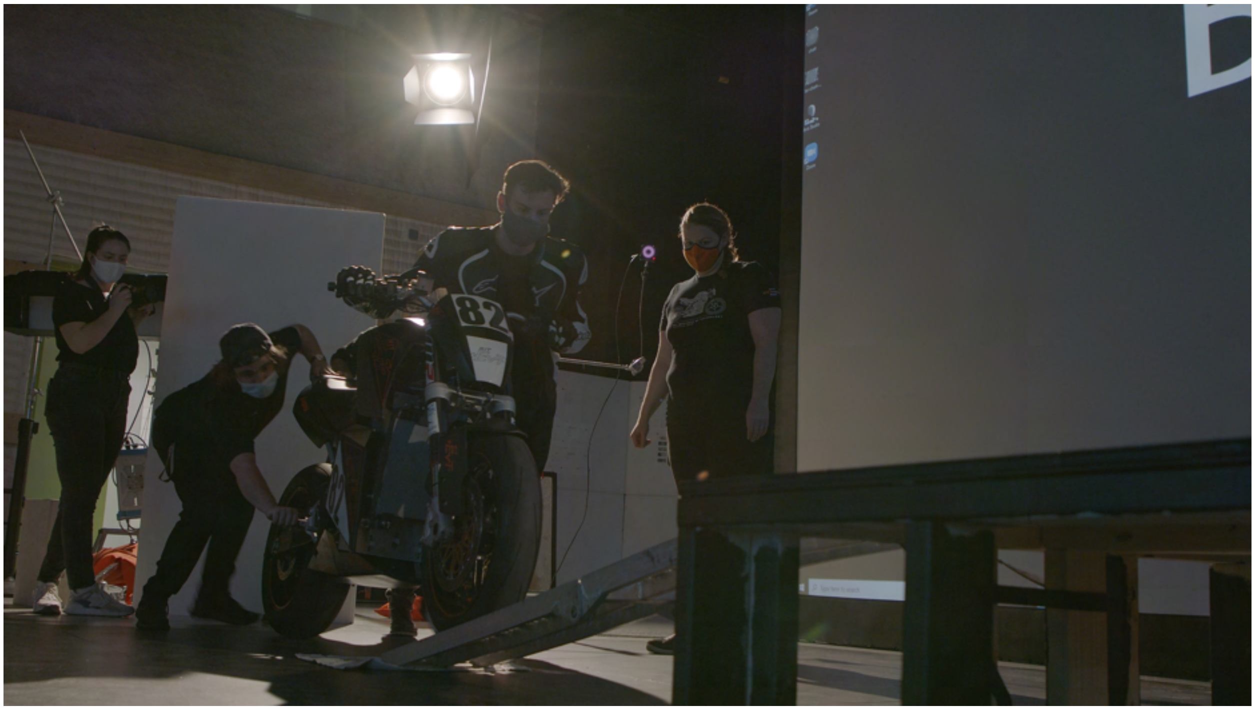


Figure 49: Crew and talent guiding the bike onto the elevated stage.

Figure 50: Dynamic shot on Pikes Peak. Crew uses an Arri L-7C and small flashlight to produce lens flare on camera.

Figure 51: Reflections from the LED wall on a helmet.

Figure 52: In-camera view of the racetrack virtual set displayed on the LED walls.

Figure 53: Crew throws whole grain flour onto the wheels of the bike to act as dirt as the DP gets a close up of the scene.

Figure 54: RIT student adjusts the camera to get a shot of the helmet with reflections of the environment produced by the LED walls.

Figure 55: Video village on set showing a shot of the POV of the rider on the bike against the rear projection screen.

