FACIAL PERFORMANCE PIPELINE

SPENCER ALEXANDER
Technical Director

OVERVIEW – SUCKER PUNCH FACIAL PIPELINE

SUCKER PUNCH

WHAT WE HAD BEFORE

- Real-Time Cutscenes
- Hand Keyed Animations
- Joint Only Face Rigs

WHAT WE WANTED FOR I:SS

- Impactful BackStory
  - Realistic Characters and Animations
- Facial Motion Capture
  - Bodies/Faces/Eyes/Audio -Simultaneous
  - Multiple Actors –Simultaneous
- Real-time Cutscenes

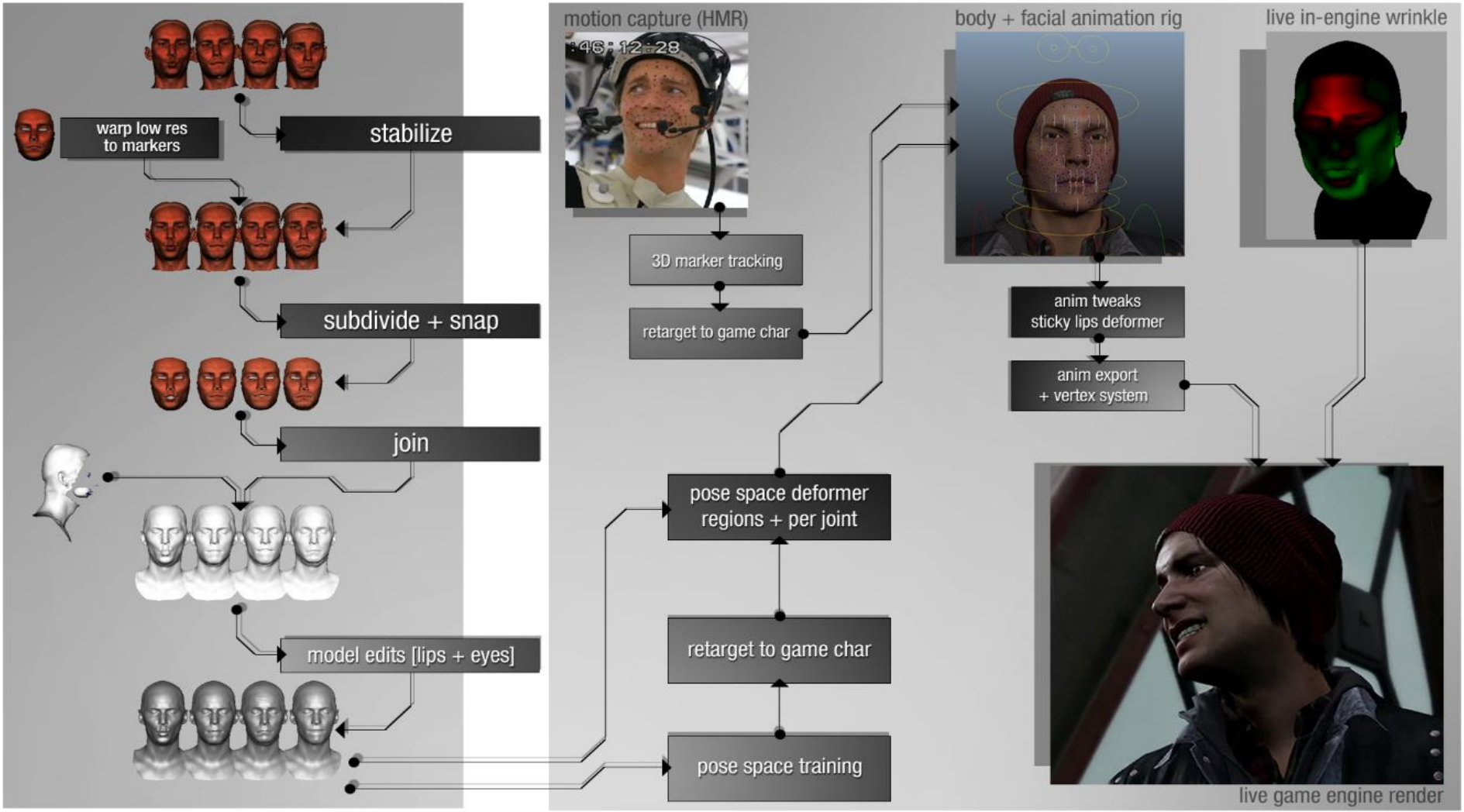
FACIAL PERFORMANCE METHODS

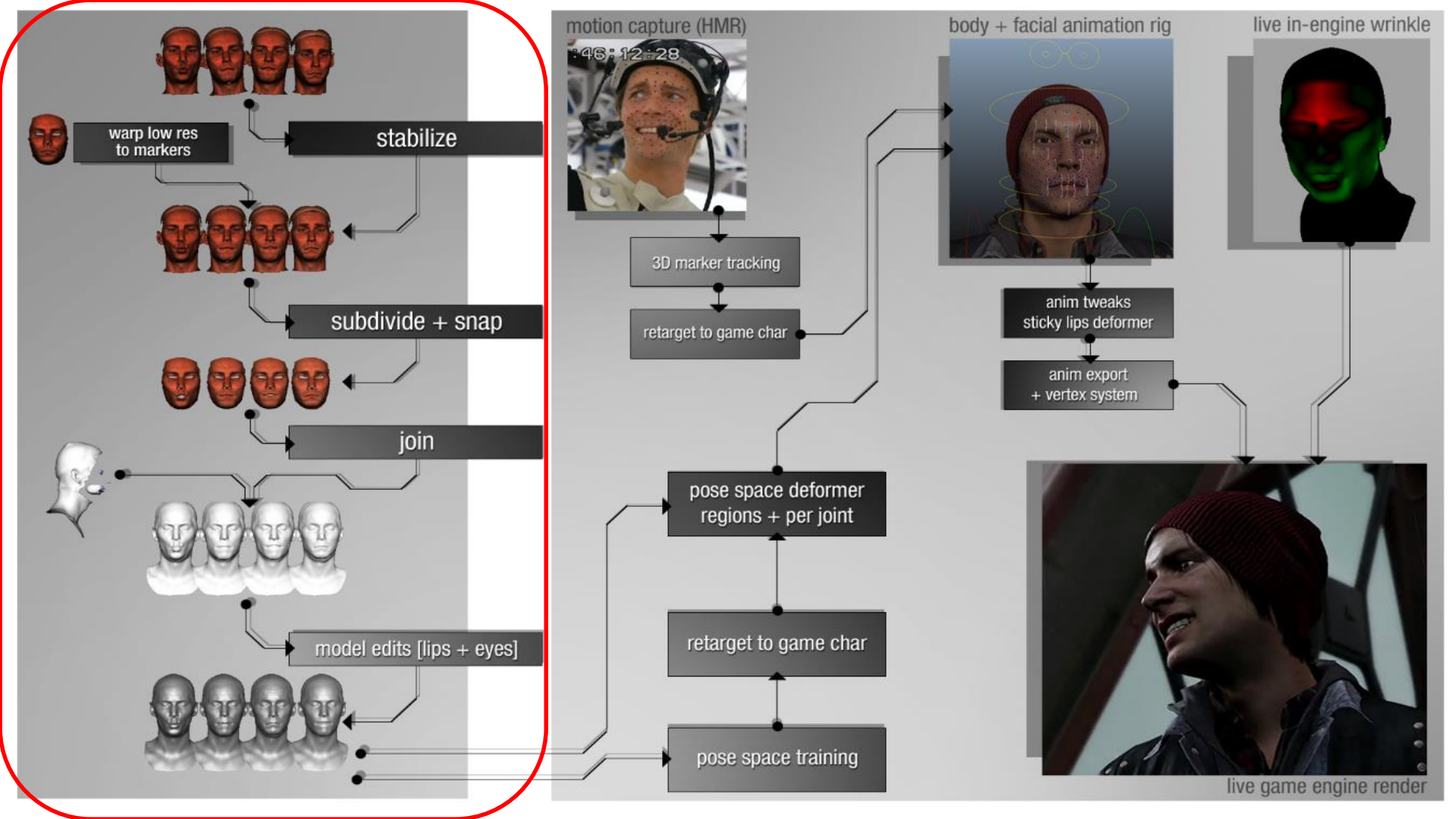
- Hand Keyframed
- 3d Scan Capture Per Frame Performance
- Muscle-Based/Simulation
- Data Driven/Example Based

