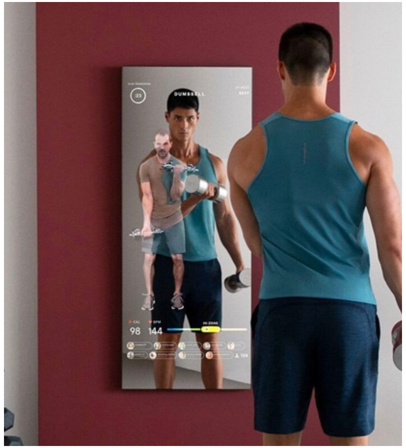
Falcon: the Pro Gym Assistant

Team Ao
Vishal Baskar
Albert Chen
Venkata Vivek Thallam

Use Case

- Advanced at-home workout system that provides:
  - Demonstration of exercises
  - Rep counter
  - Calorie estimator
  - Live stream of themselves
  - Ability to get customized workouts that comprise of the following:
    - Leg Raises, Pushups, Lunges
  - Real-time feedback regarding posture **(Unique to Falcon)**
- Workout system involves a display and a side camera
- Processing done on an FPGA to address privacy concerns
- Areas Covered:
  - Software Systems, Signals and Systems, Hardware Systems



Mirror: A popular at-home workout tool

Requirements (Setup)

- Only 1 user at a time
- Need camera at a side view with no excessive background interference
- Need user to wear multiple trackers of various colors (side facing camera)
    - Shoulder
    - Elbows
    - Wrists
    - Hip
    - Knees (2)
    - Ankles (2)
- Need user to wear simple colors (white, grey, black)
    - To prevent collisions with the trackers

Requirements (Signal Processing)

- Pre-processing of input image on computer
    - Scale down image from webcam down to a 160 x 120 pixel image
- Extracting key features
    - Computation time on FPGA:
        - Estimation: 30 cycles per pixel for a 160 by 120 pixel image = 576000 cycles ~= **11.52 ms** at a 50MHz clock
    - Accuracy of classification: 90%
        - The average set will have 10 reps, allowing for 1 rep to be misclassified.
- Challenges:
    - Background noise may affect accuracy of classification

Requirements (Hardware)

- Communication Protocol
  - Assuming bit rate of the UART channel is 921600 bits/second
  - From computer to FPGA:
    - Image is 160 x 120 pixels = 768,000 bits = **0.833 s**
  - From FPGA to computer:
    - 8 points = 180 bits = **0.2 ms**
- Challenges
  - Being able to successfully interface with the UART protocol
  - Keep response time low (since I/O is the bottleneck)

Requirements (Posture Analysis)

- Posture Analysis
    - Analyze existing models to determine thresholds
        - **Leg Raise:** Ensure angle between hip, knee, and ankle is straight. Hip to shoulder straight.
        - **Push Up**: Right Angle from shoulder to elbow to wrist
    - Accurately identify which joints are out of place, using joint location from FPGA
- Challenges
    - Fine tuning the thresholds

Requirements (Application)

- User Interface
  - User can select difficulty of workout, and body area they want to target
    - Legs, Core, Upper
  - Display recorded video of instructor doing exercise
  - Count Number of reps performed, calories burned based on user biodata
  - Live feedback
  - Variations between up and down positions in ~1.5 s.
    - Capture the static image
- Challenges
  - Live video streaming, playing recorded video and overlaying feedback efficiently

