DejaVu:

Integrated Support for Developing Interactive Camera-based Programs

JUN KATO[1,2], SEAN MCDIRMID[1], XIANG CAO[1]

MICROSOFT RESEARCH ASIA[1]

IGARASHI LAB., THE UNIVERSITY OF TOKYO[2]

Microsoft Research

東京大学 THE UNIVERSITY OF TOKYO

BACKGROUND

Camera-based programs are getting popular
with affordable hardware and useful software libraries.

KINECT™ for Windows®

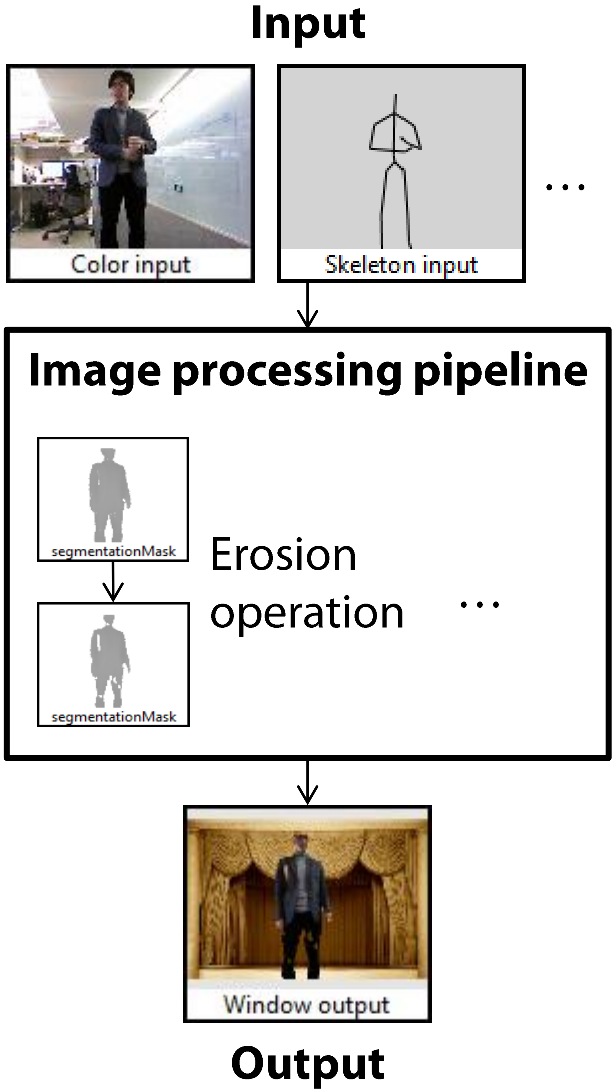
OpenCV

PROBLEMS

Various visual data

Continuous processing

Non-reproducible input



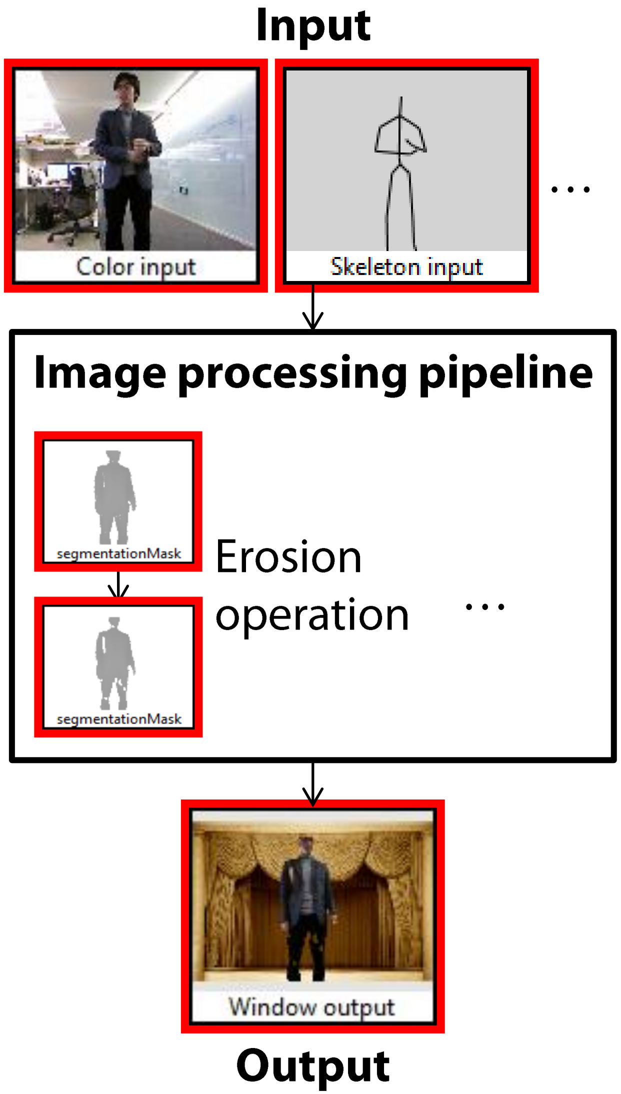
PROBLEMS

Various visual data

- Camera input
- Intermediate results
- Window output…

Current IDEs:

- Textual value only



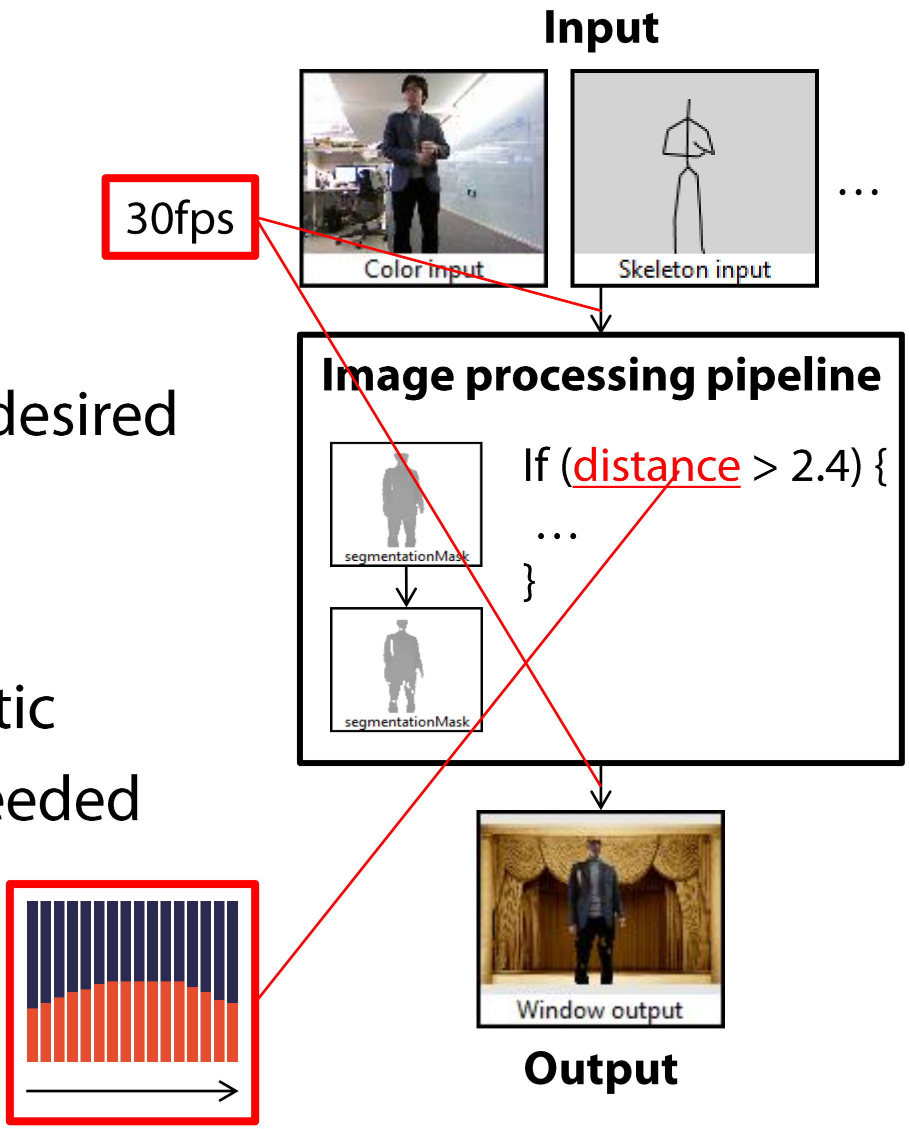
PROBLEMS

Continuous processing

- Frame-by-frame nature
- Temporal visualization desired

Current IDEs:

- Breakpoint is problematic
- Custom visualization needed



Input

Image processing pipeline

```
If (distance > 2.4) {
 …
}
```

Output

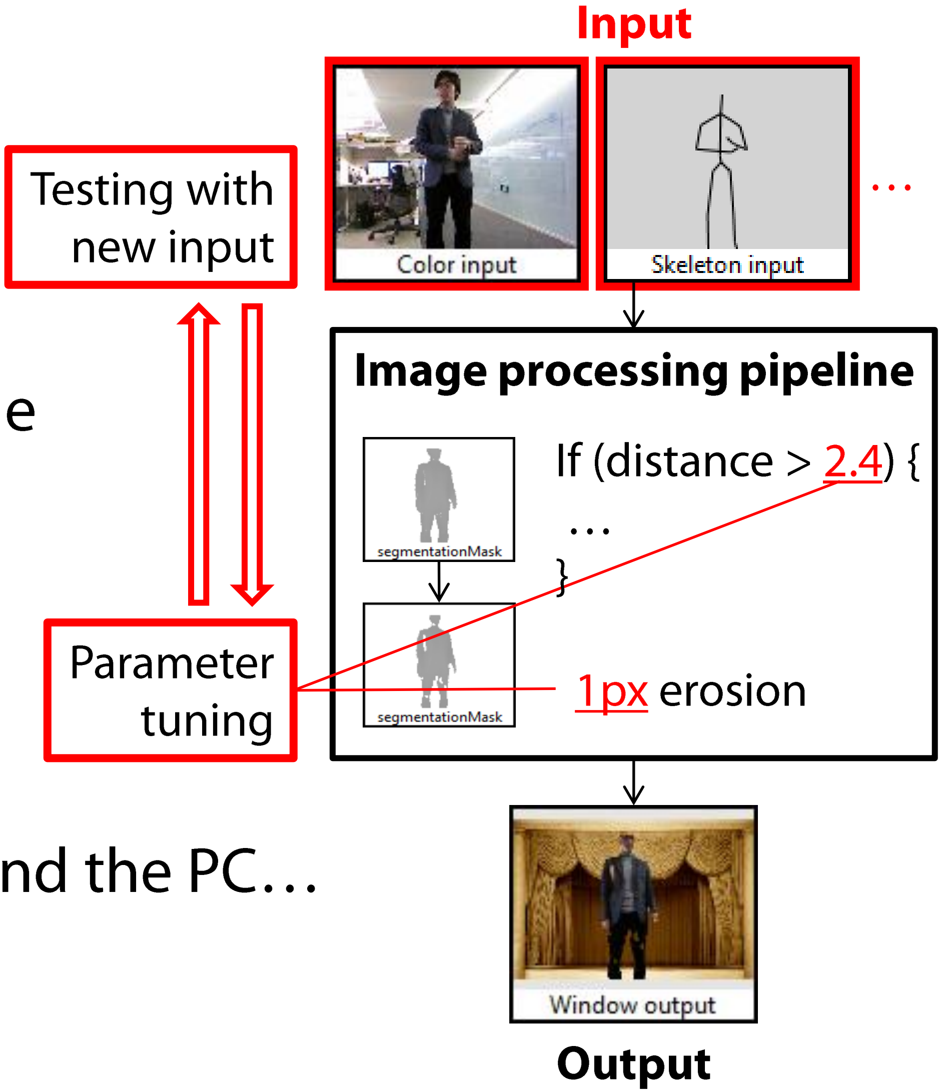
PROBLEMS

Non-reproducible input

- Iterative test process
- Same action impossible
- Environmental noise

Current IDEs:

- Just go back and forth
  between the camera and the PC…



Input

Testing with new input

Parameter tuning

Image processing pipeline

If (distance > 2.4) {
 …
}

1px erosion

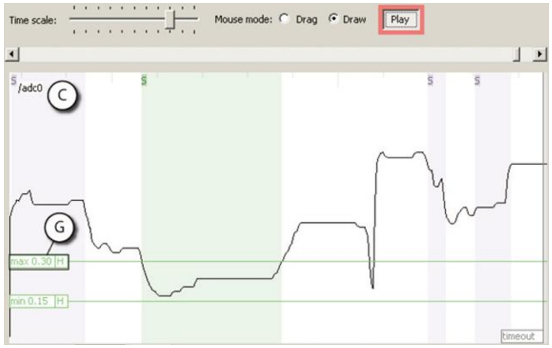
Output

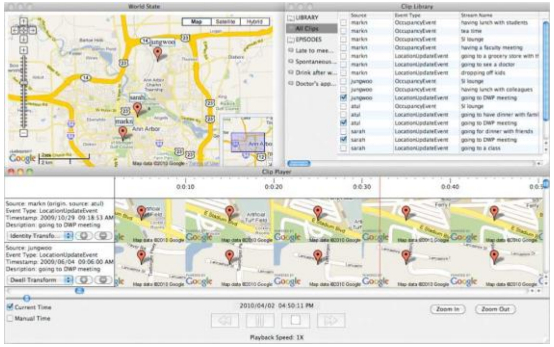
CONTRIBUTION

Enhancement to an IDE for general development of the interactive camera-based program.

RELATED WORK: PROTOTYPING TOOLS

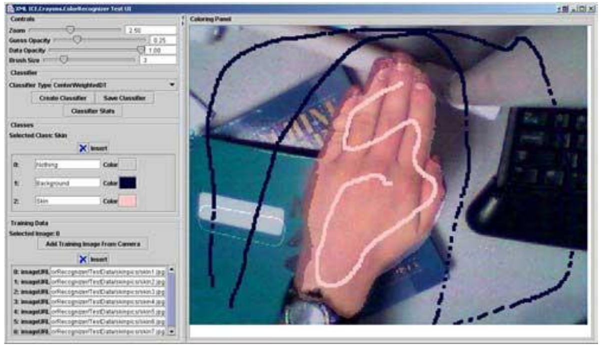
Exemplar, RePlay: record & replay of sensor input

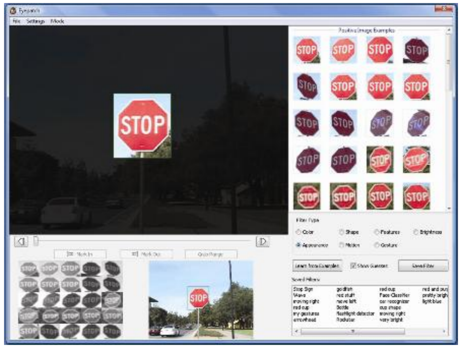




[Hartmann et al., CHI'07]
[Newman et al., UIST'10]

Crayon, Eyepatch: external tools for computer vision

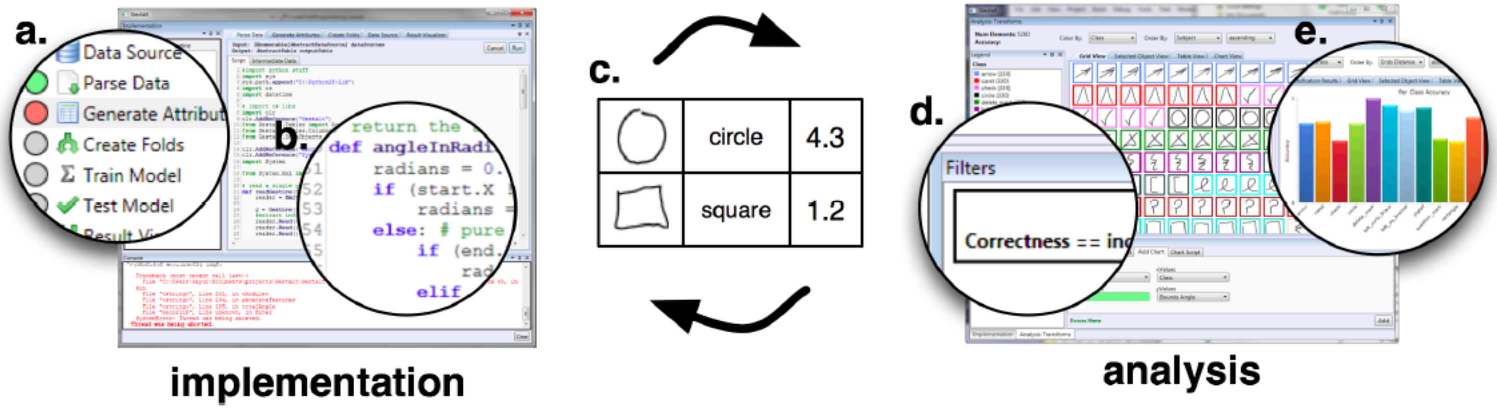




[Fails et al., CHI'03]
[Maynes et al., UIST'07]