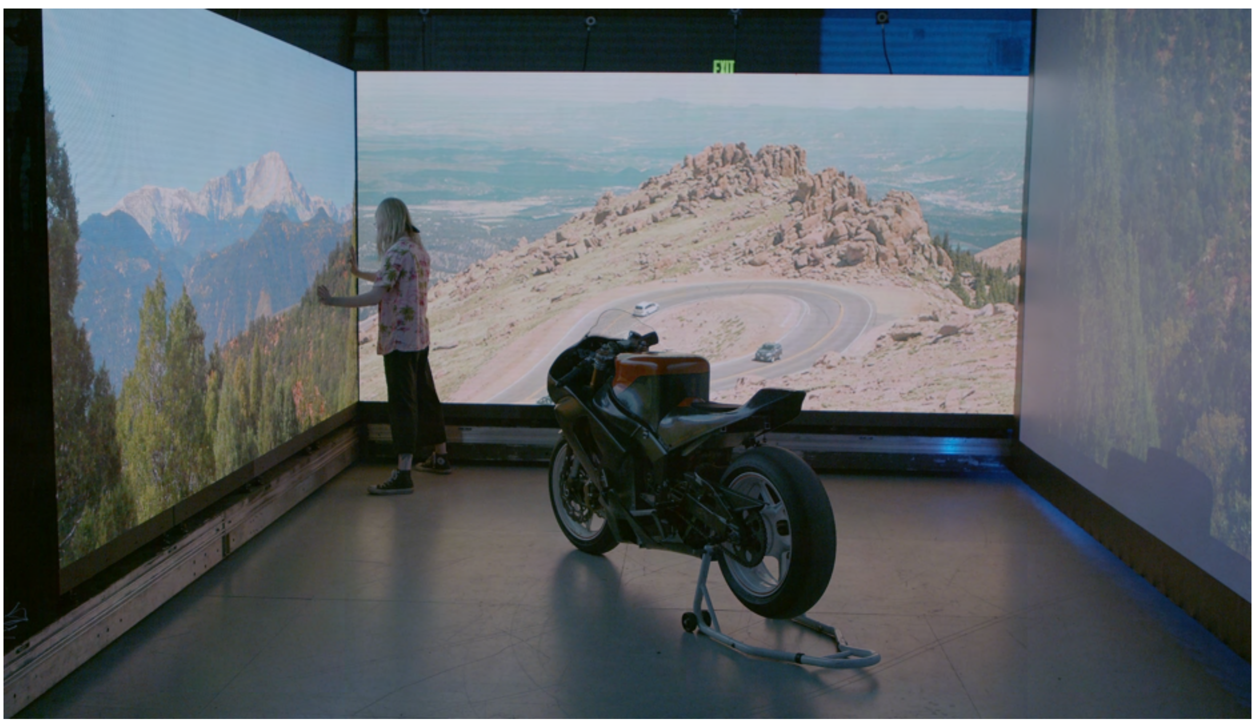


Figure 56: VP Supervisor adjusting the position of the LED walls to 90 degree angles (stage setup 1).

Figure 57: DP capturing a dynamic handheld shot on a dolly with the help of RIT students.

