Facial appearance scanning using machine vision

Andy Bastable
Lead Engineer, Rare/Microsoft
@andybastable #GDCEurope

GAME DEVELOPERS CONFERENCE™ EUROPE
CONGRESS-CENTRUM OST KOELNMESSE · COLOGNE, GERMANY
AUGUST 11–13, 2014 · EXPO: AUGUST 11–12, 2014

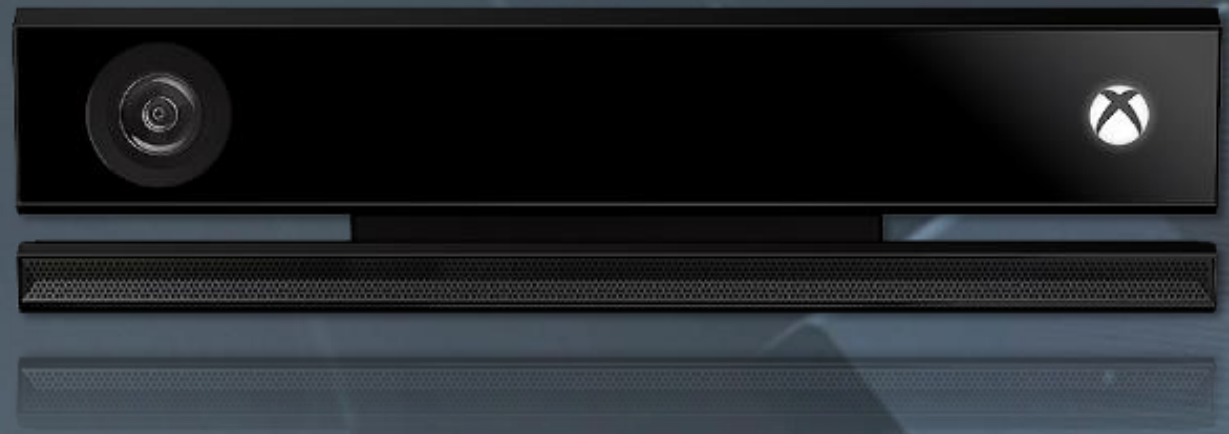
Introduction

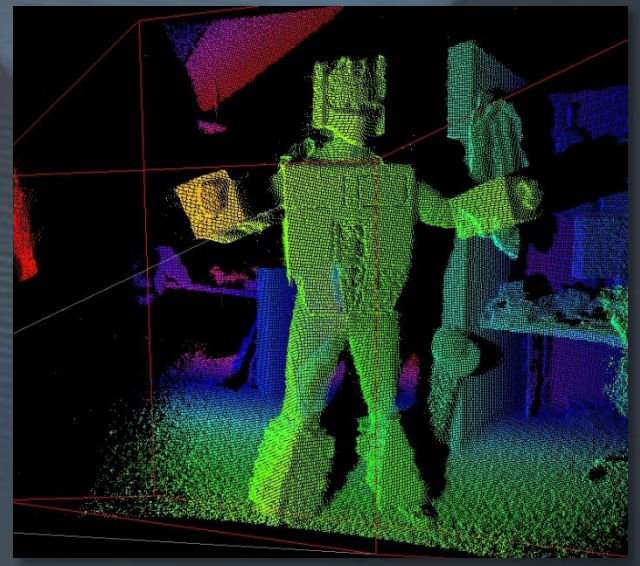
- What this talk **is**
  - An overview of some of the challenges and general principles when using machine learning in games
  - Inspiration about what's possible with Kinect
  - Inspiration about what's possible with machine learning and machine vision
- What this talk **isn't**
  - A step-by-step guide for writing machine vision systems
  - An academic breakdown of algorithms and code

Champion scanning in Kinect Sports Rivals

- What is the Champion Scan?
- Vision: “Awesome You”
  - Had to be **You**
  - Had to be **Awesome**
- Main problems:
  - Would it even work?
  - Making the experience playful yet accurate
  - Knowing that it was going to work for everyone

Primer on Kinect Feeds





How The Scan Works

- User moves into correct positions and performs certain actions
- We scan body & face
- Classifiers determine facial features
- Results used to assemble final character

Tech made up of Machine Vision "Classifiers"

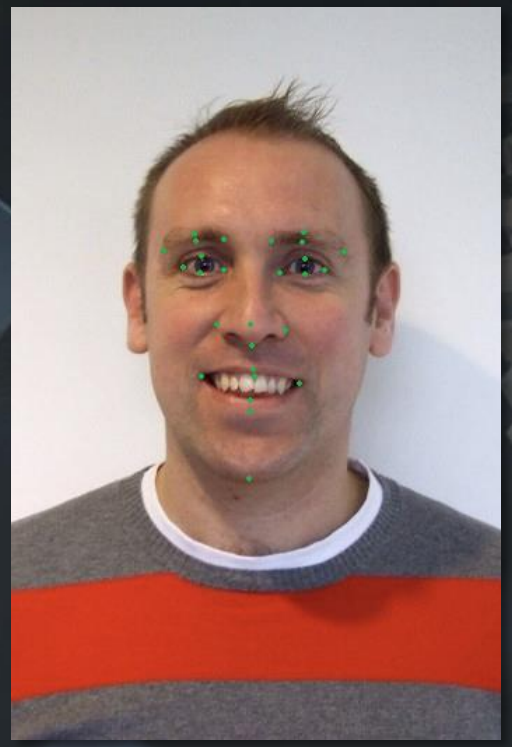
- Face Shape
- Body Size
- Glasses
- Facial Hair
- Skin Tone
- Hair Style