RELATED WORK: GESTALT

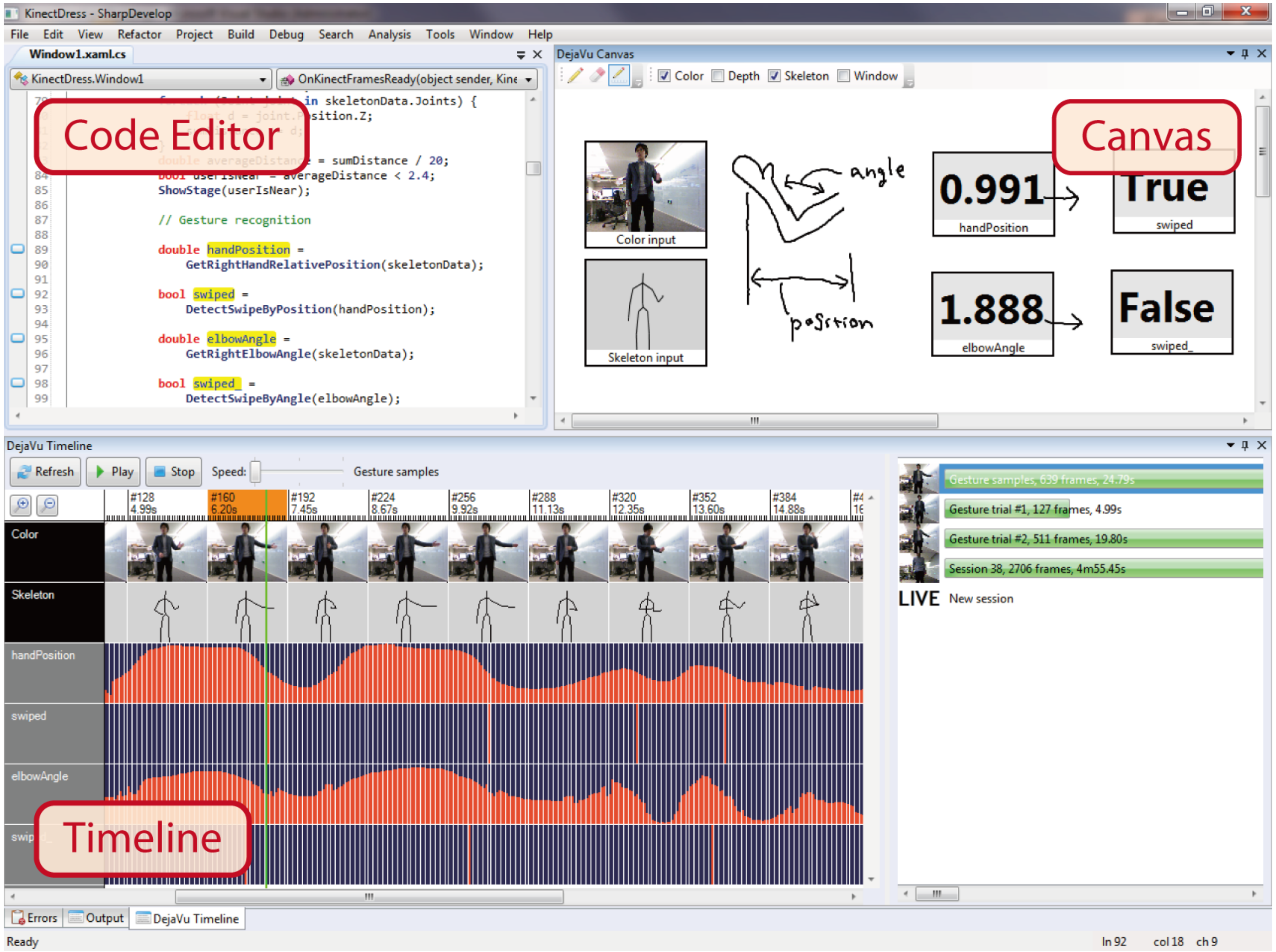
Programming for machine learning [Patel et al., UIST'10]



```
return the
def angleInRadi
    radians = 0.
    if (start.X
        radians =
    else: # pure
        if (end.
            rad
        elif
```