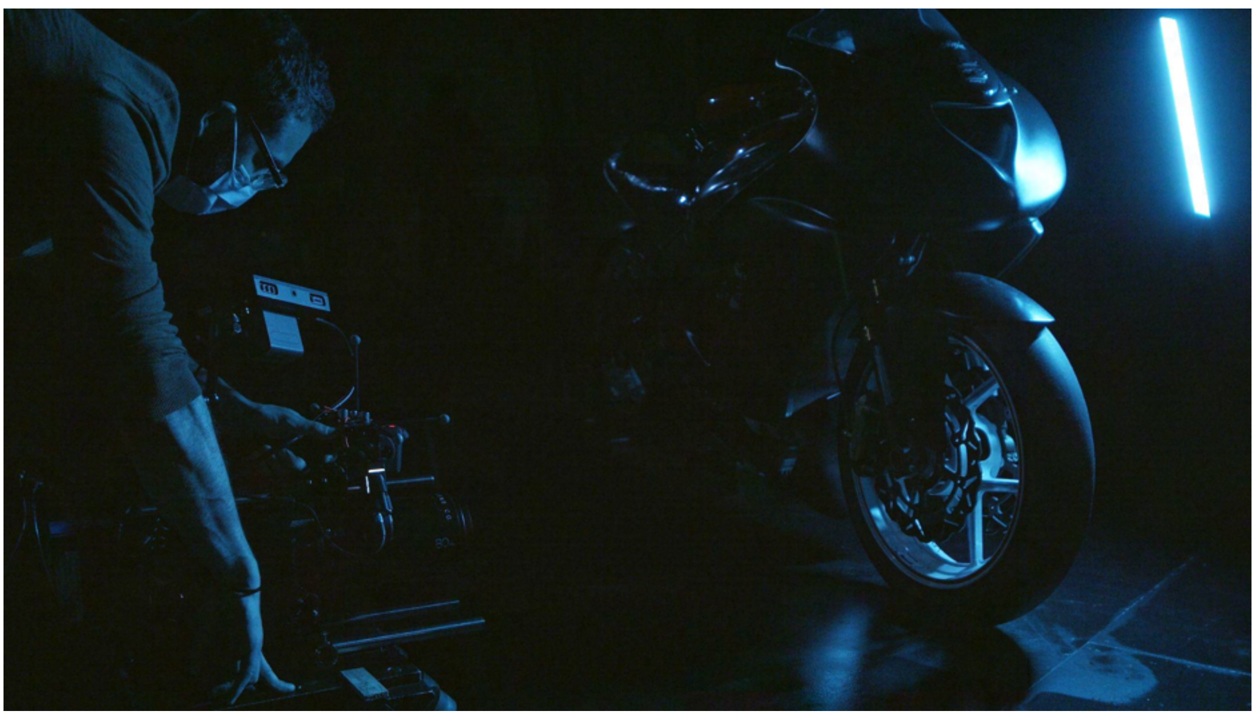


Figure 58: DP shooting an extreme closeup of the EVT logo on the bike using blue LED lights.

Figure 59: Full LED wall setup with OptiTrack cameras attached to the top.

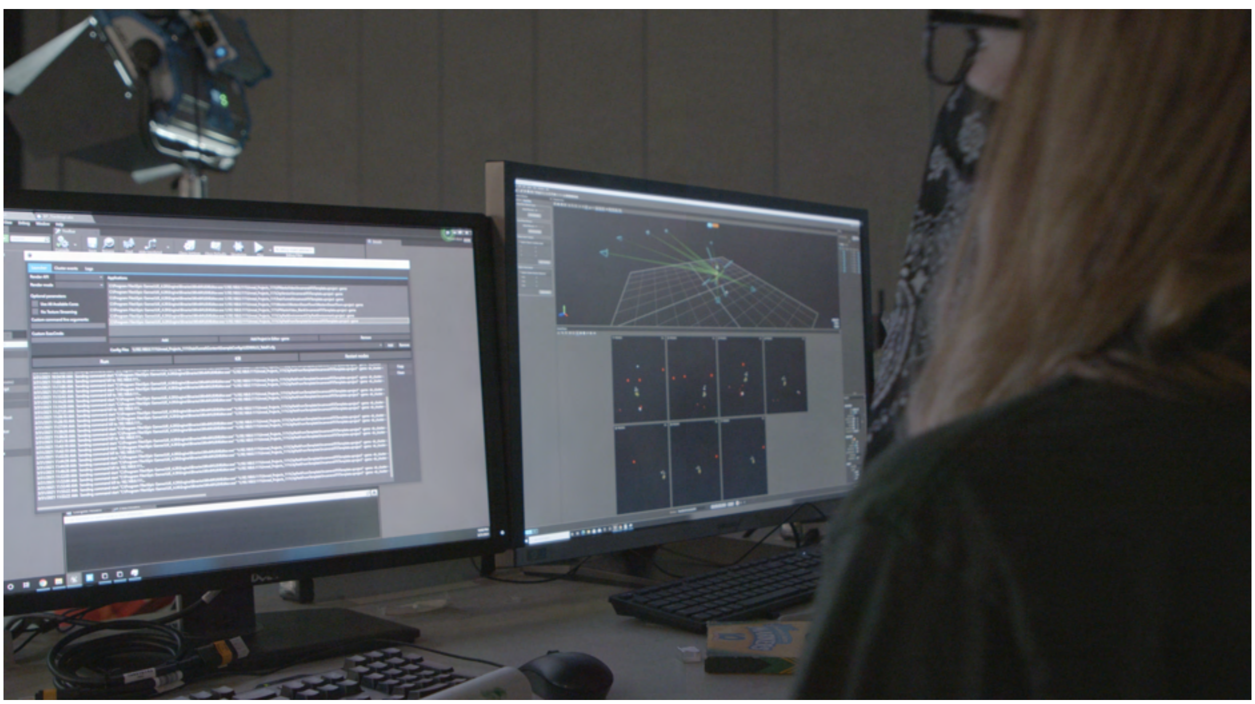


Figure 60: OptiTrack software, Motive, streaming data from a rigid body into Unreal Engine.

Figure 61: Workstation setup with a PC driving the virtual set in Unreal (left), and another streaming the tracking data from Motive (right).

Figure 62: Spline animation in Unreal used to create dynamic backgrounds in real-time.