SUCKER PUNCH

METHODS | DATA-DRIVEN/EXAMPLE-BASED

- Requires a Data Set of known example input to ouput
- Enough data to make a generalized representation
- Data-Set
  - Facial Scans (F.A.C.S) (Paul Ekman)
  - Consistent Facial Marker Set (scans/mocap)
- New Inputs
  - Tracking (same marker set)
- Solver (Applies Mocap Data to Rig/Shapes)
- Motion Retarget to Game Characters

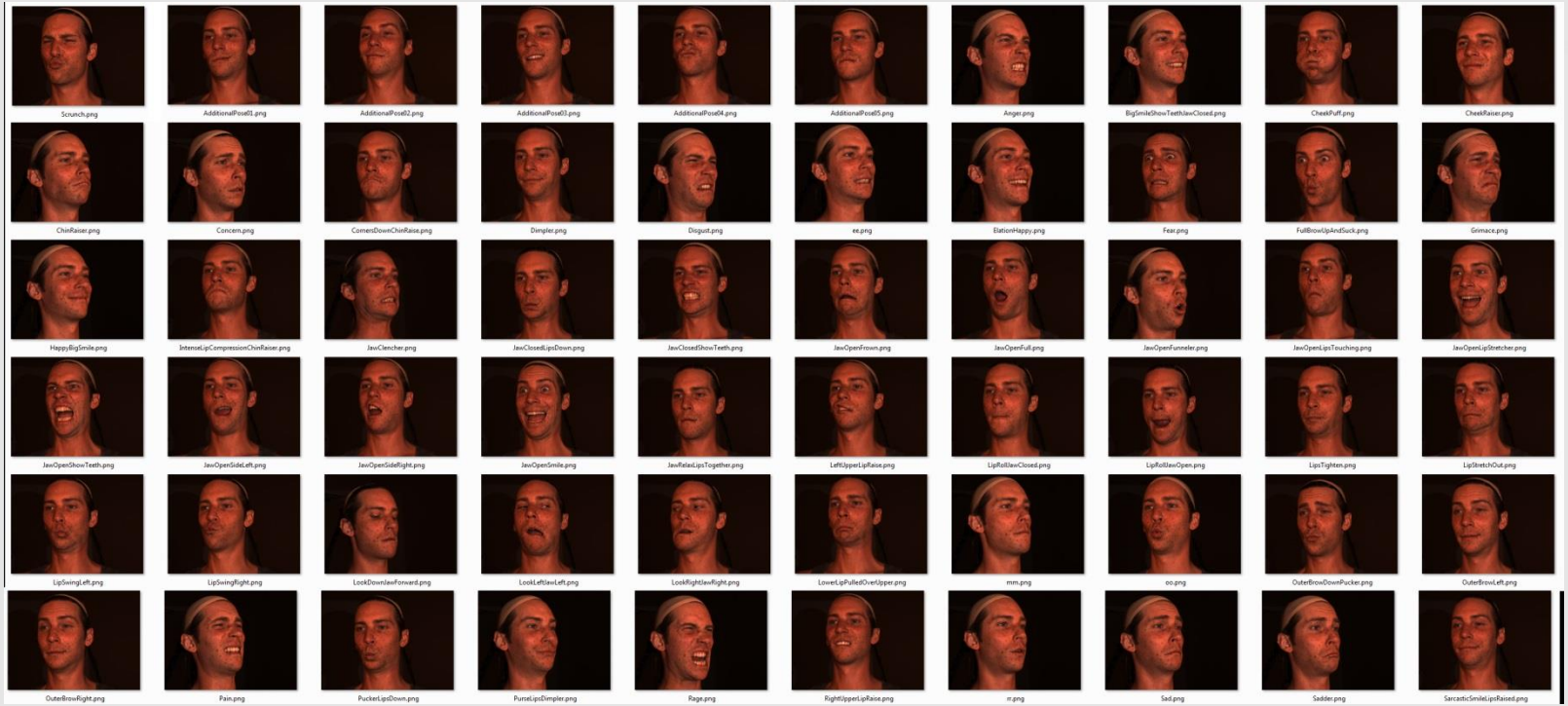

warp low res to markers
stabilize
subdivide + snap
join
model edits [lips + eyes]
motion capture (HMR)
3D marker tracking
retarget to game char
pose space deformer regions + per joint
retarget to game char
pose space training
body + facial animation rig
anim tweaks sticky lips deformer
anim export + vertex system
live in-engine wrinkle
live game engine render

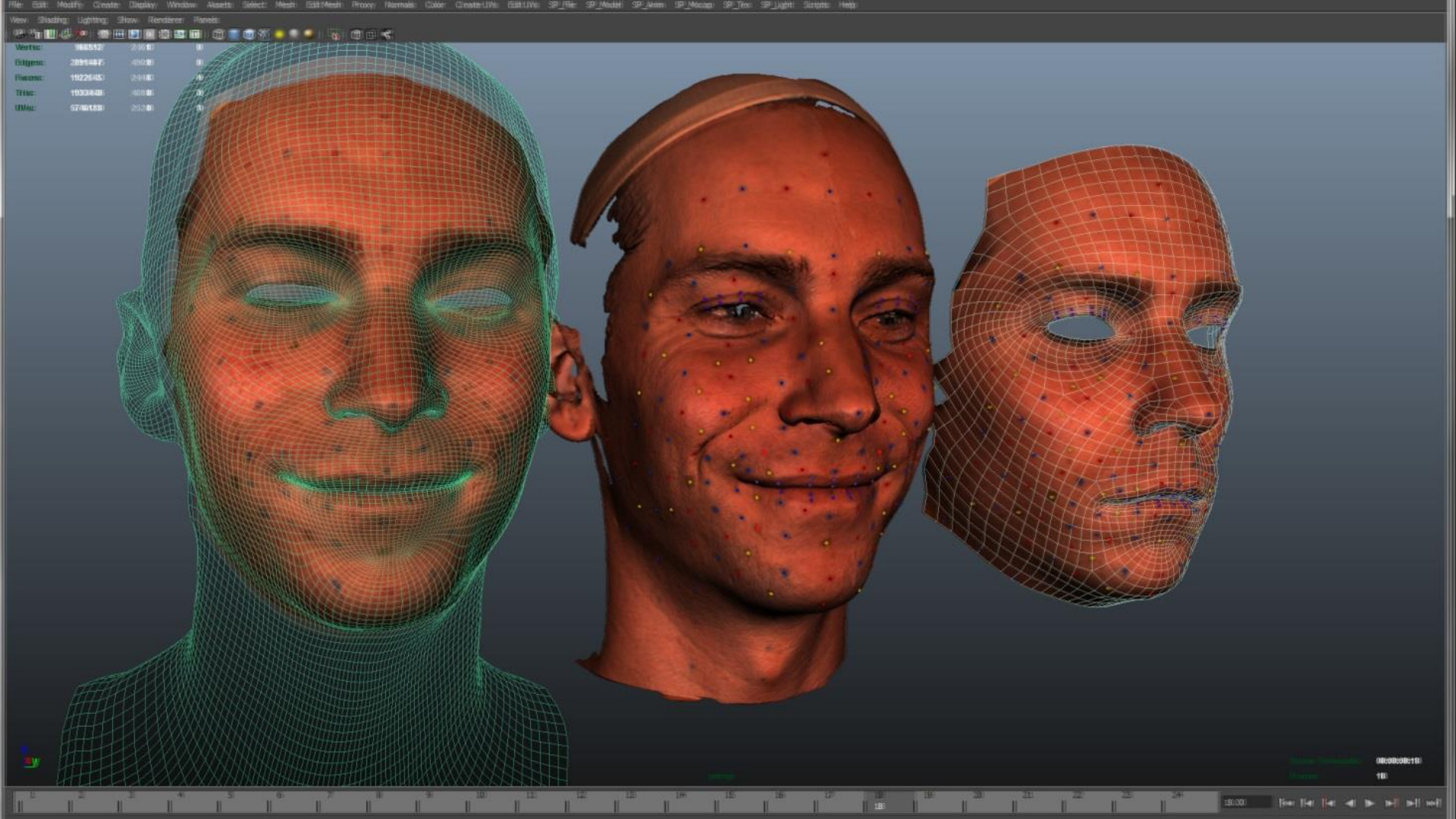

warp low res to markers
stabilize
subdivide + snap
join
model edits [lips + eyes]
motion capture (HMR)
3D marker tracking
retarget to game char
body + facial animation rig
anim tweaks sticky lips deformer
anim export + vertex system
live in-engine wrinkle
pose space deformer regions + per joint
retarget to game char
pose space training
live game engine render