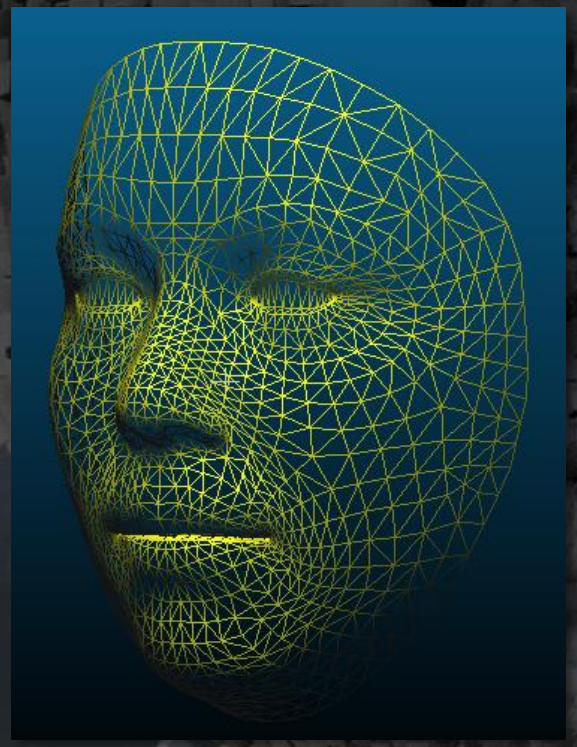
Face Shape

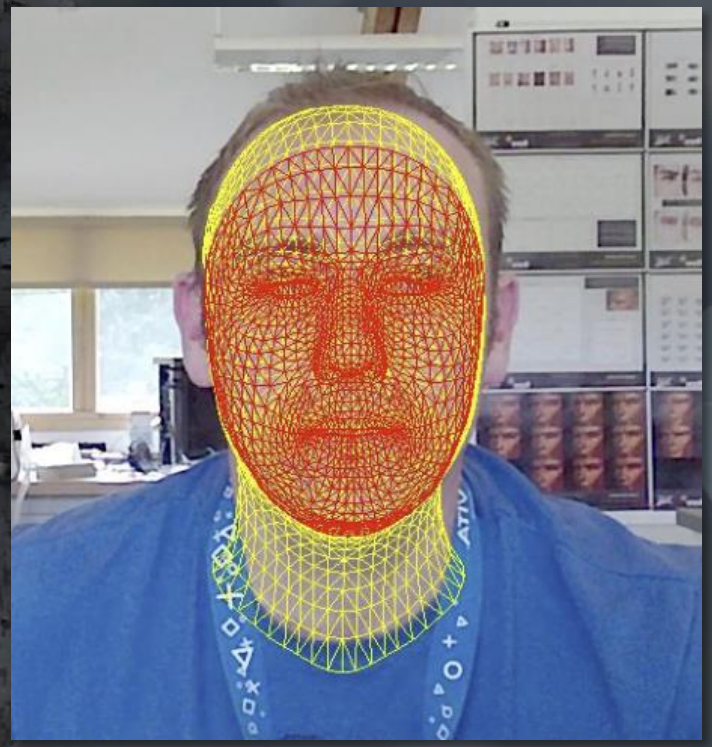
- Developed jointly by Rare and a team from Redmond
- Approach is part of the Xbox One HDFace XDK
  - Available for developers to use
  - Also provides high-quality face animation tracking
  - 10% GPU and 1 Core CPU during shape computation
  - Much less (~12ms) for animation tracking

Face Shape

- First step: register a neutral face mask to the face.