Production Insights by Department

Director, Director of Photography

One of the biggest differences from our original plan was shooting against the projector for more scenes. Our LED walls were relatively small and had less resolution. While they provided good reflections and flexibility in terms of configuring them to any angle we liked, shooting against the projector also allowed us to capture lower camera angles. Even so, the colors of the projected image looked less contrasty than the LED wall. The projector needs light to create the image and the screen reflects that light with the intrusion of some optical flare, making the black point brighter. The LED technology can emit its own light and is brighter than the projector, meaning that it has a wider dynamic range from darkest black to brightest white. The give and take was interesting between these two display technologies, but it did leave some work for after the shoot since they would have to be matched in post. We compensated for some of the difference between the two displays with minor adjustments on set such as adding purposeful lens flare.

DP: *"We built a little stage for the bike so we could move the height of it to be able to spin the wheels. The other element we used a lot of was flares to our advantage since we mixed LED and rear projection. Projection washes out a bit so we made sure that if the background was going to wash out that we were going to wash out with it, so that way we could match it later. There is no black point that's anywhere near so we flared everything so you couldn't tell that the background was washed out."*

Director: *"We're continuously learning and finding ways to go around technical issues. I think I'm going on today with a little bit more patience. There is a lot of stuff behind the scenes that I don't understand so I just need to sit and wait and let them do their thing and support the team as best as I can. For me it's figuring out creative ways to make it look real."*

VP Supervisor, Co Director

While we tested our workflow and equipment in pre-production, there were still many unforeseen issues that occurred in production that required immediate creative attention. One of those main issues was the result of a workaround we had to implement, dealing, specifically, with the animation of the virtual set around a curve in the simulated roadway.

VP Supervisor: *"So the curve isn't working because we had to entirely change the way we were moving the stage. So initially our thought process was we're gonna leave our landscape where it is and move the camera around the virtual world. Until we found out that we wound up with a lot of jitter in the OptiTrack system as we moved the origin around the space. So our solution was to leave the stage where it is in virtual space and move the entire world around it. When that happened we couldn't just put a spline into Unreal and just move the stage around the spline, we had to make some blueprints where we took a spline and translated the world in the x and y axis. We have a blueprint that understands how high the world is above the ground and uses that as it moves. So as we hit more of an uphill slope, it will say 'I'm at 4 feet, I should* *be at 5 feet; and it moves the world to be at 5 feet. So we have the world moving at all 3 axes to move around the curve. It was kind of taking stuff we were already learning and lining them up in a new way to achieve a different goal."*

This was only one example of the types of issues we had to resolve during production. Essentially, even though we had conceptualized a workaround in pre-production and techviz, the configuration caused other problems in production that were harder to compensate for.

Another trend we observed in our work was the distinction in urgency between pre-production problem solving and production problem solving. While we were testing equipment, we weren't on a set schedule with talent and a director so there was no immediate pressure to implement changes on something that previously worked. However, when we were on the active set, we had to quickly assemble new blueprints in Unreal that allowed us to perform a function with a certain key or click of a button. One example was making the lights turn on in the racetrack world. Since the shot was a dolly around the talent, the timing of the lights going off had to be given on cue as they appeared on frame. In a traditional production this would easily be done with a DMX board or a similar solution, however since we were working with a virtual asset, we had to apply an "on-set" mentality to any interactions we had with the virtual environment.

VI. POST PRODUCTION

Post Production Insights by Department

Editing

With a virtual production, we already expected that the post production timeline was going to be significantly reduced. While there would usually be tons of advantages in capturing RAW in these scenarios, we were testing the thesis of speed, so we were editing footage captured in Prores 4444. The ideal was drag and drop, basic editing needs with minimal processing workflow. That assumption proved correct in our case as the edit was completed in a matter of days.

One other thesis proved correct in the process. When conceived, we were always trying to push the edge of extreme camera moves with proper parallaxing, taking the stance that we could push the bounds of camera moves and response more farther than typical demos have showed, and that any move that seemed just off could be covered up by fast paced editing. This worked in several cases where a faster cut sold what would have been a weaker parallax shot. In fact, in some cases, the parallax properties being slightly off added to the action of the piece.

Due to the streamlined and normal editing workflow of the footage, moving to color and vfx cleanup took a matter of hours, with no special conform needs. In total, color and VFX touch-ups added just a couple additional days after the cut was complete.