	circle	4.3
	square	1.2

DEJAVU INTERFACE

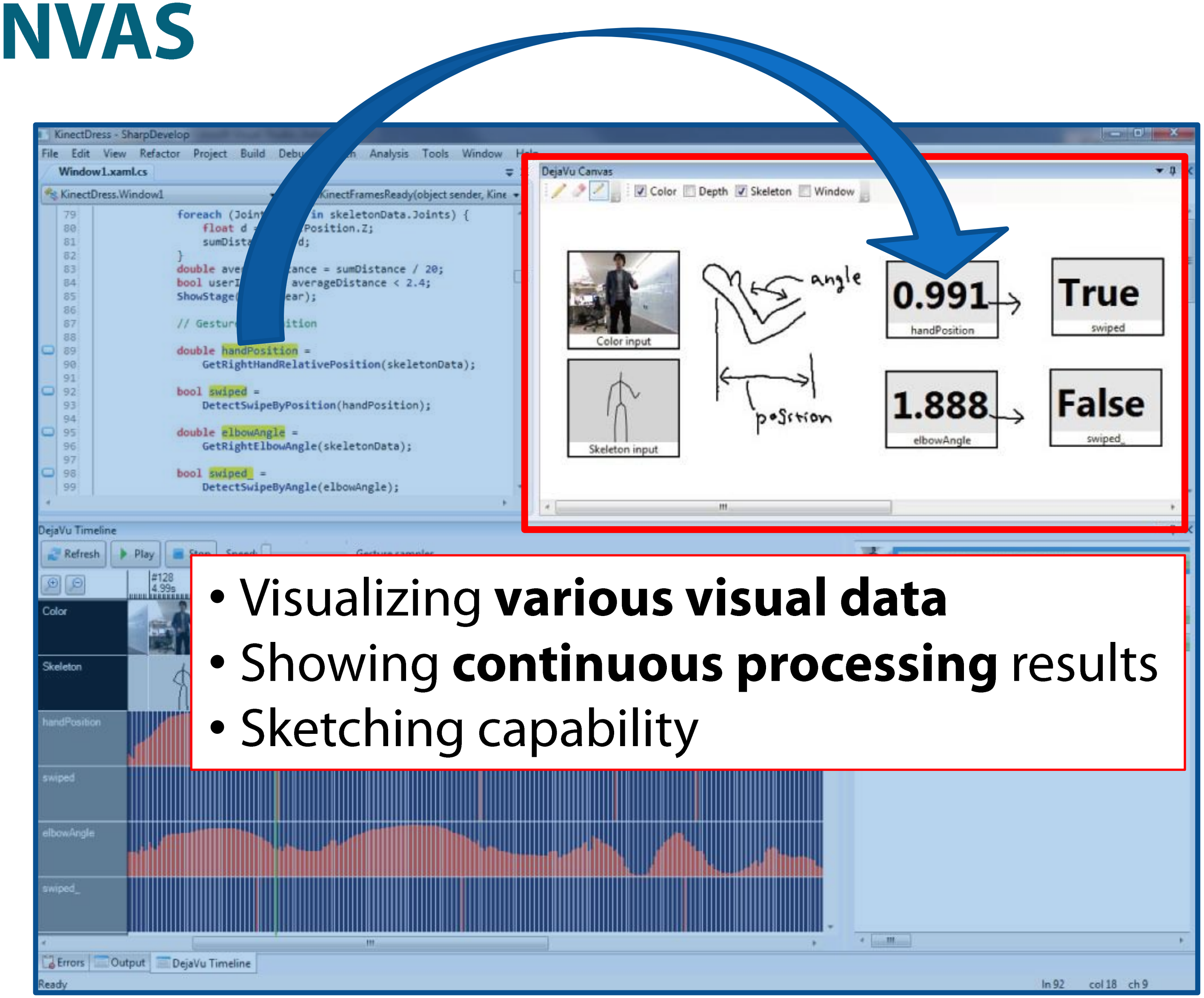


```
foreach (Joint joint in skeletonData.Joints) {
    float d = joint.Position.Z;
    sumDistance += d;
}
double averageDistance = sumDistance / 20;
bool userIsNear = averageDistance < 2.4;
ShowStage(userIsNear);

// Gesture recognition

double handPosition =
    GetRightHandRelativePosition(skeletonData);

bool swiped =
    DetectSwipeByPosition(handPosition);

double elbowAngle =
    GetRightElbowAngle(skeletonData);

bool swiped_ =
    DetectSwipeByAngle(elbowAngle);
```

Gesture samples, 639 frames, 24.79s
Gesture trial #1, 127 frames, 4.99s
Gesture trial #2, 511 frames, 19.80s
Session 38, 2706 frames, 4m55.45s
LIVE New session

Ready   ln 92   col 18   ch 9

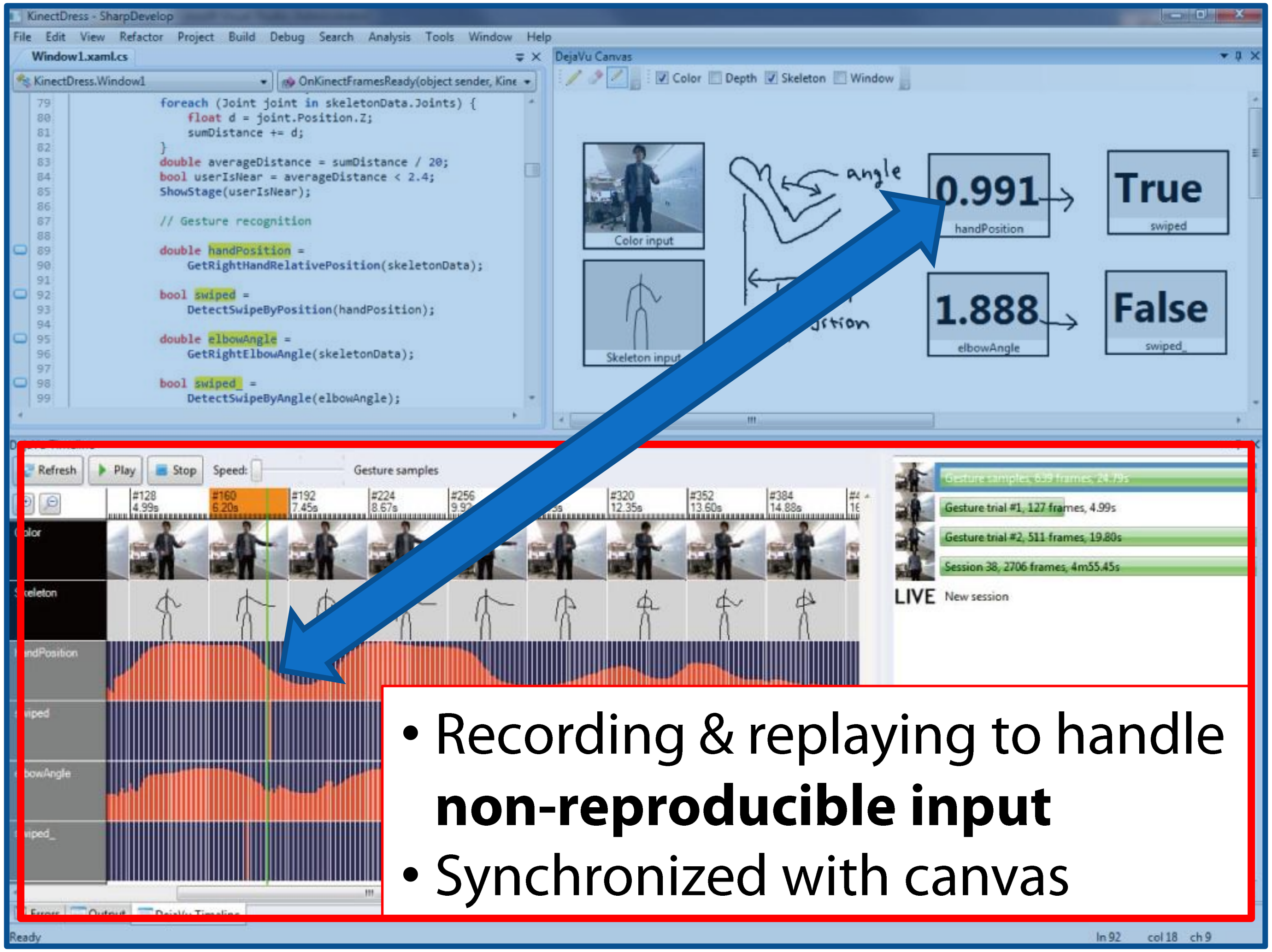
DEJAVU INTERFACE: CANVAS



```
            foreach (Joint     in skeletonData.Joints) {
                float d =     .Position.Z;
                sumDista     d;
            }
            double ave     ance = sumDistance / 20;
            bool userI     averageDistance < 2.4;
            ShowStage(     ear);

            // Gestur     ition

            double handPosition =
                GetRightHandRelativePosition(skeletonData);

            bool swiped =
                DetectSwipeByPosition(handPosition);

            double elbowAngle =
                GetRightElbowAngle(skeletonData);

            bool swiped_ =
                DetectSwipeByAngle(elbowAngle);
```