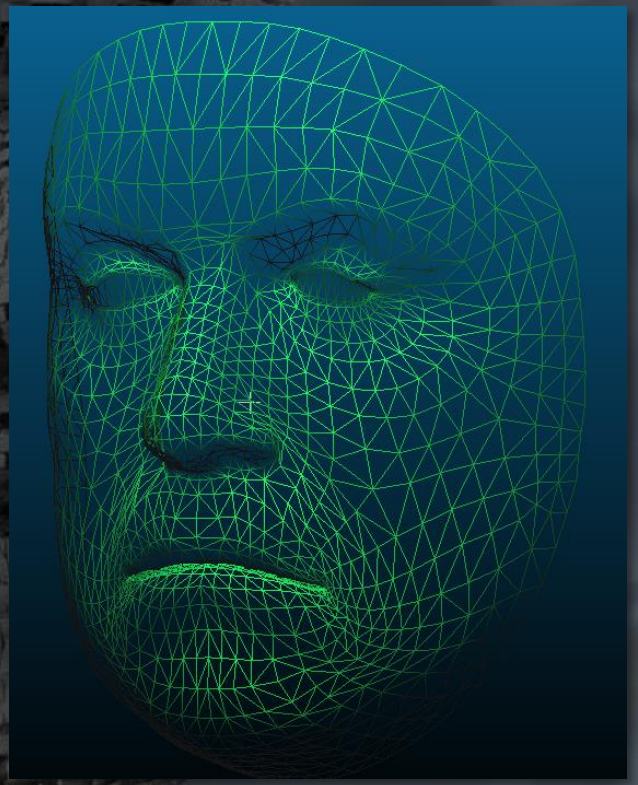






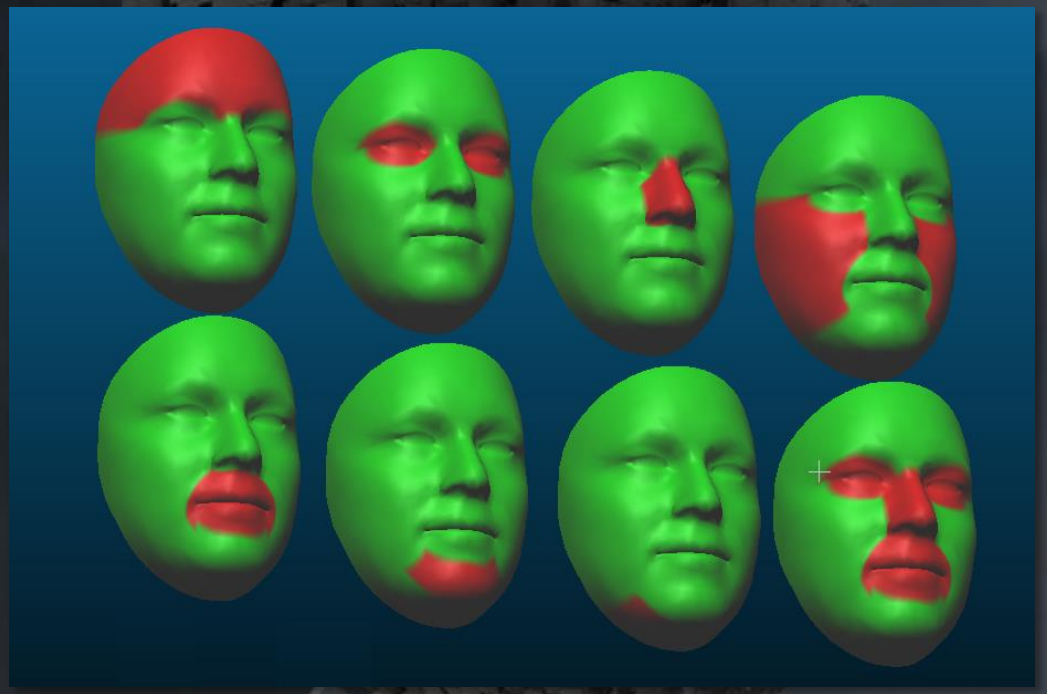
Face Shape

- Then deform mask to “shrink wrap” onto depth feed



Face Shape

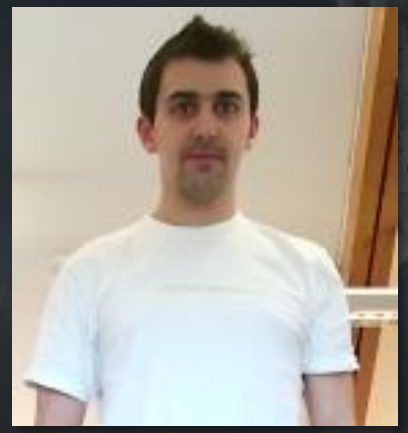
- Then recursive PCA to extract blendshape parameters
  - http://en.wikipedia.org/wiki/Principal_component_analysis



- 93 parameters (~sliders) in total

Face Shape

- Overdrive the parameters to give more characterisation
- Apply parameters to “stylised” head

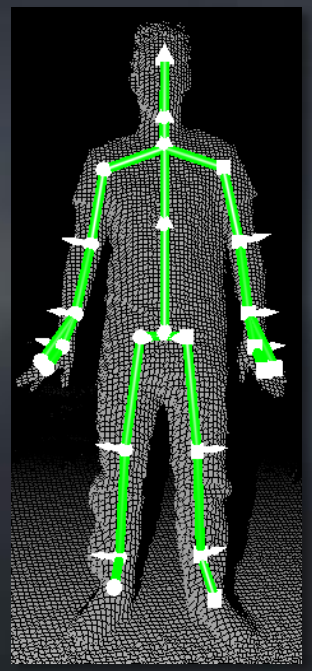


Key Learnings

- Optimise the right thing!
  - Avoid vague goals
  - Test your hypothesis
  - Optimise the correct metrics
- You need lots of data
  - A small amount of data leads to false confidence