VFX - Motion Blur

Our main issue with using a virtual set run by a game engine was missing the motion blur effect. We perceive motion blur as a function of shutter speed and the length of time the camera iris is open and exposing the light from real objects in real motion in traditional photography. The camera shutter can't capture infinite frames per second in real life, so when objects move quickly during the finite shutter duration, the movement gets recorded as a blur. However, the game engine renders intermittent stills and so when we display a dynamic render of the virtual set on the screen, the effect is absent. Additionally, we could not fake motion blur with a basic toolset in-camera since Unreal doesn't currently support real-time rendering of motion blur effects due to computational expense. This resulted in the background looking 'choppy' on camera while it moved, which was an issue we needed to resolve in post.

One note to clarify here is that this was only visible when trying to fake backgrounds zooming by camera without ever moving the camera. This would be an argument to still use pre rendered backgrounds for those types of process shots, assuming parallaxing isn't needed. When doing a motorcycle or car scene, motion blur is absolutely key to making the footage look truly real.

DP: *"It's why in a lot of demos there's action in the foreground because the background is always static. A lot of VPs haven't tried that because they are scared of motion blur. You can tell if something is being blurred by the DOF of camera vs motion blur because they are two different effects, one symmetrical and one directional."*

Color Grading

In terms of color grading, we wanted to bring in a trusted colorist that could correct two main things: the exposure on some mountains in the Pikes Peak shots, and for every shot to match each other. Matching the contrast of reflection to background. Matching differences between projector and led wall.

Upon first pass, our colorist couldn't believe how normal the process was, easier even in some cases since it was a single layer timeline(as opposed to many VFX timelines with stacked layers). The one challenge was the mix of LED and projection backgrounds, but because of the intentional flaring, it was often unclear which was which. We had to go with a look that didn't put too much false contrast back in because it would start to give away which shots were which. Another reason to keep contrast a bit lower was the resolution of the LED screens. Due to their lower overall resolution, certain elements of the backgrounds, especially the race track, lacked detail and rather resembled a ton of artifacts. By keeping contrast lower, it helped blend some of those elements, looking like distance background dots.

VII. CONCLUSIONS

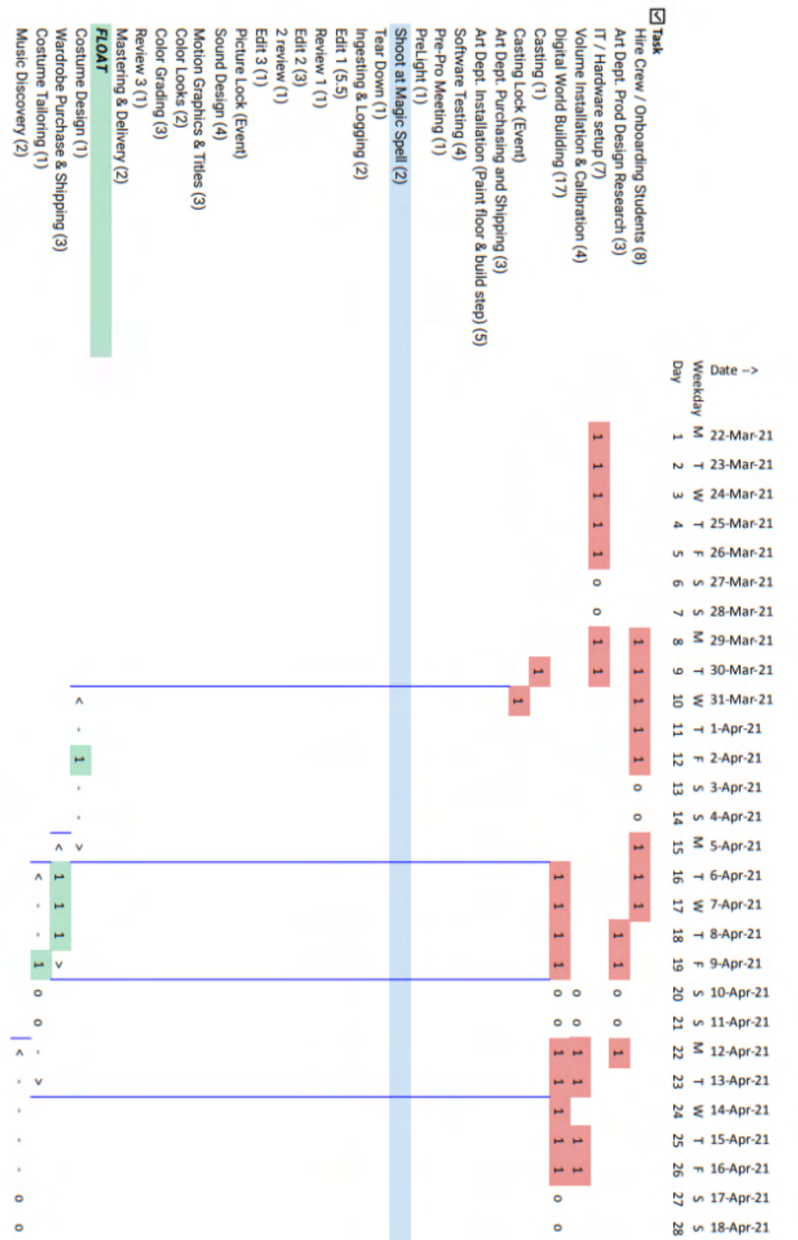
Upon exploring virtual production technology and engineering our own workflow, we were able to further understand and first-hand experience the benefits this type of production brings to each step in the pipeline; including insightful previsualization of shots, real-time rendering of imagery on set, and reduced post-work. We were also able to experiment and assess color management and aliasing dilemmas and bring to light several considerations to be made when calibrating an LED volume to behave as a scene rather than a standard display. However, we were unable to establish a complete genock system for our workflow as well as to implement motion blur in a game engine. This is something that we would like to achieve in a future virtual production.