DATA-DRIVEN | F.A.C.S. SCANS

- Pre-Captured (One time setup cost)
- 70 Facial Pose Scans per actor
- Scan Re-mesh to consistent Topology
- Scan Hardware Used
  - 2 linked structured light field Scanners
- Captured in Static Volume
- Stabilize/Align Scans

DATA-DRIVEN | FACS POSES

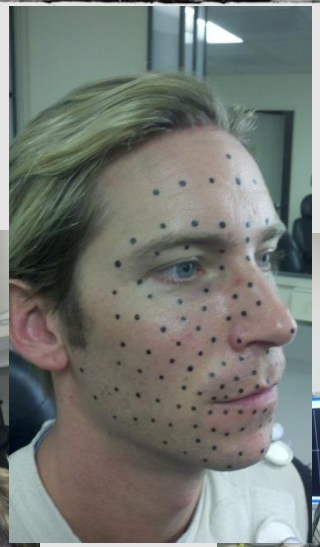
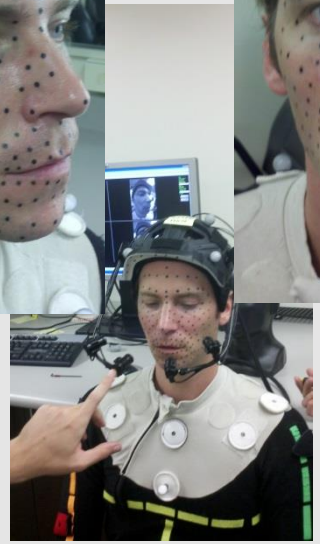
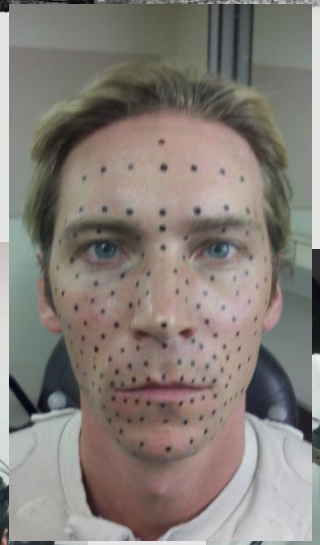
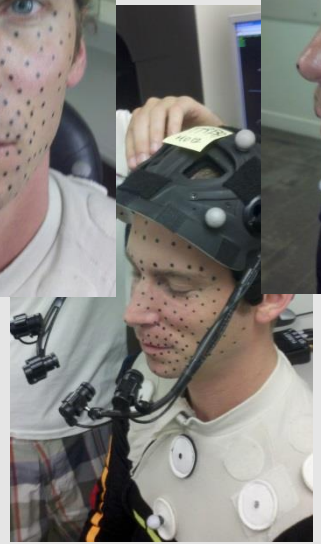
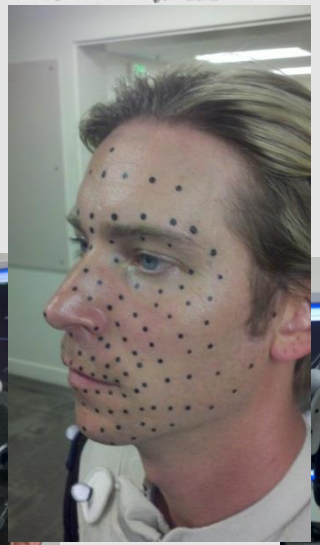
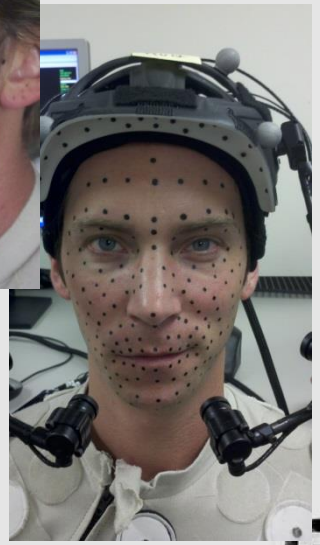




DATA-DRIVEN | F.A.C.S. SCANS

- Consistent Topology/Blendable
  - Consistent Marker Correspondence
  - Custom Vacuform Masks
  - Track Markers/Warp Fit/ Subdivide/Snap
  - Combine Faces with Entire Head
  - Model Tweaks (Noisy/Missing Scan Data Eyes/Mouth)

MARKER PLACEMENT

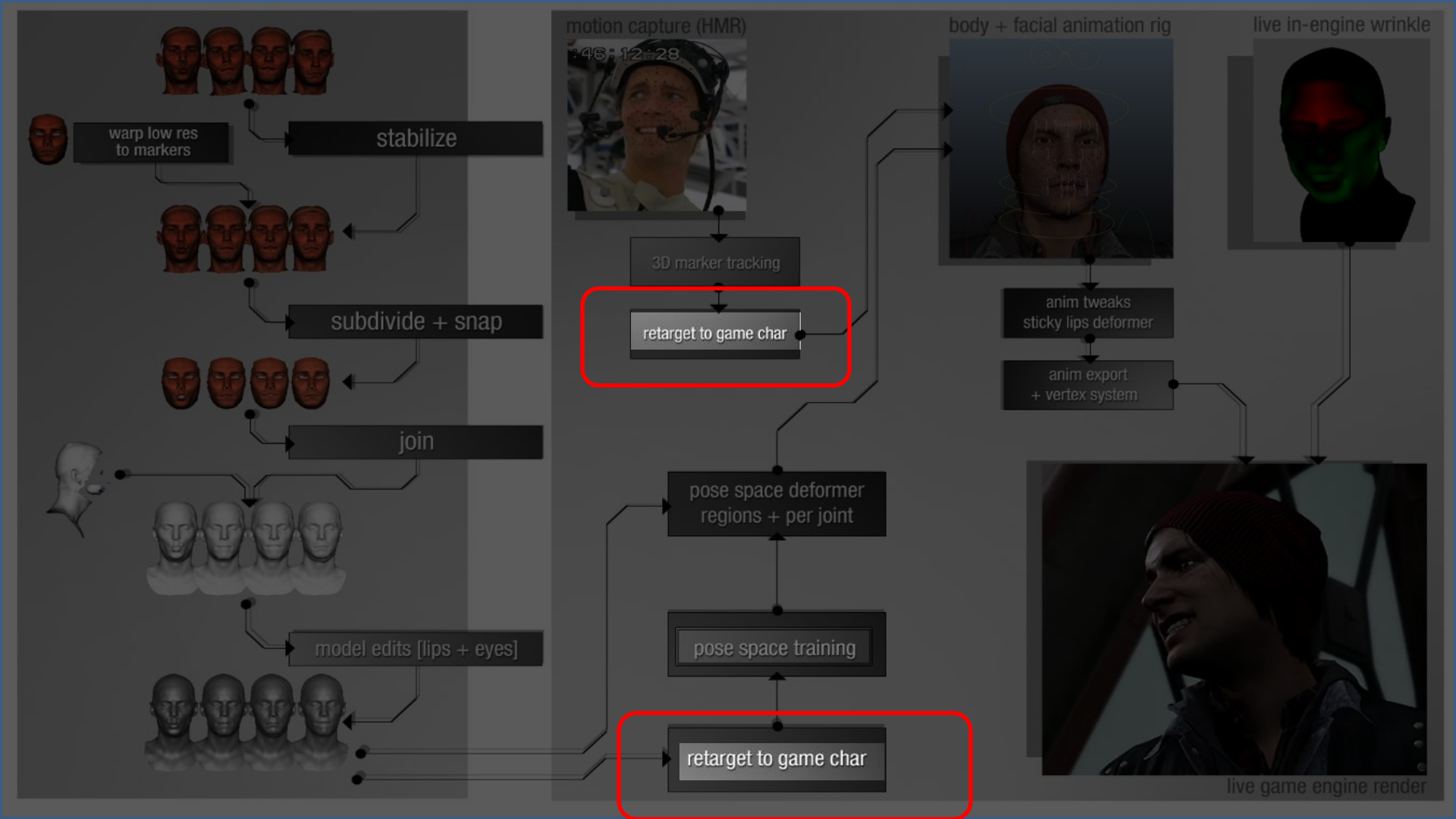

motion capture (HMR)
:46:12:28
3D marker tracking

DATA-DRIVEN | MOTION CAPTURE

- Head Mounted Camera Rigs (HMR)
- Simultaneous Capture – Face/Body/Eyes/Audio
  - Timecode Hardware Sync
- Multiple Actors
- Projection Mask Animation
- Marker Tracking