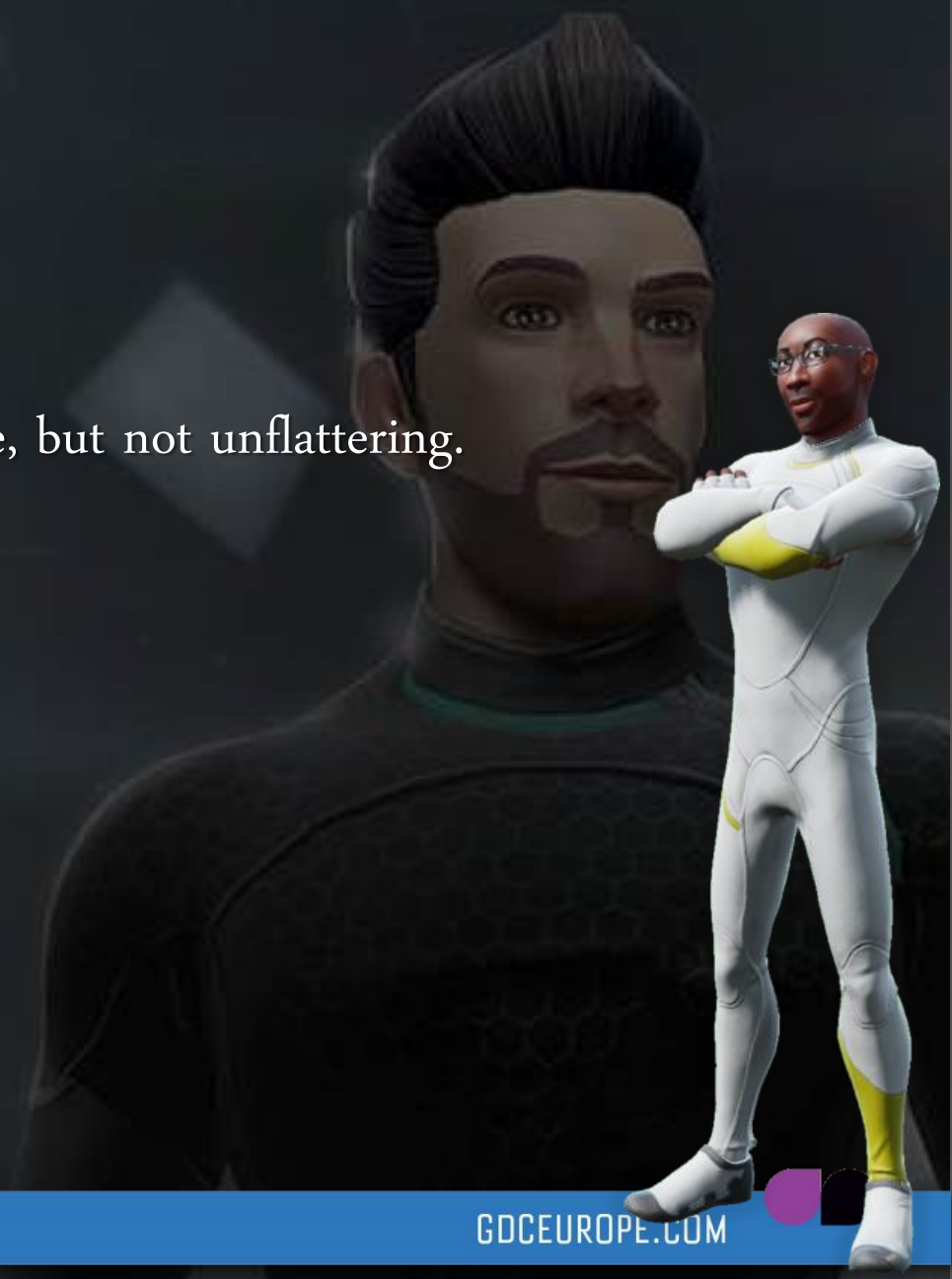
Body Size

- First we measure height using Kinect skeleton tracking
- Then torso width and extent using depth and RGB feed
- Then apply to final model



Key Learnings

- Hard to get positive aspirational result
  - Weight often key part of visual identity
  - Weight not often “aspirational”
  - Solution was to find an aesthetic which validates size, but not unflattering.



Glasses & Facial Hair

- Raw classifier developed by team in Redmond
- Available as part of Xbox One Expressions XDK
- Uses ActiveIR for lighting independence
- Random Decision Forest classifier, trained with thousands of images

Glasses & Facial Hair

- Expressions API gives us a point result
  - Noisy
  - We need to average the result over frames and test above a tolerance
  - Initially tweaked by hand, then automated in a script
- Facial Hair
  - Not a binary classification
  - Created a “low confidence” beard to cope with false positives

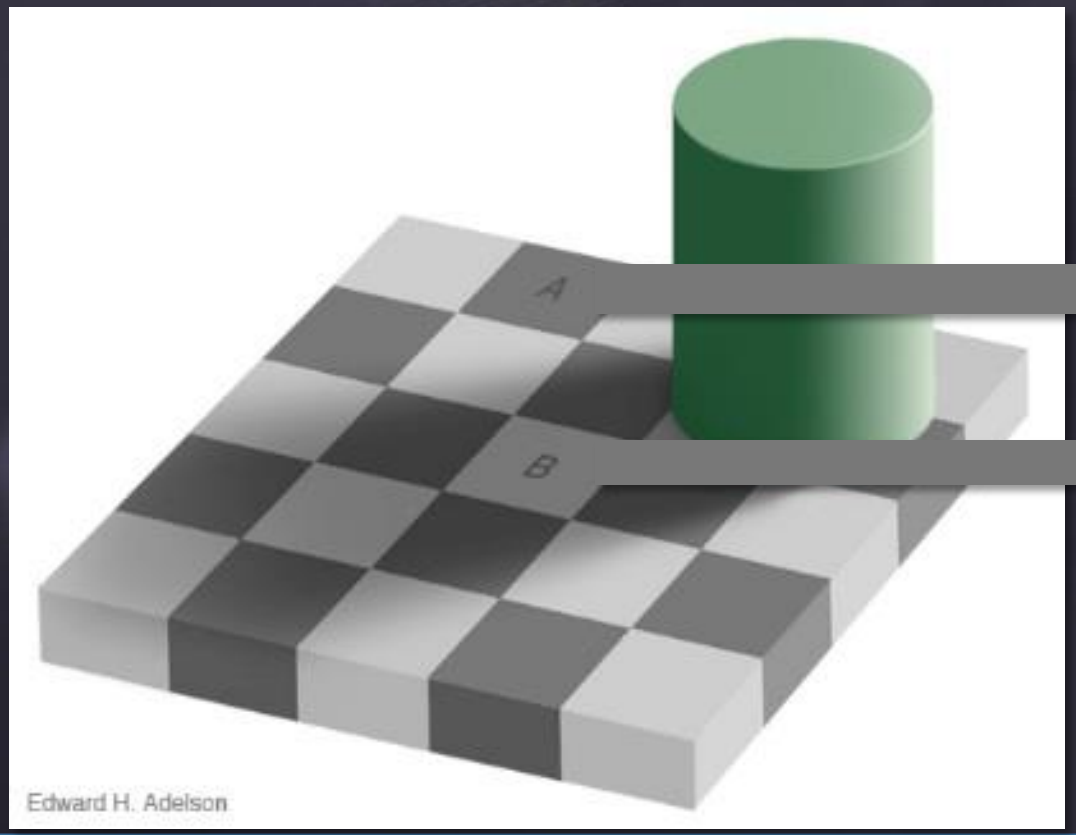
Key Learnings

- Machine learning does not have to be complex!
  - Can be as simple as a brute force offline tool
- Weight your failure cases to get best results
  - Score “correct” results highest
  - Then “acceptable”
  - Then “ok”
  - And sort results to give the best overall result.

Skin Tone

- Your brain is very good at estimating lighting models.

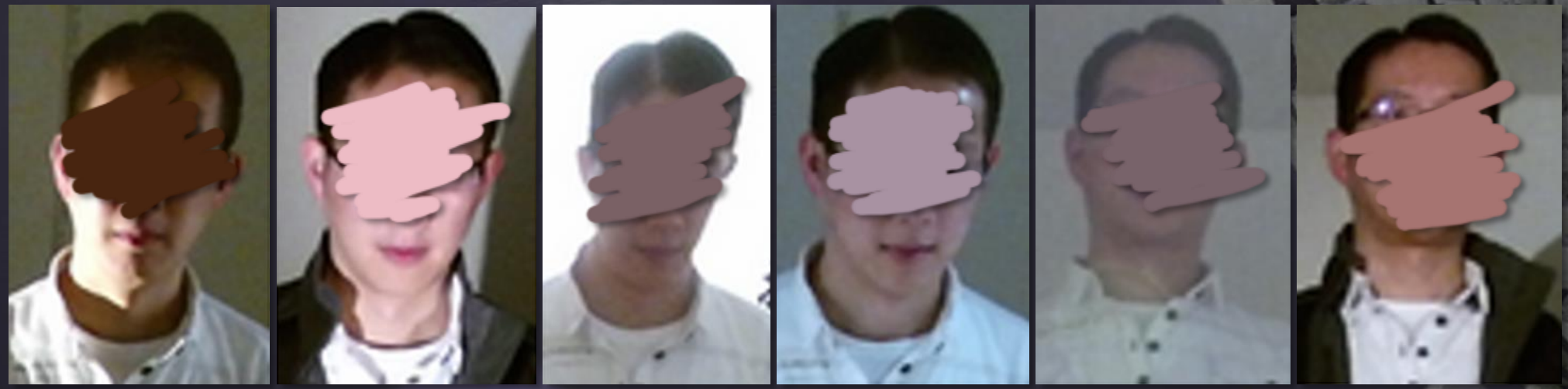
  Computers are not.