Moving forward, RIT intends to research ways in which to correct the observer metamerism effect . Optic sky hopes to blur the line between real and virtual by pushing the boundaries with the next generation of technology in their future virtual productions.

DP: *"I still see virtual production as a toolset. I see it as a tool that's used when we need it. That being said, I think that it is going to get bigger because it's going to replace a lot of things that we deal with right now like greenscreen. The way I look at it is you can either spend 4 weeks in post fixing your shots or you can have that in prepro with the director, and by the time you get to post it's really efficient. I think that type of creative control mixed with that efficiency is why we're trying to build out the commercial pipeline as fast as we are."*

VIII. APPENDIX

Gantt chart



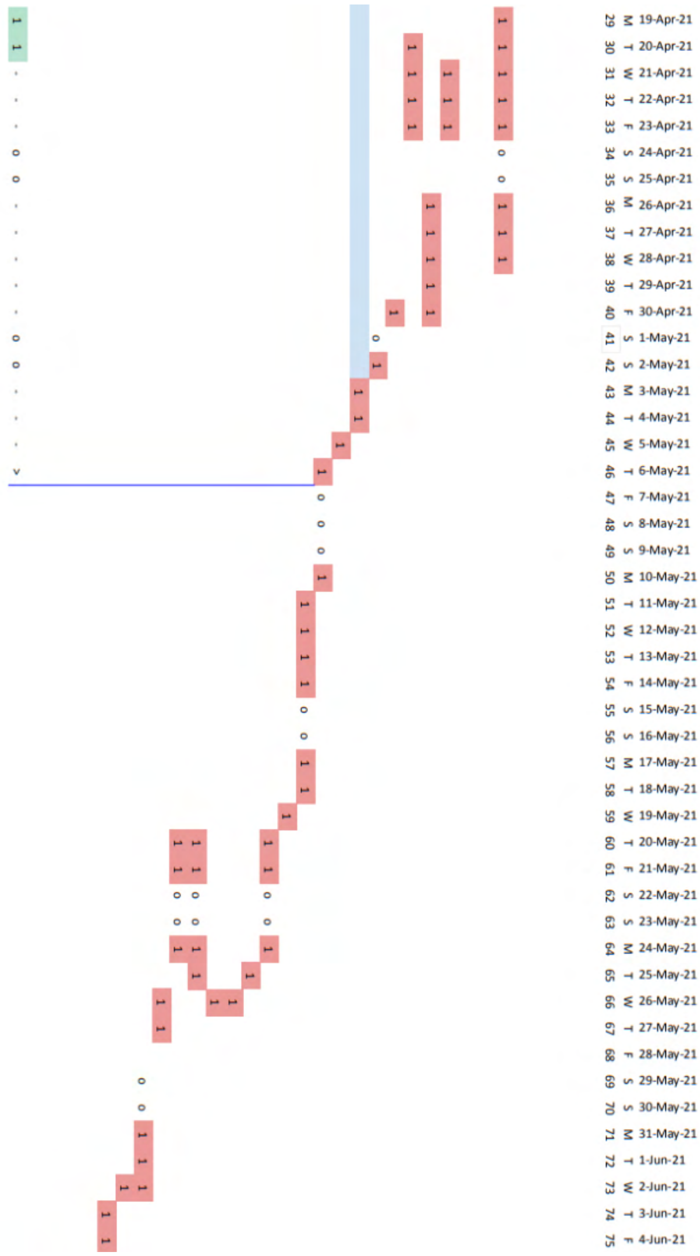


Figure 63: Gantt chart of the full production schedule.