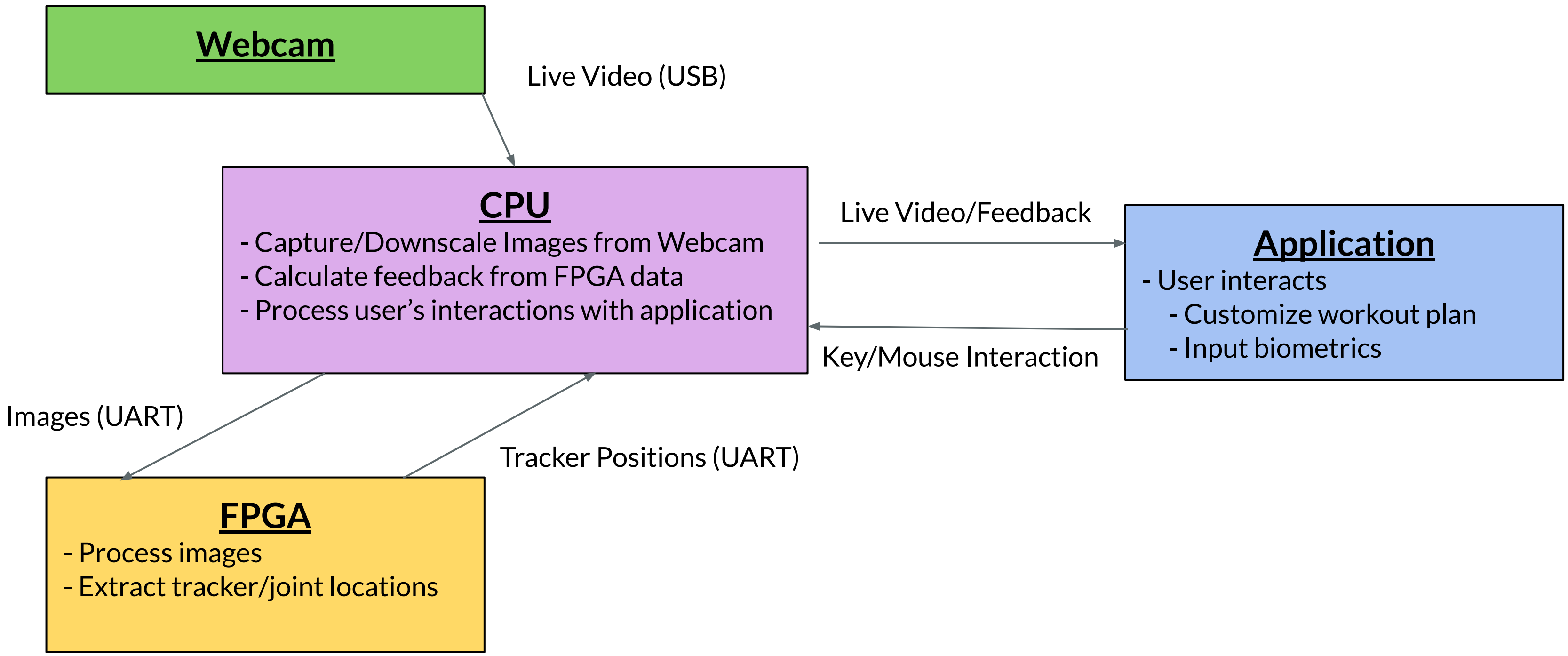
Solution Approach

Leg Raise
Exercise 3/10   5:30 Remaining

Rep: 1/10   320 Cal   150-180 BPM

Raise Legs Higher

Solution Approach



Testing, Verification and Metrics

Requirement	Testing Strategy	Metrics
Downscaling of image	Software testbench (analyze size and quality of resulting image)	100% size match
Detect trackers	Software testbench (analyze trackers over various images)	1 rep to be misclassified every set
Communication between computer and FPGA	Hardware testbench (analyze various packets of data sent)	Latency < 1s & 100% data accuracy
Posture Analysis	Software testbench (analyze various positions to extract info)	100% accuracy according to our designed models
UI (Workout Data + Feedback)	Human Eye (analyze the metrics are met from what is done)	Workout Data: ~tracker accuracy Feedback Delay < 1.5 secs

Tasks and Division of Labor

- Signal Processing
  - Pre-Processing to downscale the image (Python) -> **Albert**
  - Learn and implement algorithm to extract trackers and determine joints (Python) -> **Albert**
  - Determine thresholds to provide feedback regarding posture given joint locations -> **Albert/Venkata**
  - Implement posture analysis from the joint positions for feedback (Python) -> **Albert/Venkata**
- Hardware
  - Convert software algorithm to optimized synthesizable FPGA code (SystemVerilog) -> **Venkata/Albert**
  - Learn how to communicate efficiently between the FPGA and CPU and vice-versa via UART -> **Venkata**
- Software
  - Design the computer application interface (PyGame) -> **Vishal**
  - Integrate live video feed and capture image -> **Vishal**
  - Integrate recorded exercises and create timed workout with feedback -> **Vishal**

Schedule

Falcon: the Gym Pro Assistant

Tasks	Week 1 8/31	Week 2 9/7	Week 3 9/14	Week 4 9/21	Week 5 9/28	Week 6 10/5	Week 7 10/12	Week 8 10/19	Week 9 10/26	Week 10 11/2	Week 11 11/9	Week 12 11/16	Week 13 11/23	Week 14 11/30	Week 15 12/7	Week 16 12/14
Milestones	Project Planning			Design Implementation						Integration and Verification				Project Report and Presentation		
Signal Processing																
Pre-processing image to downscale image																
Learn algorithm to extract joints																
Implementing algorithm to extract joints																
Determine thresholds to determine posture																
Implementing Posture Analysis																
Hardware																
Convert joints algorithm to FPGA code																
Optimize algorithm for better performance																
Learn communication protocol (UART)																
Implement communication protocol logic																
Software + UI																
Design the UI Model																
Setup PyGame (Basic Framework)																
Implement Camera Display + Capture Image																
Integrate Recorded Model Exercise																
Customize User Biodata and Create timed workout																
Keep Track of Data and Output Feedback																
Refining Implementation																
Extraneous Setup																
Create the Trackers																
Integration + Final Verification																
Verification of individual parts																
Integrate + Verify I/O with Image Processing																
Integrate + Verify processing output with UI																
Refining App and Integration																
Proposal/Report/Presentation																
Project Ideas																
Abstract			DUE													
Project Proposal				DUE												
Design Presentation							DUE									
Design Report								DUE								
Demo in Lab											DUE					
In Lab Demo														DUE		
Final Presentation															DUE	
Final Report																DUE

- Vishal
- Venkata
- Albert
- Everyone