

vi

- SRA

SRA

frame by to problem to solve, how its solved, before and after metrics

STAR: Situation Task Action Results

SRA

SRA is a real-time digital human workforce solution designed to support businesses and customers anytime, anywhere. We iterated through multiple concepts before we achieved our launched product.

- **Reported directly to Sr. VP of Research who lead the Generative AI Digital Human Platform**
 - Building Design Team from ground up as Hiring Manager to include Directors, CG Artists, Product Designers, Concept Artists, Technical Artist, SWE, and Digital Asset Manager
- Problem: ML model is hard to evaluate, how to QC data going in to machine learning
 - What can be automated and what cannot be?
 - Can we assess data without looking at the images/video?
 - If we have to look at it, can we pinpoint the problem immediately.
 - Since our ML data is digital humans and shot doesn't change, can use YUV diff graph which compares adjacent frames.
 - Don't build EVERYTHING from scratch. Do research on third party stuff first. Learn from open source
 - QC tools created for libraries
 - if cannot be automated right away, can we create to view
 - key point outliers by collapsing keypoints
- Building the automated Quality Control and Media Asset Management pipeline using Python and FFmpeg for training realtime digital humans via generative AI, to assess crops, transcodes, sound, frame anomalies, rig, key points, sync, color, proxies
 - System runnable by a digital asset manager with CLI experience
 - editable json, yaml file, runnable on Linux and Mac
 - FFmpeg, MediaArenas QCtools, Mediapipe, Sox (sound),
 - code that determines Loudness Units (Lufs) and correct codecs
 - Analyze data as graphs. for instance, YUV diff graph to get bad frames by getting a spike in the graph. Since we know the data is recorded in one long shot.
- PROBLEM: design delivery and Research/SW is not synced with deliverables
 - Futureproofing data by capturing at best quality, but slows down the process immensely
 - Can we compromise on both?

- Converting greenscreen stage from 12k resolution video capture to multicam broadcast capture and mocap, as well as building the end-to-end hardware/software solution for audio capture and DSLR facial photogrammetry within a reasonable budget.
 - Problem: Too much future proofing by capturing too much data with unclear requirements.
 - Solution: work with r and d on narrowing down requirements and building a capture system for production.
 - Maintain flexibility for r and d, but keep the rigid requirements of software team.
 - Increase speed of capture tenfold
 - Problem: How to capture with a midsize budget
 - Understand project requirements
 - Ensure studio is color correct, but when deploying to various devices, its lesser priority
 - Problem: How to sync cameras
 - Dont trust hardware specs, always verify
 - Our recorder claims to be able to shoot synced cameras, but found out via timecode it is off a few frames heads and tails
 - wrote code to take videos with timecode and cuts heads and tails to insure sync
 - Experimented with AI solutions first to see if there results are good
 - Experimented with 3d Cameras but frames drop for long recording sessions
 - Need timecode and genlock because will triangulate data later for inhouse mocap solution
- **On-set Technical Director for Talent data capture, and Pipeline Lead for the Design Team**
 - Previous experience as a filmmaker/improv and onset director helped with coaching actresses to perform better
 - Actors strike affected getting good quality data because we couldnt hire SAG
 - Solved all Technical issues onset
- Collaborating with Business Strategy and Product Designers on pitch and video decks in order to prototype Minimal Viable Products using Maya, Blender, Resolve, Unreal, Unity
 - Presented to C-level people
- Working with R&D scientists on 2D and 3D digital human methods found in latest papers
 - Focused on dataops and capture related portion
 - determine how to emulate public datasets or buy them, since for product licensing prevents open use
 - Renderpeople vs capture inhouse. Use as ground truth
 - Driving an avatar via unreal engine, metahumans
 - dependent on unreal engine platform
 - works with iphone, but not so much about google phones
 - arkit vs mediapipe
 - at the time mediapipe did not do well with anything 3d points
 - arkit was trained on blendshapes
- **Asset Management**

- PROBLEM: data is being passed without going through QC.
 - data needs to be more secure. Log everything
- Lead effort to storing data in Resource Asset Management for more security and less error prone file management
 - Only deploy "published" data
 - proper versioning across entire pipeline
- Project Asset Management solutions like Kitsuru, Ftrack, Shotgrid

ML

- Problem: How to scan 1000 heads for a inhouse ML dataset?
- Building the 3D digital human publish/review pipeline, which processed hundreds of scans for machine learning models using Python, Maya, Nuke, Shotgrid, Deadline, Unreal
 - Created a synthetic data set consisting of 1000 3d heads from scans
 - Instead of vfx scanning, opted for third party to scan with handheld scanner
 - cleaned up by character artists
 - created a publish tool where the model is sent to dailies via shotgrid in a turntable with 3 point lighting
 - Created mocap review from external vendor to assure the data was good
- Problem: VFX recon takes a long time. how to scale up capabilities of renderfarm to be in hybrid/cloud (learned in lytro)
 - used GCP as well and worked with Devops engineer
- Extending capabilities of the DSLR body and facial photogrammetry rigs using Python with MetaShape, PYQT, Maya, OpenCV, Wrap, Deadline, and Flask web framework
 - worked with Scan Technician to improve the system
 - ex. adding external gpus
 - increasing boundary boxes
 - working with scale markers
 - fixing broken cameras, triggers, and grip rigging
 - two stage pipeline.
 - Capture to 3d recon is fully automated
 - Artists review scan neutral mesh after recon, selects points for Wrap and place in frankfurt alignment for it to be wrapped production neutral mesh, then that mesh gets mapped to to FACs scan meshes using optical flow wrapping.
- Problem: infrastructure tech debt, everyone working on custom versions of things. render farm failing constantly
- Creating CI/CD using Jenkins and Ansible, and hybrid renderfarm with Deadline and Google Cloud Platform in order to scale up rendering for volumetric video
 - Worked with DevOps and IT to use our SW Deployment system to reimage inhouse
 - Applied the CD to work with cloud render nodes via Google Cloud Platform.
 - Refactoring cg artist tools to use latest APIs, be multiplatform, and utilize open source through Academy Software Foundation and VFX Reference Platform
 - Looking at ways to scale up our platform, which is mostly license bound
 - looked into open source solutions such as OpenCue, Blender, FFMPEG, Shotgrid, instead of Deadline, Maya, Kitsuru, and Nuke
 - refactored legacy code to follow more modular code