DATA-DRIVEN | MOTION CAPTURE | TRACKING

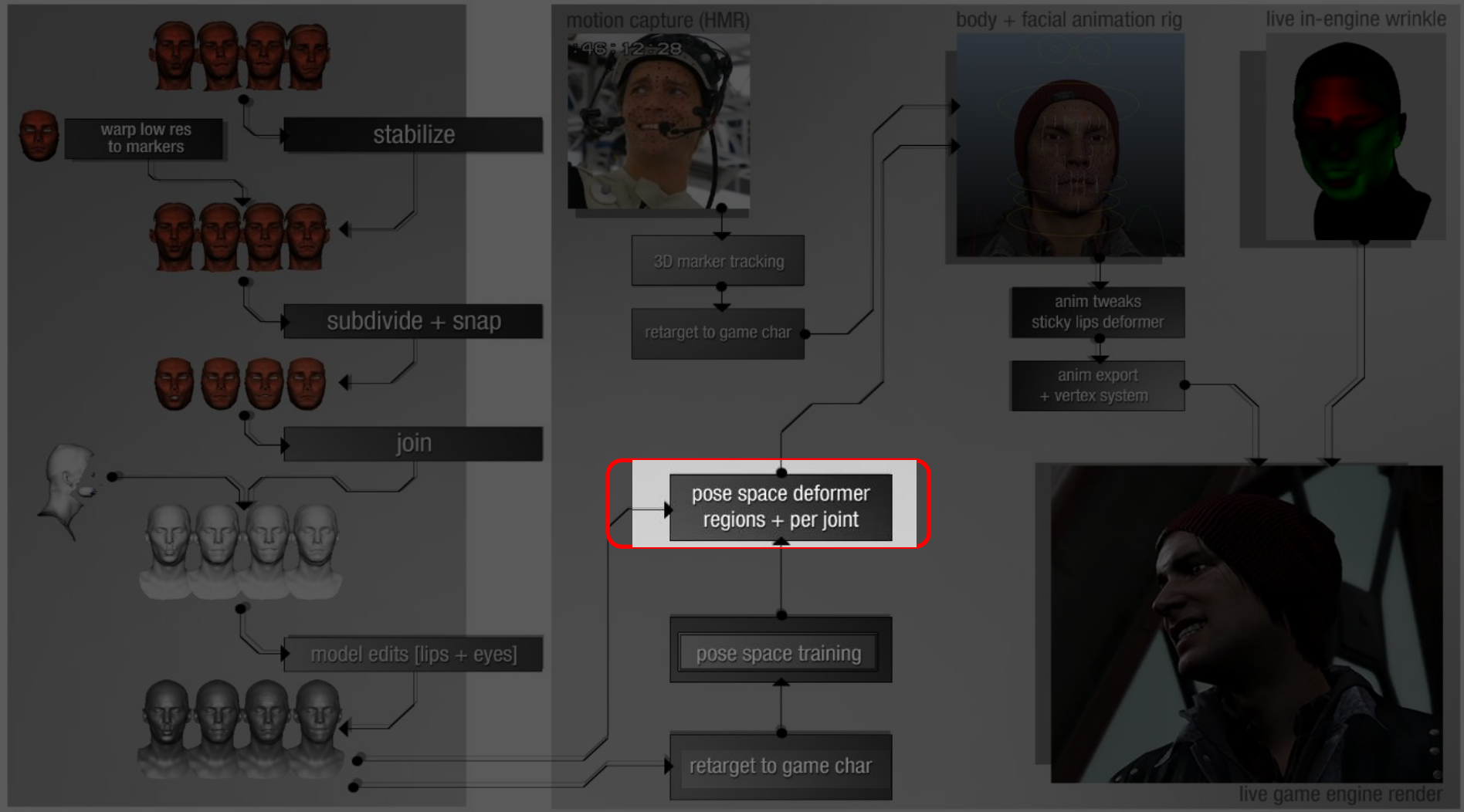
- Methods (2d vs 3d)
  - Optical Flow
  - Feature Recognition
  - Triangulation Software
- Noise Reduction
- Stabilization
  - Moving helmets/Day to Day/Shot to Shot align
- Missing Marker Data/ Occlusion
- Lighting Conditions


warp low res to markers
stabilize
subdivide + snap
join
model edits [lips + eyes]
motion capture (HMR)
3D marker tracking
retarget to game char
body + facial animation rig
anim tweaks sticky lips deformer
anim export + vertex system
live in-engine wrinkle
pose space deformer regions + per joint
pose space training
retarget to game char
live game engine render

RE-TARGET TO GAME CHARACTERS

- Methods
  - Game Character Face Rig, trained to use Drivers from actor for Blend Weights
  - Transfer Blend Weight results (Shape Weight Correspondence between character poses if needed)
  - Hint Based Scalars On Base Scaffold
  - Proportion Constraint (method we used)
    - Shape Re-target
    - Marker Motion Re-target
    - Easily Tie motion to joints


persp
Scene Timecode:
Frame:


warp low res to markers
stabilize
subdivide + snap
join
model edits [lips + eyes]
motion capture (HMR)
3D marker tracking
retarget to game char
body + facial animation rig
anim tweaks sticky lips deformer
anim export + vertex system
live in-engine wrinkle
pose space deformer regions + per joint
pose space training
retarget to game char
live game engine render

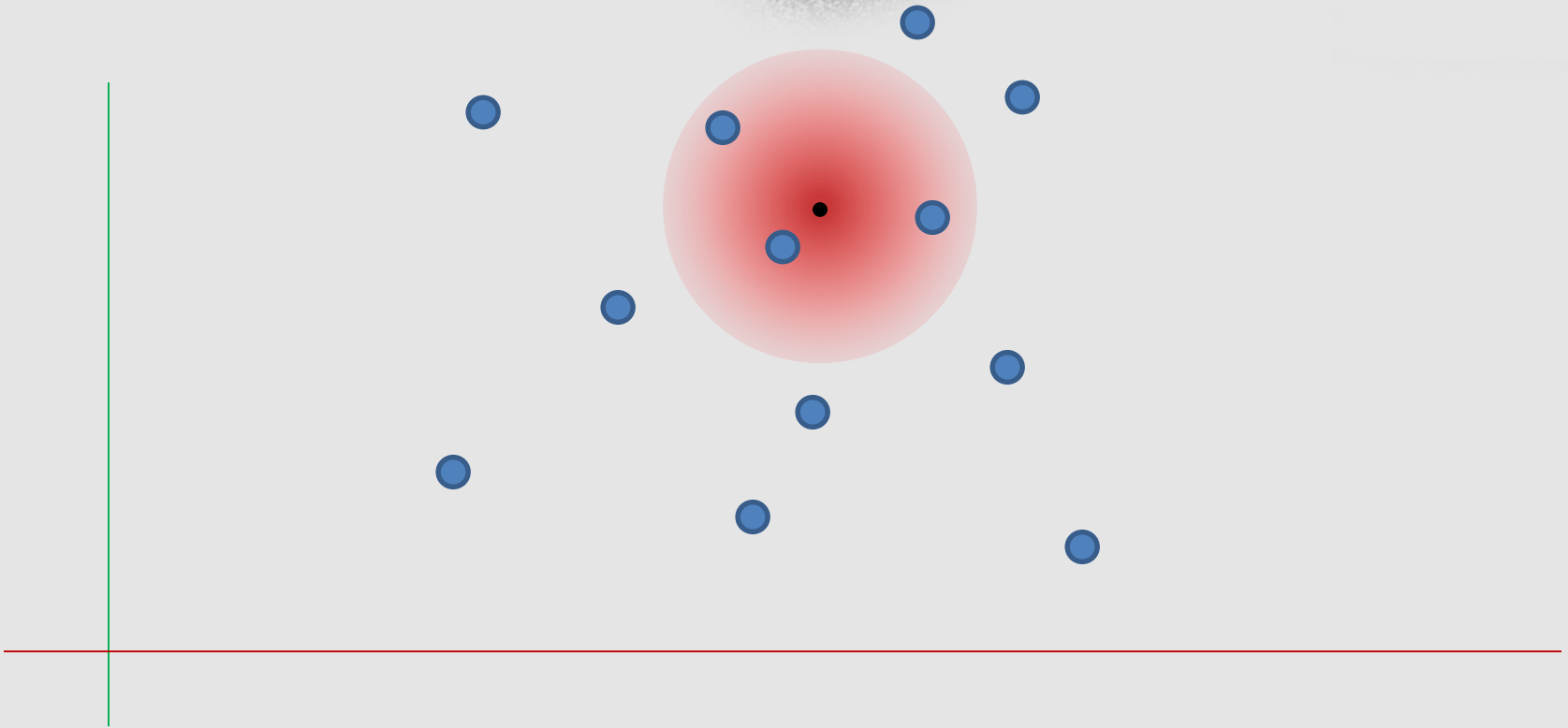
BLENDSHAPES ALONE (NOT PREFERRED)