#787878

Skin Tone

- How important is lighting?



Skin Tone

- The solution: Active IR
- First, we correct for the lighting

RGB | Skin Weighting | Raw IR | Shape Contour | Weighted IR

- Then we average the pixels and compare with known ranges

0.98875 = “light medium” =

Key Learnings

- You need to manage expectations with Machine Learning systems.
  - Perception was that it was “easy” problem
  - Unlikely that any Machine Learning system will hit 100%

- Identify problem data for your approach, and source more of it!
  - We gathered lots of clips of people in low-light conditions
  - Allowed us to quickly test hypotheses to see if they showed promise
  - Iterate, Iterate, Iterate!

Hair Style

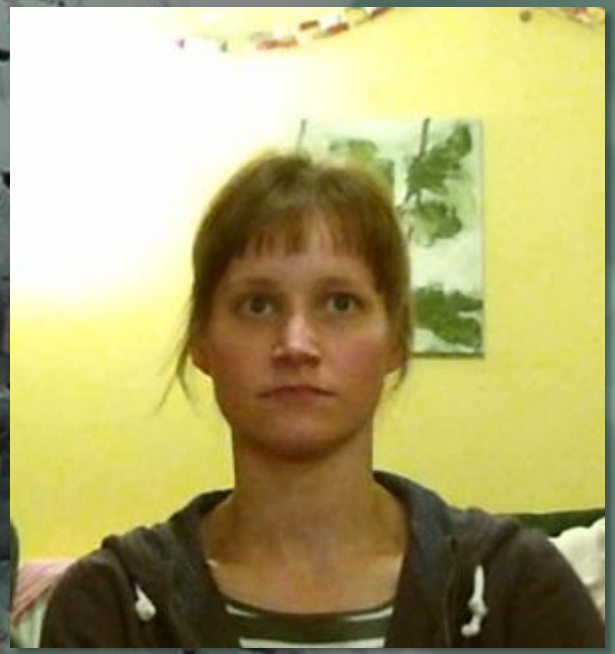
- Uses combination of depth feed & hair segmentation in RGB
  - Estimates volumes of hair for: Top, Side & Below Ear
  - Picks most appropriate hair asset based on results

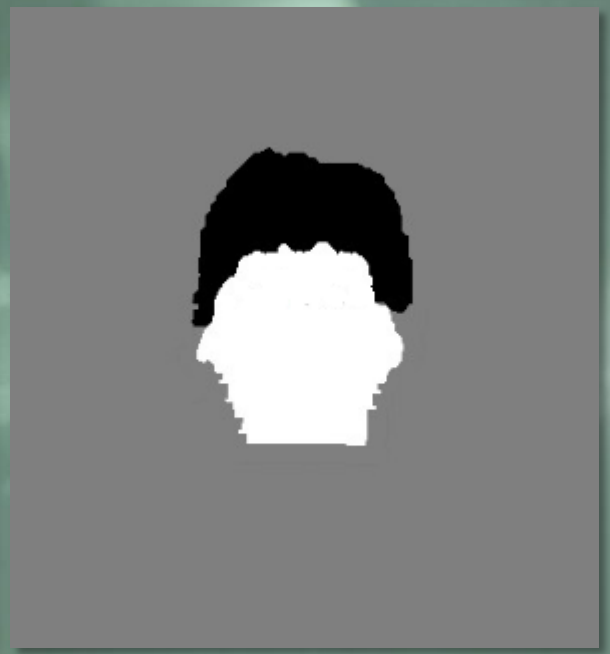
Hair Style

- If subject is male and has very short hair we run a bespoke baldness classifier
  - Looks for curvature of the forehead
  - Looks to find hairline using RGB feed

Fringes & Pony Tails

- Use hair segmentation for fringe
- If subject is female and has short hair, we assume a pony tail





Hair Colour

- Average RGB pixels from hair

Key Learnings

- It's OK to cheat
  - If we don't detect long hair on females we assumed a pony tail
  - Least offensive "wrong" result
- If a result is sensitive add as many layers of security as possible
  - 3 separate tests for baldness
  - Very low false positive result

Animation

- The final result needed to be deformable, yet animate
  - In total 490 blendshapes to deform and animate head
  - Full animation rig mapped onto blendshapes
- GPU bottleneck was transferring blendshapes to GPU
  - Optimisation was to bake “static” blendshapes into new mesh
  - Only transfer animation blendshapes

How We Validated Our Results

- We sourced 850 clips of people being scanned.
  - 6 territories (London, Madrid, Turkey, Japan, China, US)
  - Strategic mixture of age, gender & ethnicity

How We Validated Our Results

- Each clip annotated to give "ground truth" details about subject.
  - Simple csv file with id, path to recording and results expected