- ex. remove version dependencies on api calls.
- Troubleshoot headset
 - With immersive tech, there is no chatgpt or youtube videos to help you, need to work with engineers that created it directly.
 - digital human group was a factor in increasing FOV vertically instead of horizontally

L

- Problem trying to solve: How to democratize render monitoring outside of using proprietary software?
- Developing tools to monitor hundreds of hybrid render nodes using Python, Deadline, Excel, Slack, and GCP, in order to democratize render wrangling across all teams
 - Joined, then three weeks later google acquisition announced.
 - Render Wranglers contract was all canceled, in middle of production that still needs to be finished
 - Justin Timberlake video was deployed
 - parts of the lightfield version was finished, but never released
 -

SIGG

Lead AR App Engineer | Video Production Director and Correspondant | 2019 Tech Lead for AR/VR/MR
 Lead AR App Engineer | Video Production Director and Correspondant | 2019 Tech Lead for AR/VR/MR

ACM SIGGRAPH ACM SIGGRAPH Jan 2009 - Jan 2020 · 11 yrs Jan 2009 - Jan 2020 · 11 yrs Science and Technology Science and Technology

- Leading a team of volunteer developers and artists to launch SIGGRAPH's Scavenger Augmented Reality Mobile Conference App from 2017 to 2019, using Unity and Vuforia
 - Help recruit volunteers
 - Came up with technical stack
 - Gamesparks for cloud backend to save data (did not exist in first version, users lost data)
 - Determined ar pipe, used Vuforia due to reliability and east of use. Unity MARS (cost money) and AR Foundation was still at early stages

Collaborating with UX designers on implementing features such as ground plane and image targets that spawn 3D creatures in AR, as well as photomode and social share

- Analyzed data on cellphones at gaming conferences to determine features
- applied more traditional ux design patterns for easier navigation
- Mentored students

- Implemented more modularity, such as consistent fonts
- Ensuring compatibility for iOS and Android, which lead to 1000+ attendees playing
- Oversaw 22 AR/VR/MR exhibits in 2019 as tech lead subcommittee member for Immersive Pavilion
- Director and Correspondent for Siggraph Asia Highlight Videos, leading an onsite video team in order to deliver daily recap videos for Facebook and Youtube Channels

PS

Providing technical and render support to CG artists, as well as automating third-party 3D asset conversion from games to VFX commercials, rendered offline with VRAY

- Hard to build a consistent pipeline because third party assets are all different
- wished there was a "USD" back then but it does require building a lot of short term pipeline really fast
 - Experimented with game commercials rendered out of PS4, but post production rendering always wins because can tweak environments better and use better rendering (end goal is to sell, at whatever means necessary). Also less complicated pipeline
 - Challenges, we are told not to "bother" the studio too much so we typically have to get internal tools running ourselves. For instance, for Uncharted, Had to pinpoint a siggraph paper and talk in order to get their "Surf" shaders to work
 - For Spiderman, I had to redo the toon shaders to match for rendering engine
- Researching VFX pipelines and supporting next gen game console transition to VR 360, HDR, and 4k resolution workflows for Cinema4D, Maya, Nuke, VRAY, and After Effects.
 - Went to conventions like DevCon and was the liaison to R&D to ensure the smooth transition from 1080p PS4 era to PSVR and PS4Pro (4k and HD)
- Full stack web development of internal wiki and technical documentation using Drupal, Capistrano, Git, and programmed in Javascript, PHP, Ruby, HTML, and CSS
- Managing and upgrading software, hardware, and network consistency with Sony IT across the renderfarm, servers, and workstations using Deadline Monitor and CLI
- Performing photography, studio lighting, and photo editing for Sony Products to be used for brand packaging, instruction manuals, and promotional marketing
- Volunteering with R&D for hackathons to create PlayStationVR demos with PS4 devkit

Pros and cons working in tech

- tech has surprisingly limited resources

- a lot of everything needs to be proprietary since launching a product with a bunch of third party dependencies cost money
 - opensource over using DCC
- a lot of build up and tear down pipelines when proving something doesn't work
- generalist over being a specialist, which has pros and cons

other light fiend companies

talk about SPIEVR.AR

-leia inc , looking glass, creal, magic leap internal demos

Fix resume to work with ATS (add keywords like work experience, one column resume)

Add multiprocessing python work under SRA

add backend web services for internal web tools