- Linear BlendShapes
  - Easy to Animate Weights
  - Over-Adding Issues-need to normalize
  - “Posey” results

DATA-DRIVEN | SOLVER

- Pose Space Deformation (PSD) as solver
  - A method of deriving shape weights from marker data
  - Uses Radial Basis Function (RBF)
  - Applied as Deformer node, shapes driven indirectly though a face rig- controls will get automatic shape interpolation
  - Markers as Drivers / Motion Vectors
  - Setting up Matrix/ Single Value Decomposition (SVD)
  - Shapes Converted to PreSkinCluster –Relative to Default

POSE SPACE DEFORMER/ RBF SOLVER



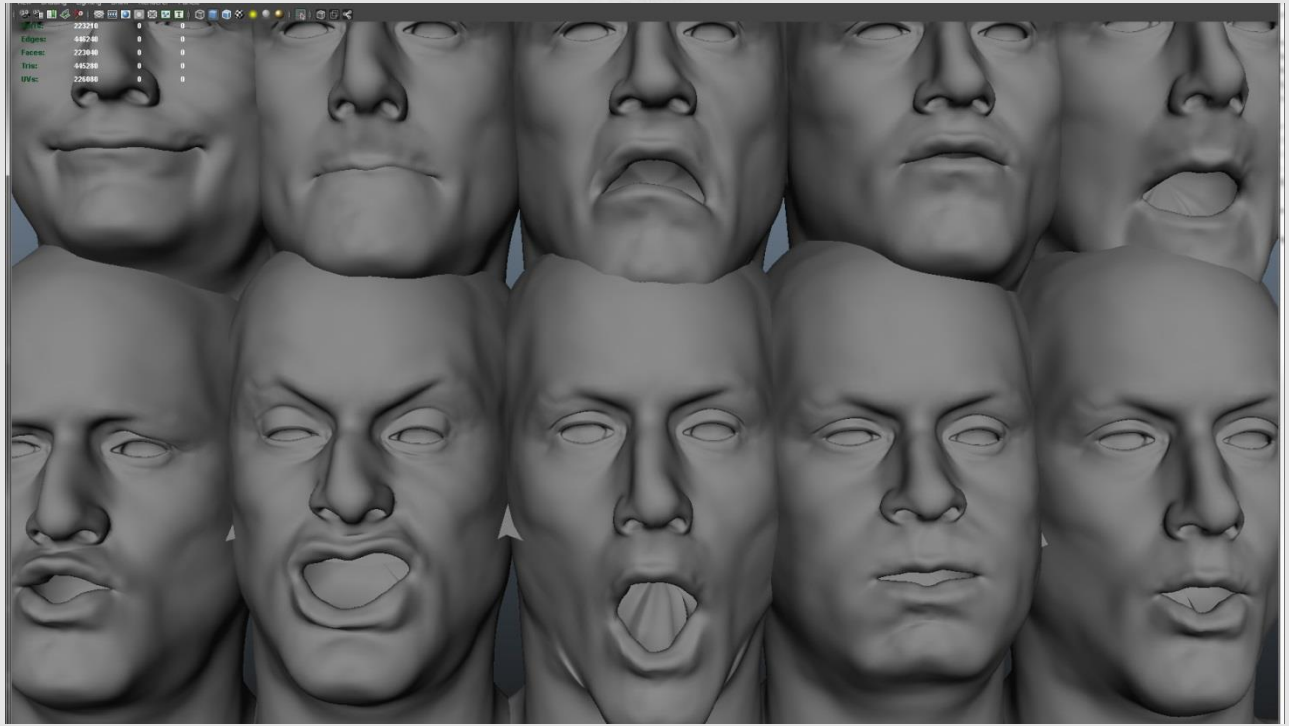
SUCKER PUNCH

Actor
BlendShape

Game Char
SkinCluster
& PSD

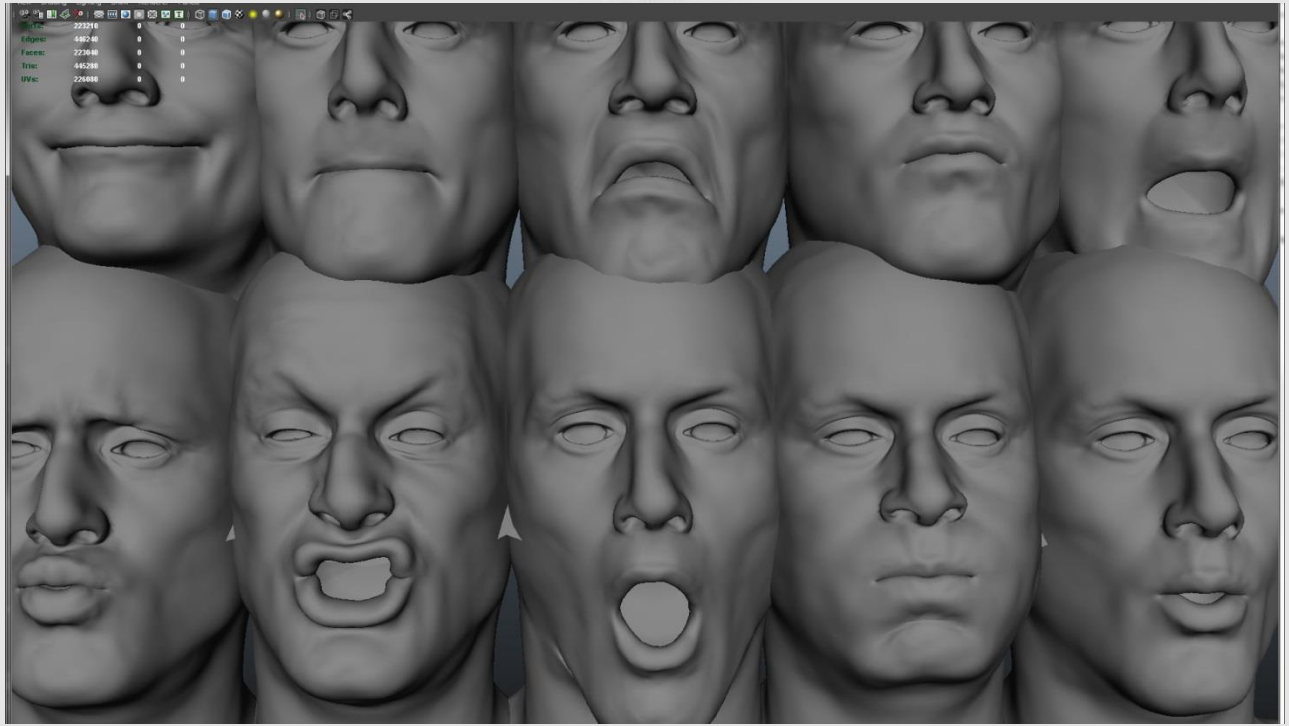
Game Char
BlendShape

SKINCLUSTER ONLY



SUCKER PUNCH

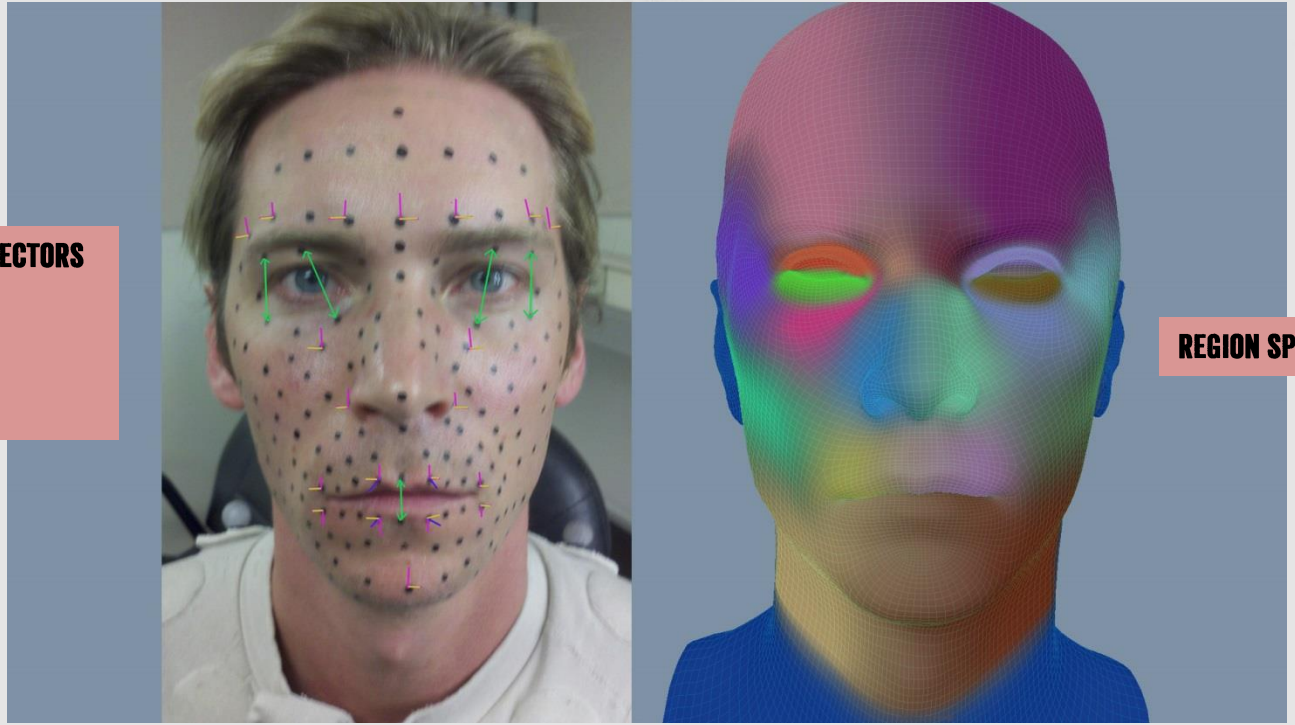
PSD + SKINCLUSTER



SUCKER PUNCH

KEY DRIVERS