- Visualizing **various visual data**
- Showing **continuous processing** results
- Sketching capability

DEJAVU INTERFACE: TIMELINE



```
foreach (Joint joint in skeletonData.Joints) {
    float d = joint.Position.Z;
    sumDistance += d;
}
double averageDistance = sumDistance / 20;
bool userIsNear = averageDistance < 2.4;
ShowStage(userIsNear);

// Gesture recognition

double handPosition =
    GetRightHandRelativePosition(skeletonData);

bool swiped =
    DetectSwipeByPosition(handPosition);

double elbowAngle =
    GetRightElbowAngle(skeletonData);

bool swiped_ =
    DetectSwipeByAngle(elbowAngle);
```

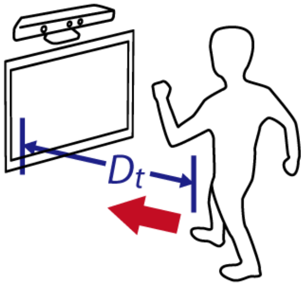
- Recording & replaying to handle **non-reproducible input**
- Synchronized with canvas

DEMONSTRATION

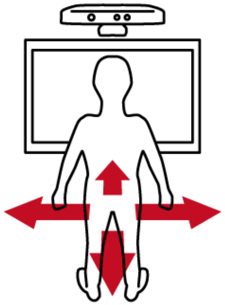
KinectDress: virtual dressing room application

KinectDress

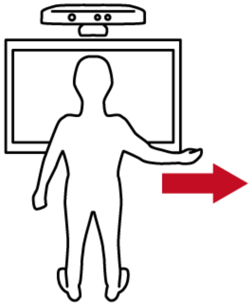
KinectDress



Go close
to start



Move
to try-on



Swipe
to change clothing

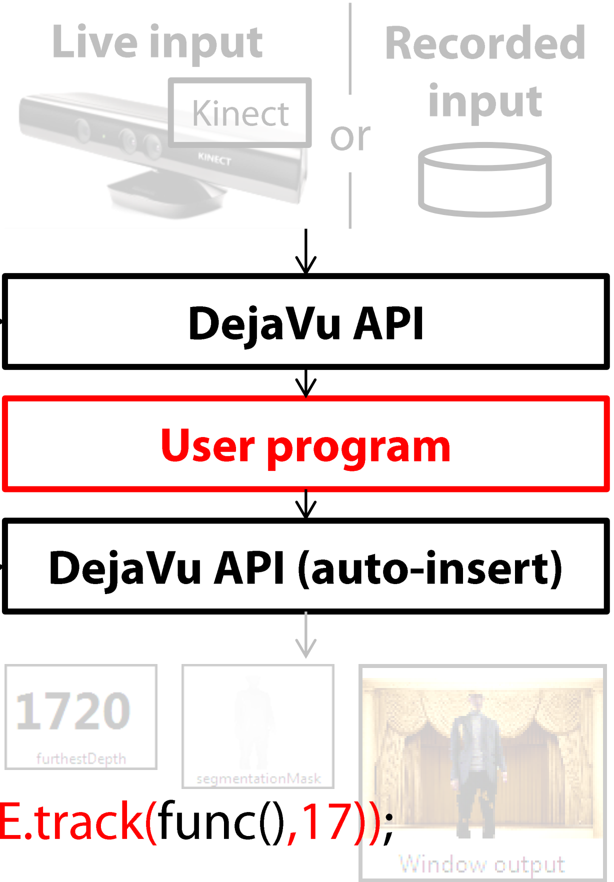
IMPLEMENTATION

Extension to SharpDevelop IDE

**Thin-wrapper of
Kinect for Windows SDK**

**Silently inserting code
to track variables**

Example) Original: double a = func();
Rewritten: double a = (Double)(IDE.track(func(),17));