Signal and Routing Diagram

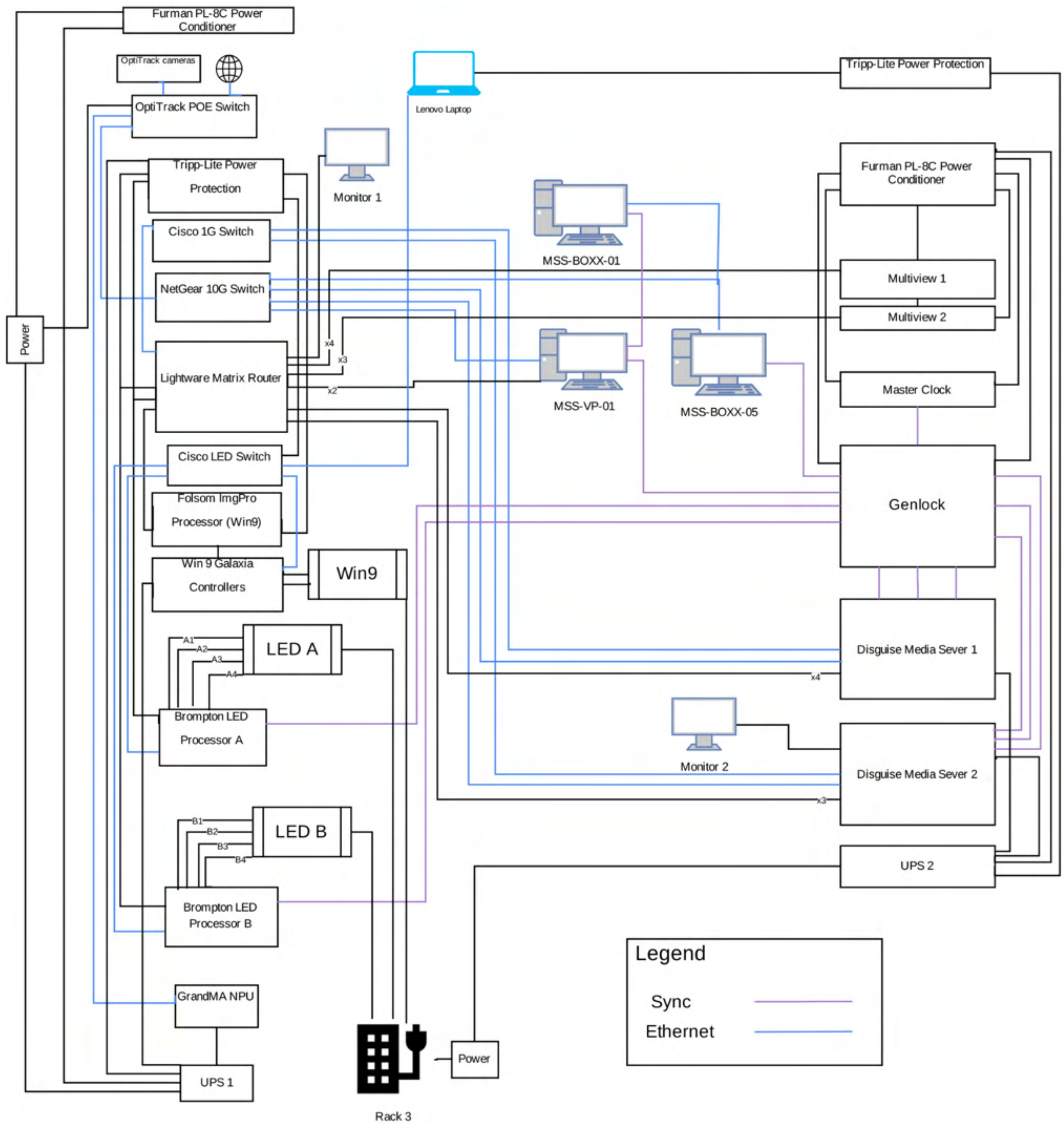


Figure 64. Full signal flow and routing diagram of all data and media management hardware.