LOCAL OFFSET VECTORS
LID ROTATIONS
EYE DIRECTION
DISTANCES
JAW ROTATION



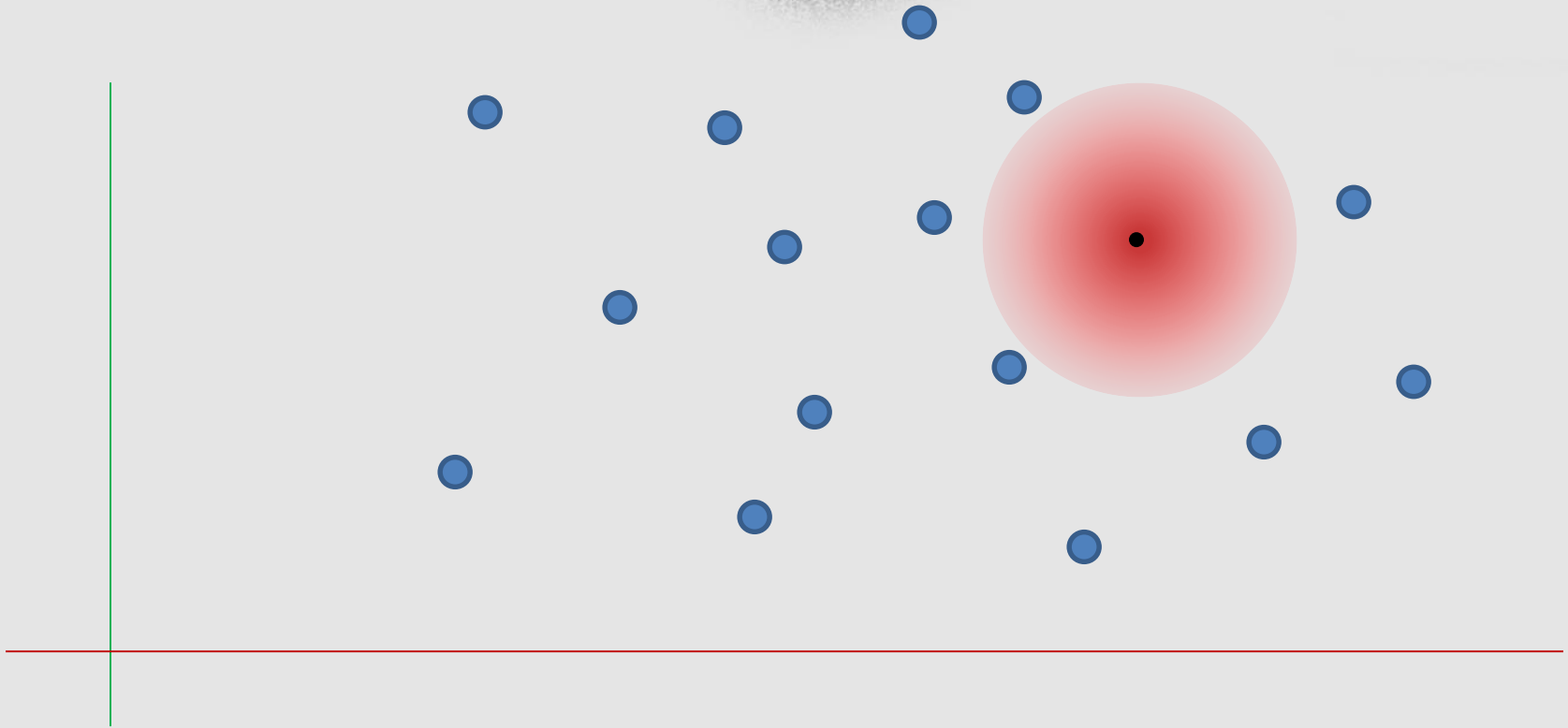
REGION SPECIFIC DRIVERS

SUCKER PUNCH

DATA-DRIVEN | PSD SOLVER

- Fine Tuning PSD Attributes
  - Input Width Regularization
  - Balance Pose Space (Gaps/Shape Overlap)
  - Drop poses too close
  - Testing Drivers – ROM test/ BlendShape test
- Expose Solver Attributes to Animators
- Using Multiple PSDs with normalized deformer weights
  - Fewer Drivers for Independent Regions
  - Like Weighted PSD

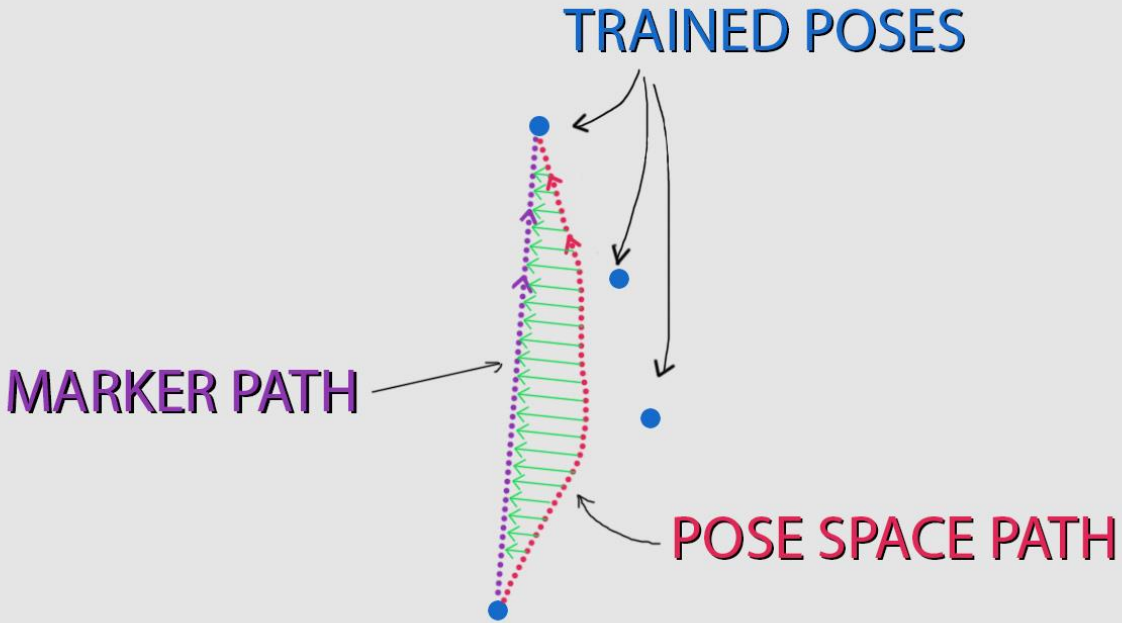
POSE SPACE DEFORMER/ RBF SOLVER



DATA-DRIVEN | PSD SOLVER

- Pose Training Phase (One Time) – surface locators
- PSD Deformer applied Pre-SkinCluster
  - Joints per Facial Marker
  - Track Drives Skin and Vertex Level Deformations
  - Efficient Streaming – No Dense Shape Memory
  - Smooth Natural Arcs in Motion