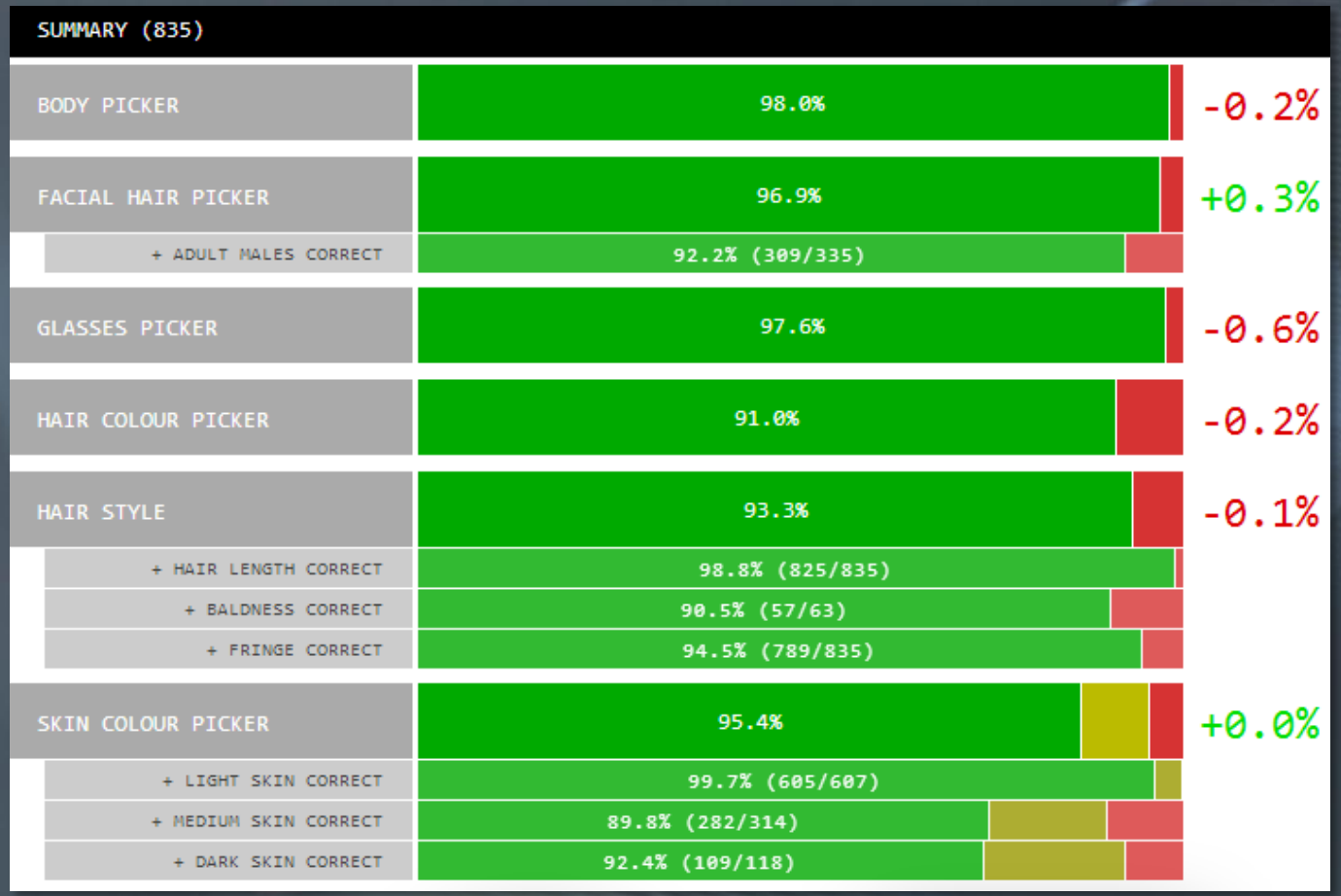
clip	age	gender	exp_facialhair	exp_glasses	exp_haircol	exp_haircategory	exp_fringe	exp_skin	exp_body
\\rareshar	adult	male	yes	no	brown	short	yes|no	light	thin|medium
\\rareshar	adult	male	yes|no	no	grey	short	no	light	thin|medium
\\rareshar	adult	male	no	no	blonde|brown	short|buzz	no	light	thin|medium
\\rareshar	adult	male	no	no	brown	short	no	light	large
\\rareshar	adult	male	no	yes	black	short	no	light	thin|medium
\\rareshar	adult	male	no	no	blonde	short	yes|no	light	thin|medium
\\rareshar	adult	male	yes|no	yes	bald	bald	no	light	medium|large
\\rareshar	adult	female	no	yes	brown	pony	no	light	medium|large
\\rareshar	adult	male	yes|no	no	brown	short|tall	no	light	thin|medium
\\rareshar	adult	female	no	no	black	long	no	light|med	thin|medium
\\rareshar	adult	male	yes	no	black	short|tall	no	light	thin|medium
\\rareshar	adult	male	yes|no	no	black	short	no	light	thin|medium
\\rareshar	adult	male	yes	no	black	buzz	no	light	thin|medium

How We Validated Our Results

- Automatic process to run each clip with latest code
  - Hooked into our automatic build process
  - Ran on 16 devkits in 3 hours
  - Twice daily