Live input

Kinect

or

Recorded input

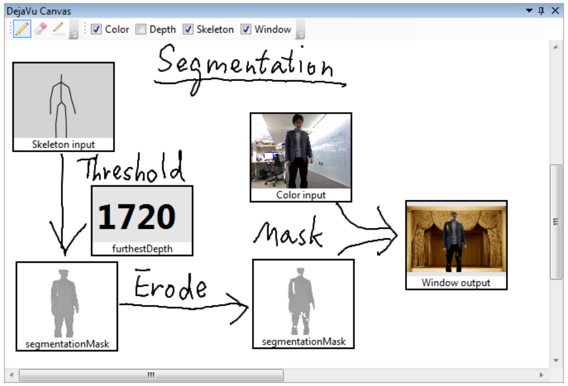
DejaVu API

User program

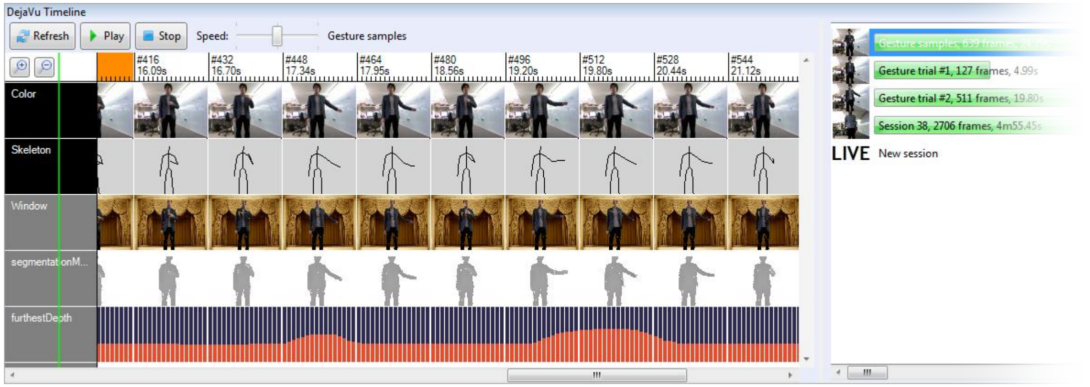
DejaVu API (auto-insert)

CONCLUSION

- **DejaVu** provides enhanced **integrated support** for **interactive camera-based programs**.
  - **Canvas**: visualization of current situation + sketch
  - **Timeline**: visualization in temporal fashion + replay
- We received positive feedback from target users.