WARPED SOLVE – SNAP TO 3D TRACK



TRACKING

- 2D versus 3d Tracking
  - 2d requires only 1 camera per head rig
  - 3d requires 4 cameras per head rig
  - 3d track can use the 3d data to verify accuracy
  - 3d track gives better visualization/sillhouette
  - 3d track can be used to warp/fit results tighter
  - 3d track provides extra dimensionality for lips

2D TRACKING ERROR

SUCKER PUNCH

2D TRACKING ERROR

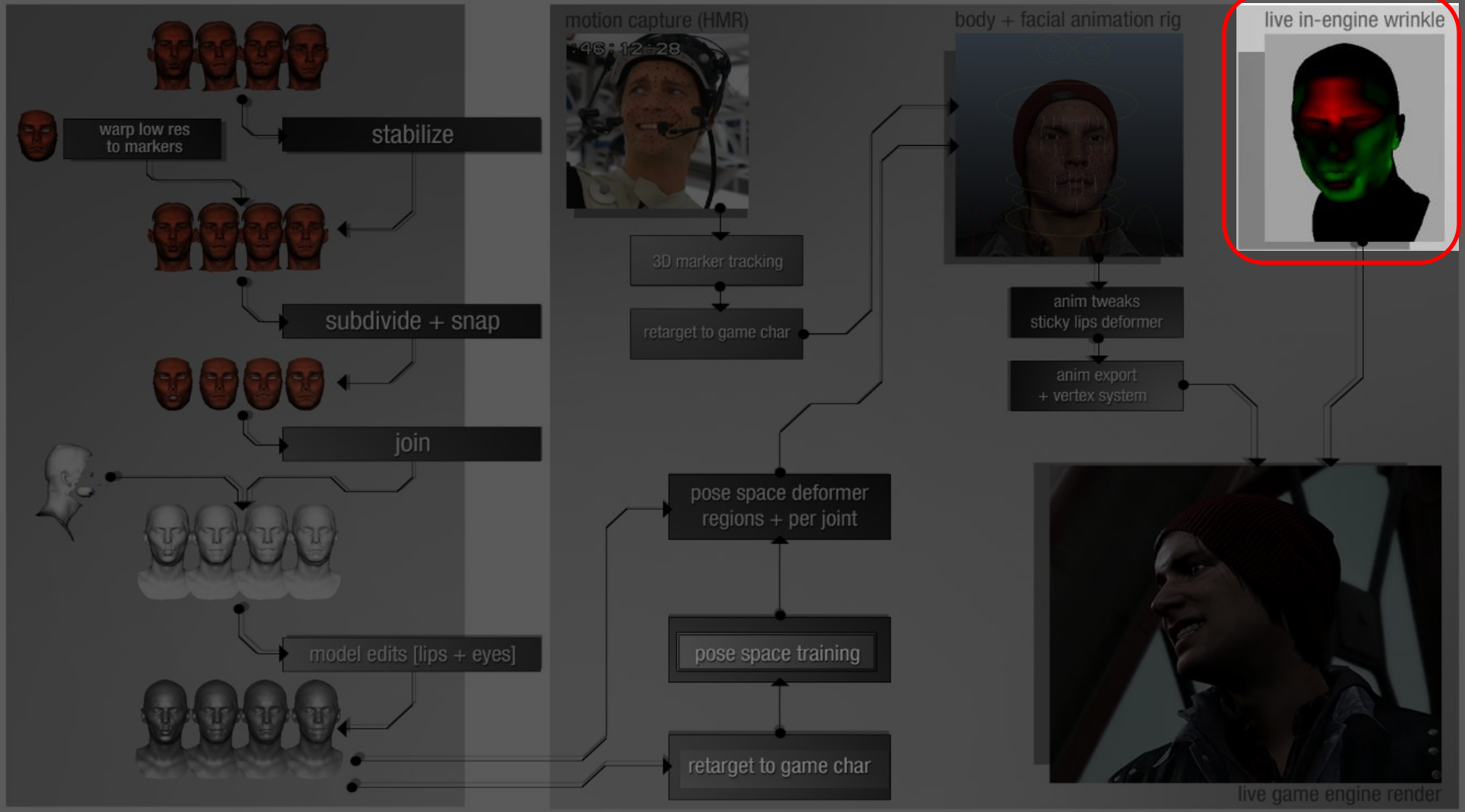

warp low res to markers
stabilize
subdivide + snap
join
model edits [lips + eyes]
motion capture (HMR)
3D marker tracking
retarget to game char
body + facial animation rig
anim tweaks sticky lips deformer
anim export + vertex system
live in-engine wrinkle
pose space deformer regions + per joint
pose space training
retarget to game char
live game engine render

FACIAL RIGS

- Variety of Animation Controls (Gross and Fine Control)
- Gross Controls -Directly Manipulate Key Drivers
- Fine Controls -Directly Manipulate each joint
- UI – Picture per actor pose with multiple Region Sliders
- Automatic Shape Interpolation with all controls