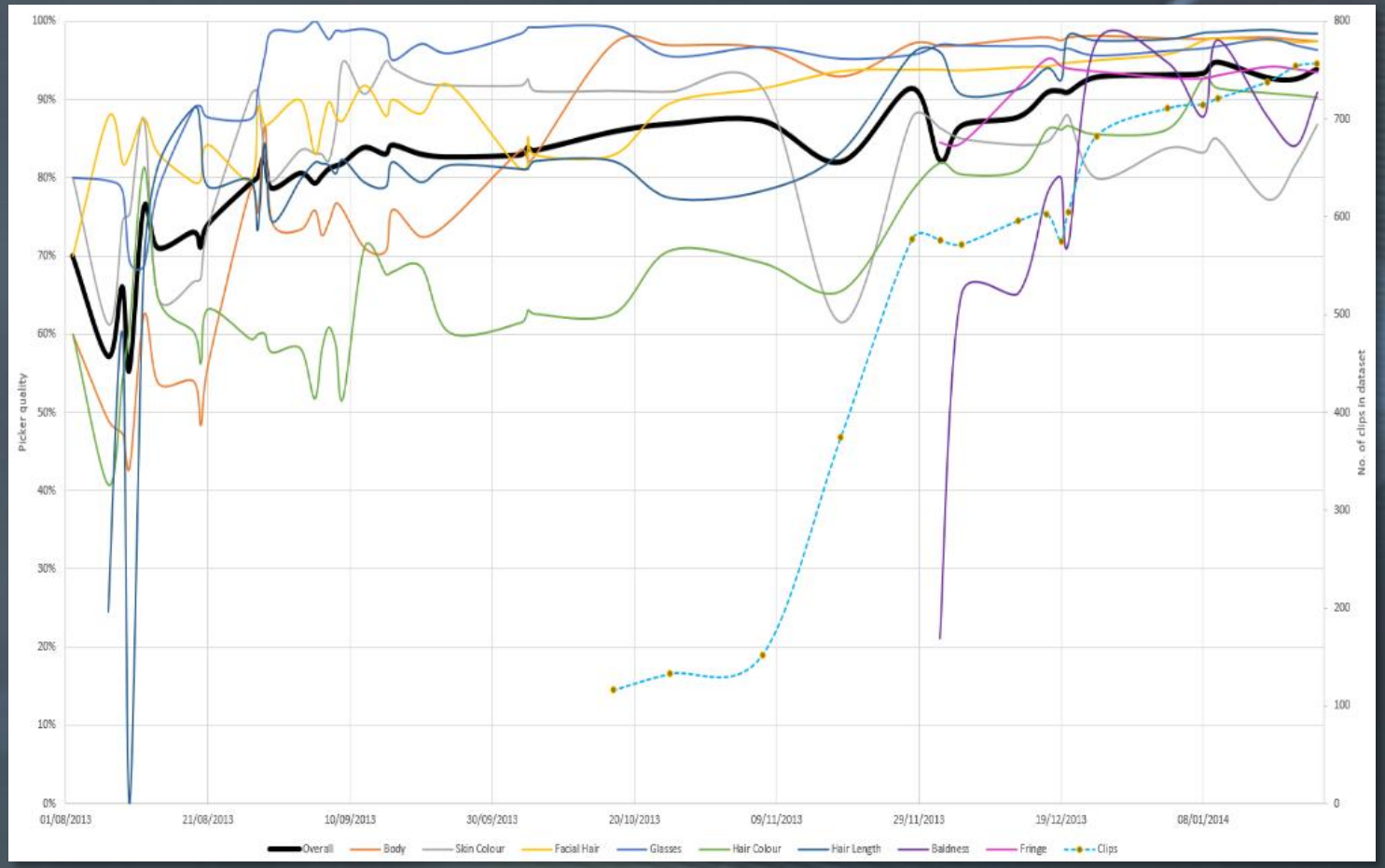
How We Validated Our Results

- Generated html report with full data and deltas



How We Validated Our Results

- We were able to track progress over time



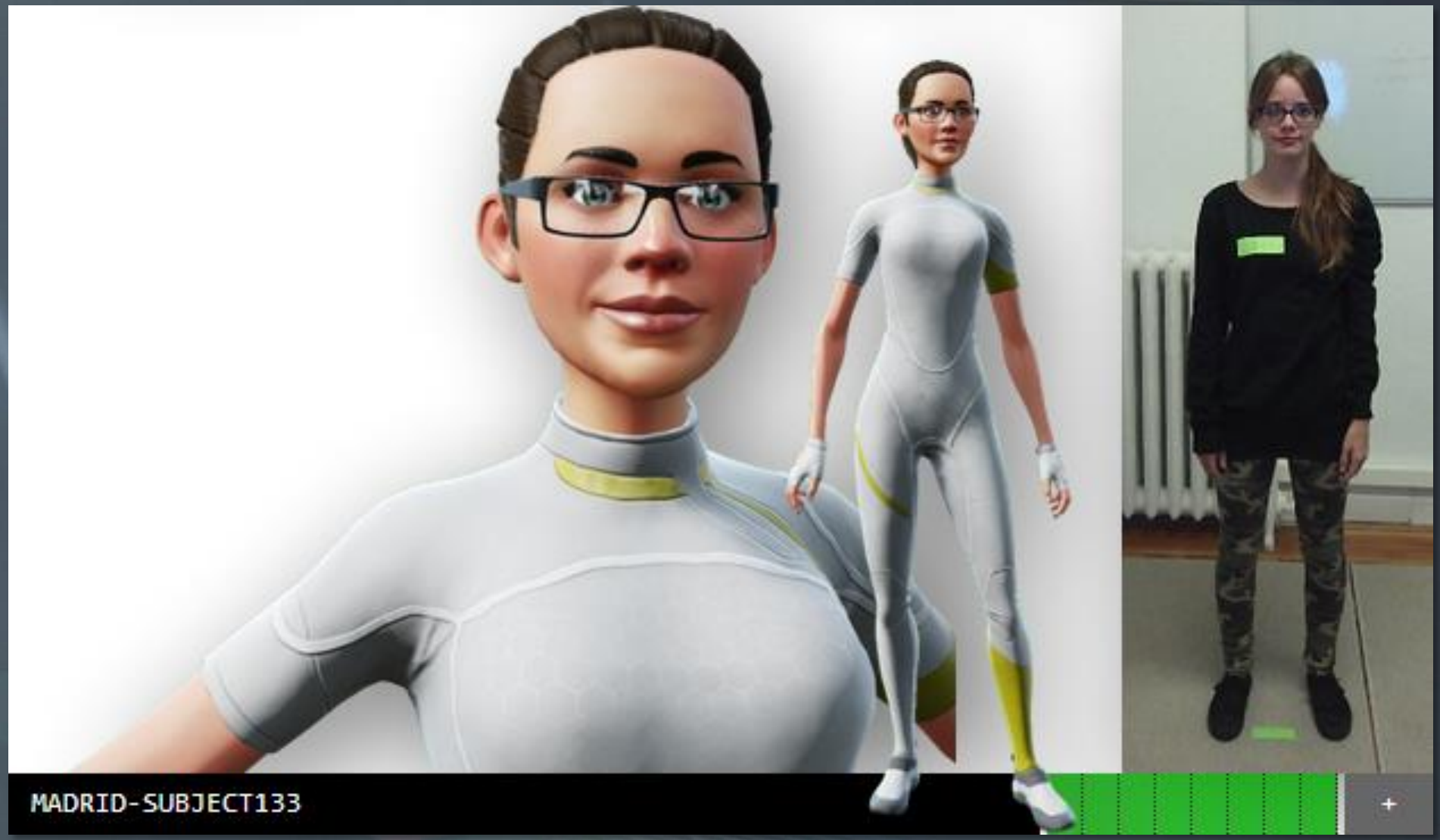
Key Learnings

- Machine learning lives or dies on the quality of source data
- 24 hour cycle of improve, observe, validate, repeat
- Cut corners where you can
  - You are unlikely to hit 100%, so goal is to maximise results
  - Test a simple assumption, it might save a lot of work

Some Results

Some Results

Some Results



Some Results

Some Results

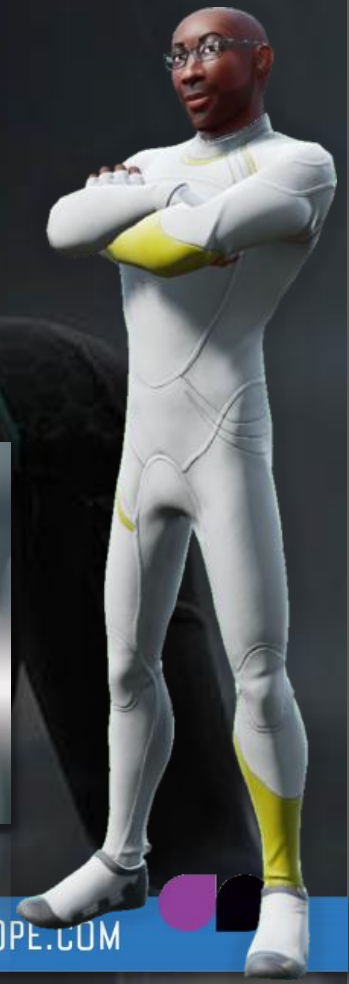
Some Results

Some Results

Some Results

Scanning Experience

- Whimsical/playful tone
- Dr Who!
- Required a LOT of User Research
- Biggest challenge: positioning the user