Additional Colorimetric Assessments

The following table and graphs support the findings and conclusions made in section IV, "LED Wall Characterization and Calibration". We found that the chromaticity of the white point as well as the green primary had a significant trend as the intensity of the display increased.

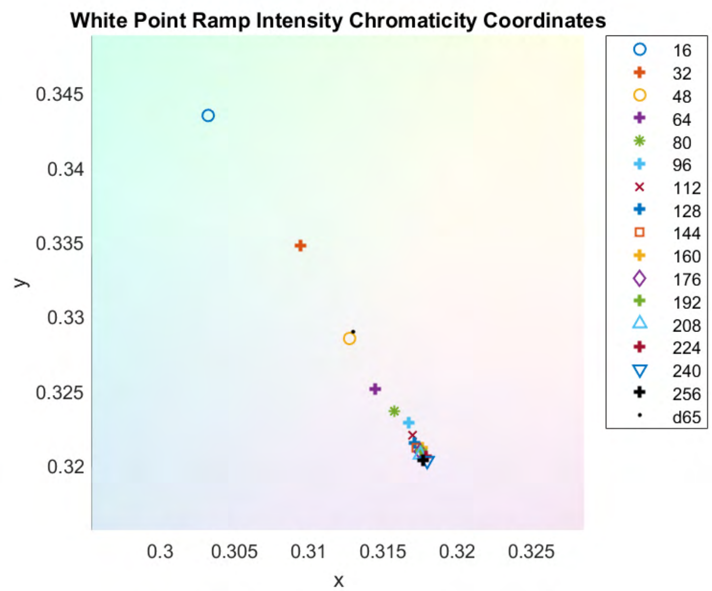


Figure 65: White point chromaticity variation as a function of ramp intensity.

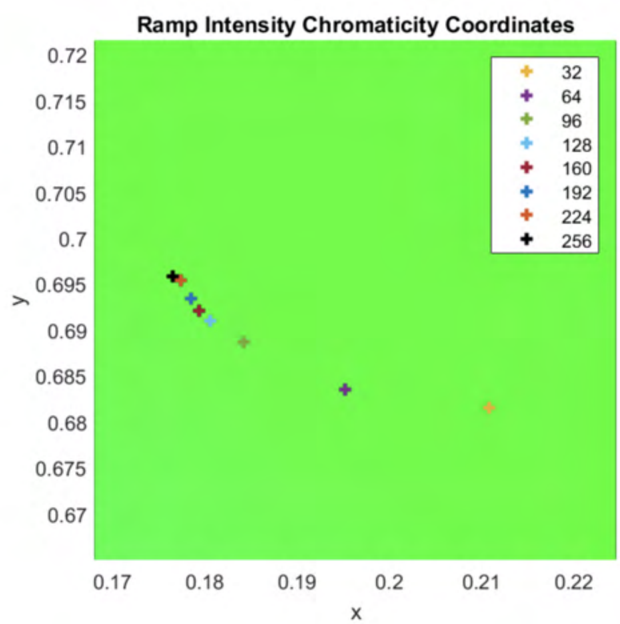


Figure 66: Green primary chromaticity variation as a function of ramp intensity.

From Figure 65, the white point seems to shift in hue towards a cooler correlated color temperature as the display intensity decreases, meaning that it is not as stable with lower intensity values. Similarly, the green primary actually shifts towards a redder hue as the intensity decreases. An unstable green primary could affect the perceived color of an image at different intensities and result in unwanted color differences.

Table 2 demonstrates the colorimetric variability of the white point and primary chromaticities for different locations on the LED wall relative to the center. These correlate to the differences in luminance as discussed on section IV.

	Colorimetric variability between different locations on LED wall							
Location	top left	top middle	top right	middle left	middle right	bottom left	bottom middle	bottom right
deltaE from center	3.601 0.793 0.951 1.383	3.464 0.409 0.526 0.866	4.522 0.244 1.345 1.942	1.222 1.271 0.985 0.639	2.606 0.519 0.910 0.940	2.931 2.037 2.053 0.755	1.871 1.252 1.063 0.184	2.734 1.421 2.201 0.975

Table 2: DeltaE perceptual color difference between various locations on the LED wall relative to the center.