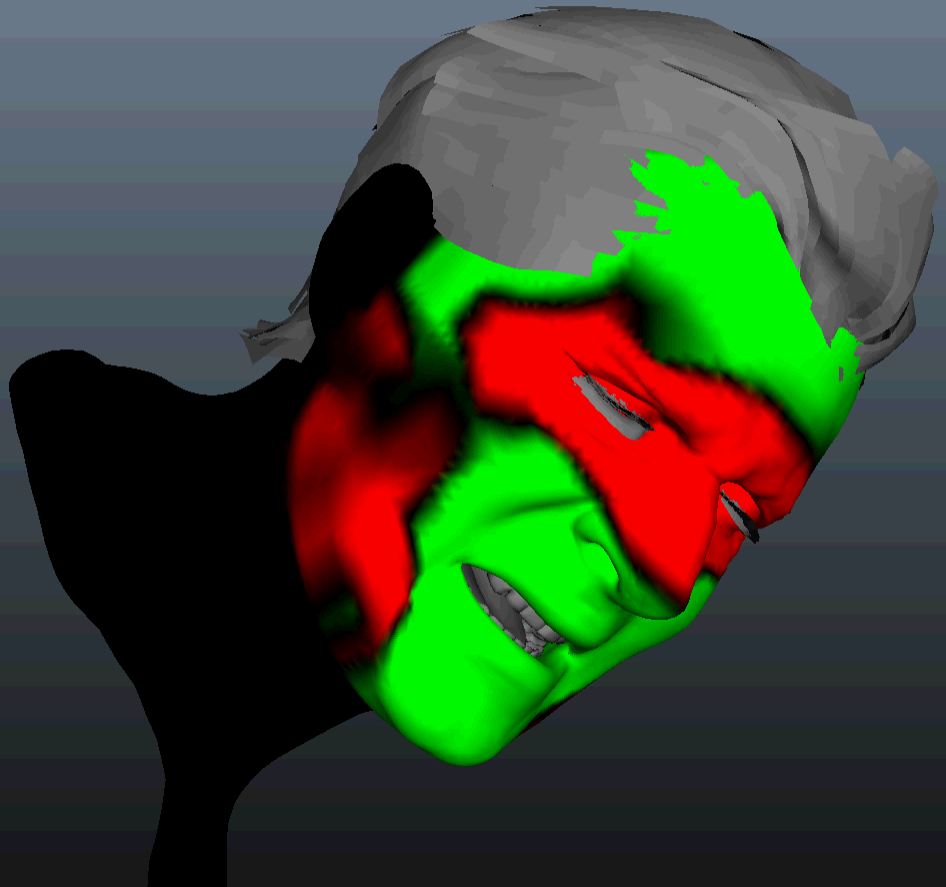
FACIAL RIGS

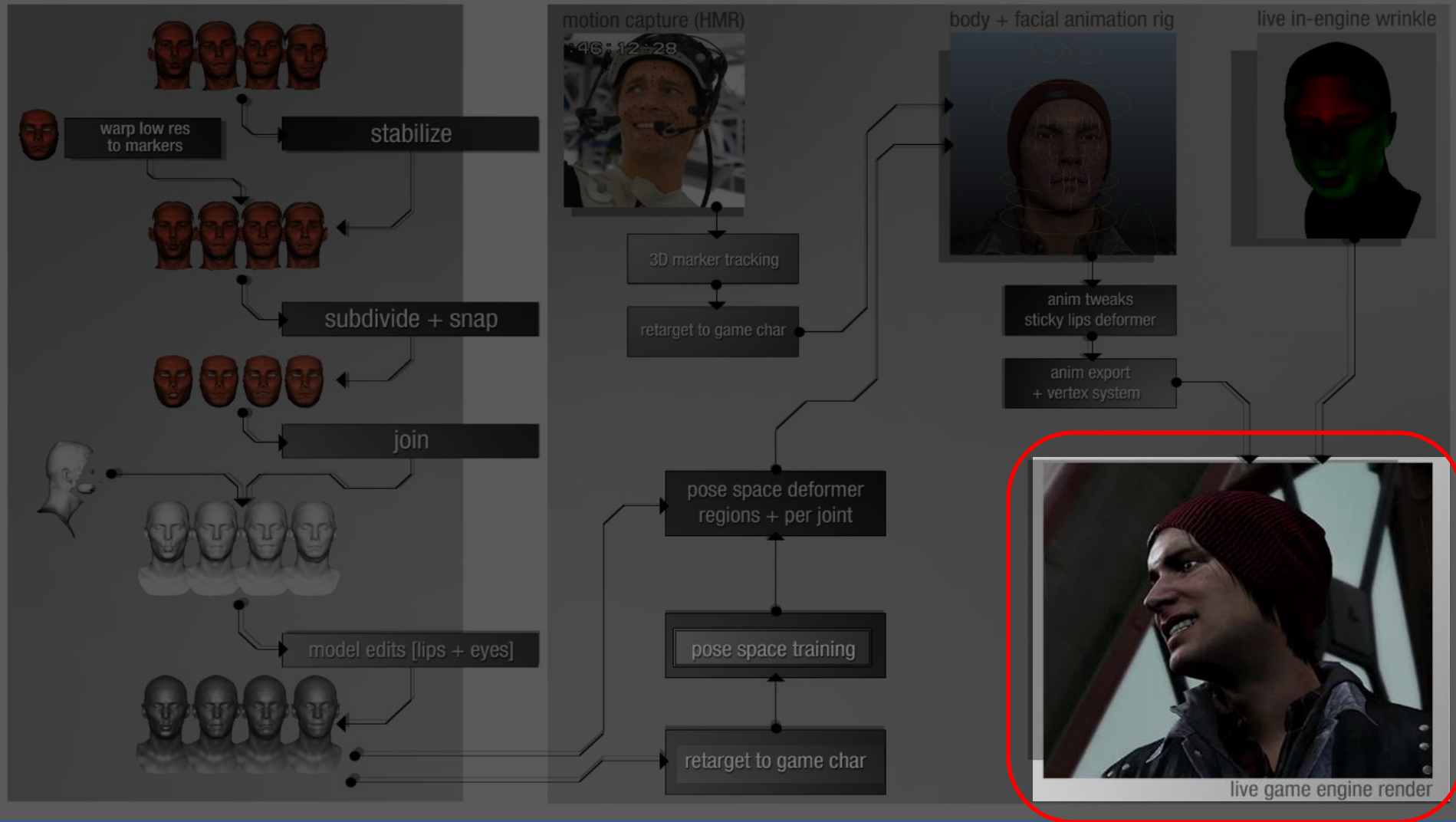
- Animator Attributes
  - RBF, input widths, Shape Influence, Drop Shapes
  - Per Joint control weight –From Pure Pose Space to exact 3d track
  - Additive versus Animation Over-takes Mocap automatically
  - Sticky Lips Attributes
  - Eye Drag Multiplier Attributes


warp low res to markers
stabilize
subdivide + snap
join
model edits [lips + eyes]
motion capture (HMR)
3D marker tracking
retarget to game char
body + facial animation rig
anim tweaks sticky lips deformer
anim export + vertex system
live in-engine wrinkle
pose space deformer regions + per joint
pose space training
retarget to game char
live game engine render

WRINKLES AND CREASES – TENSION MAPS

- Compression and Stretch – Independent across face
- Detects Local Volume changes relative to Default
- Low Res Cage associates connectivity of high res
  - Faster Calculation
  - Smoother Falloff
  - Art Direct which Areas to be Driven Together (cage shape)
- Created the Normal Maps using depth difference between Raw dense FACS scans and final resolution




warp low res to markers
stabilize
subdivide + snap
join
model edits [lips + eyes]
motion capture (HMR)
3D marker tracking
retarget to game char
body + facial animation rig
anim tweaks sticky lips deformer
anim export + vertex system
live in-engine wrinkle
pose space deformer regions + per joint
pose space training
retarget to game char
live game engine render

REAL TIME PERFORMANCE

- Deformer Applied Pre-SkinCluster – Better Compression with small range offsets
- Joint Animation as Animation L.O.D.
- GPU Decompression during streaming
- Normal Recalculation Required
- Seam Fixing (shared edges)
  - Snag GPU round-off versus Maya skin Weights
- In Engine Wrinkle/Crease – matches anim rig

ADDITIONAL CONSIDERATIONS

- Tracking Issues
  - Stabilization, Camera Calibration
  - Requires Fast Feedback loop applied to “actor-only Rig”, for iteration to verify track
- Retarget – some room for improvement
- Infer Jaw Motion/Teeth – aimed at center of two lower jaw markers-(not perfect so allow offset animation)
- Attention to deformations of neck where join PSDs

References

"Facial Action Coding System",
[Paul Ekman, Wallace V. Friesen, Joseph C. Hagar, 2002, 1978, Consulting Psychologists Press, Palo Alto]

"Pose Space Deformations: a unified approach to shape interpolation"
[J.P. Lewis, Matt Cordner, Nickson Fong, Siggraph 2000]

"Realtime Facial Animation with On-the-fly Correctives"
[Hao Li, Jihun Yu, Yutin Ye, Chris Bregler, Siggraph 2013, ACM 2013]

"Animating Blendshape Faces by Cross-Mapping Motion Capture Data"
Zhigang Deng/USC, Pei-Ying Chiang/USC, Pamela Fox/USC, Ulrich Neumann †/USC

WPSD- "Modeling Deformable Human Hands from Medical Images"
Tsuneya Kurihara1 and Natsuki Miyata2

"A Brief Introduction to Statistical Shape Analysis"
Mikkel B. Stegmann and David Delgado Gomez¤/Informatics and Mathematical Modelling, Technical University of Denmark
Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, Denmark

"Interactive Region-Based Linear 3D Face Models"
[J. Rafael Tena, Fernando De la Torre, Ian Mathews, Siggraph 2012]

"EM algorithms for pca and spca. Advances in Neural Information Processing Systems", MIT Press, 626–632
ROWEIS, S. 1998.

"Dense 3d motion capture for human faces. "
FURUKAWA, Y., AND PONCE, J. 2009.
In 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25
June 2009, Miami, Florida, USA, IEEE, 1674–1681

TECH CONTRIBUTORS

Co-Contributor

Adrian Bentley – Lead Engine and Render Team

Additional Contributions

Josh Jersild – Senior Programmer

Shuai Liu – Senior Programmer

Stephen White - Co-Lead Tools & Engine Team

SPECIAL THANKS!

THANK YOU!!

...And special Thanks to the GDC Committee!

...and GDC mentor Ru Weer

Questions?

email: spencera@suckerpunch.com
twitter: spensa3d