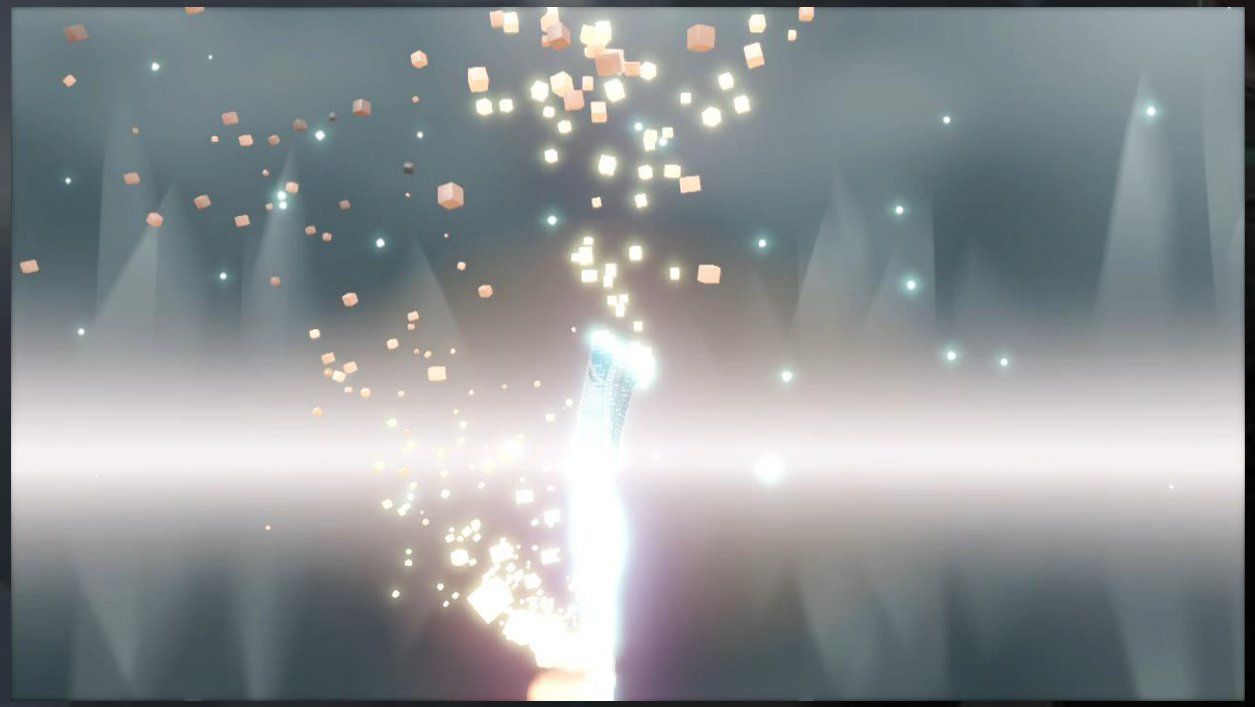


Scanning Experience

- 24 hour cycle of User Research and reaction
  - All engineers observed sessions
  - Quick deadline to verify improvements

Scanning Experience

- The reveal cutscene
  - Create tension and anticipation
  - Fun payoff to the experience



What Went Well?

- End result is good
- Scanning flow works well for almost all users
- Machine vision works well for most users
- Automated testing gave us launch confidence

What Could We Have Done Better?

- Data Capture was started late.
  - Get data early!
- "Experience" user research was started early enough but not initially useful due to missing build functionality
- Result trends towards generic for ~50% of users
- Hair styles were correct, but often uninspiring


Any Questions?
andy.bastable@microsoft.com
@andybastable
#GDCEurope