Canvas



Timeline

APPENDIX

EXECUTION MODES

Live execution

Replay

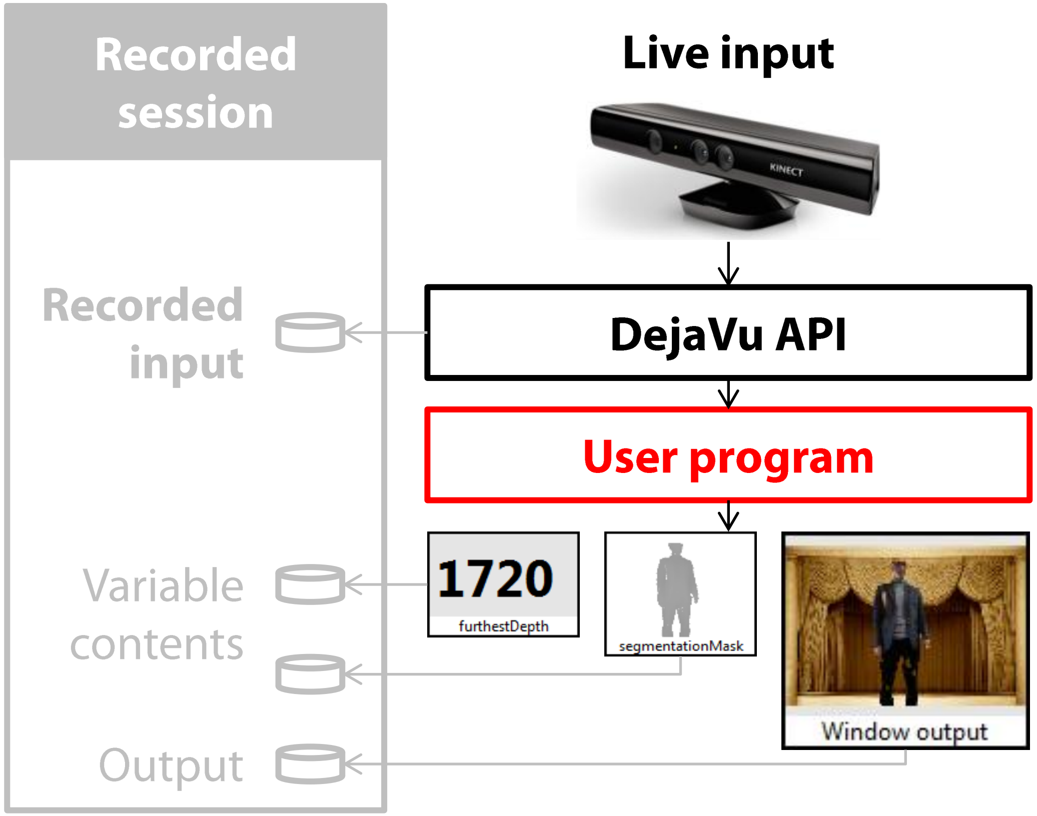
Refresh

EXECUTION MODES

Live execution

Replay

Refresh



Recorded session

Recorded input

Variable contents

Output

Live input

DejaVu API

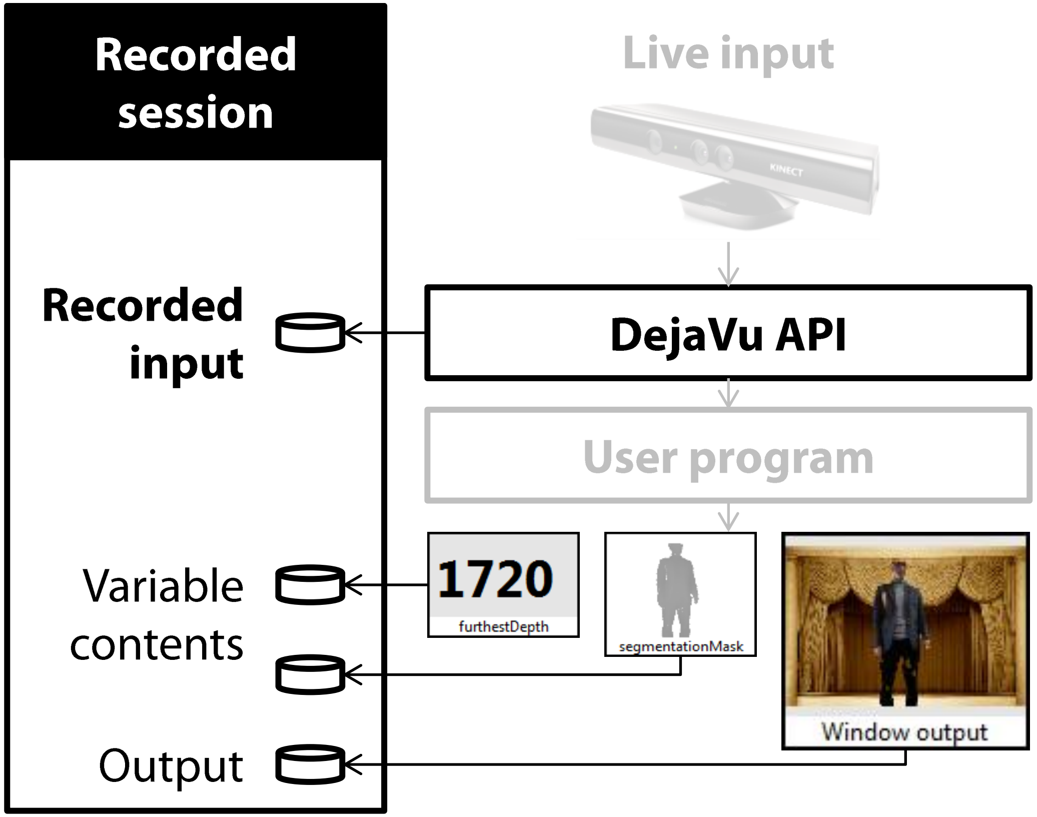
User program

EXECUTION MODES

Live execution

Replay

Refresh

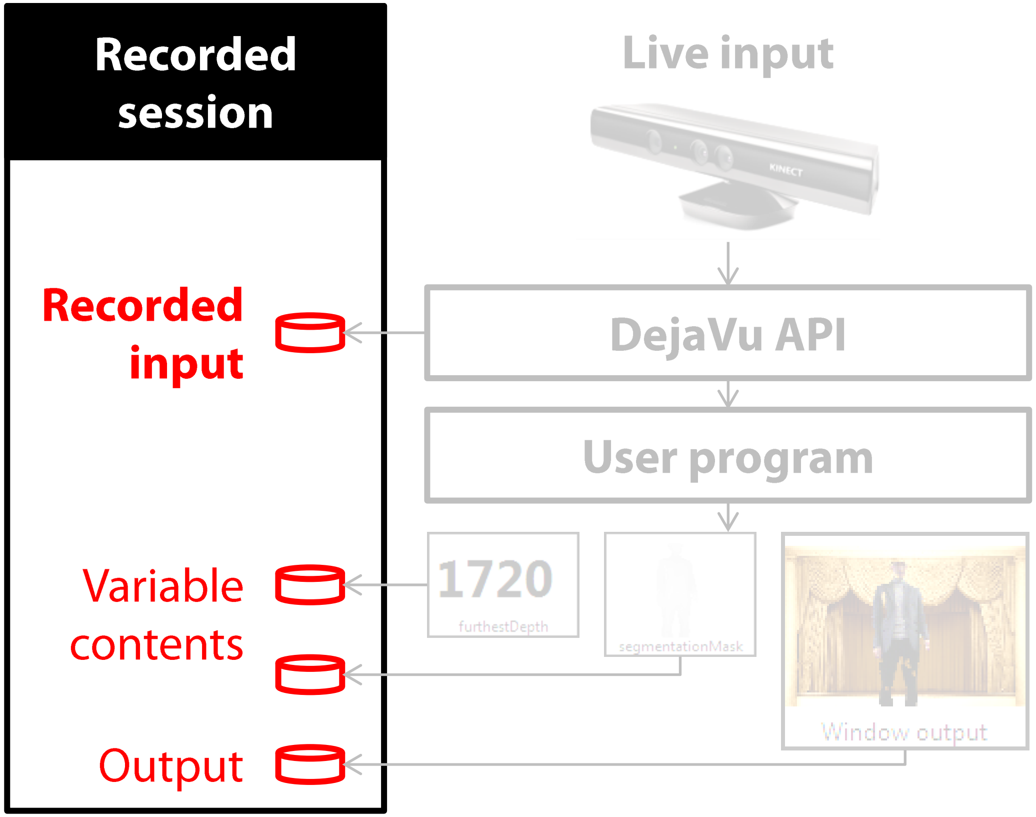


EXECUTION MODES

Live execution

Replay

Refresh



Recorded session

Recorded input

Variable contents

Output

Live input

DejaVu API

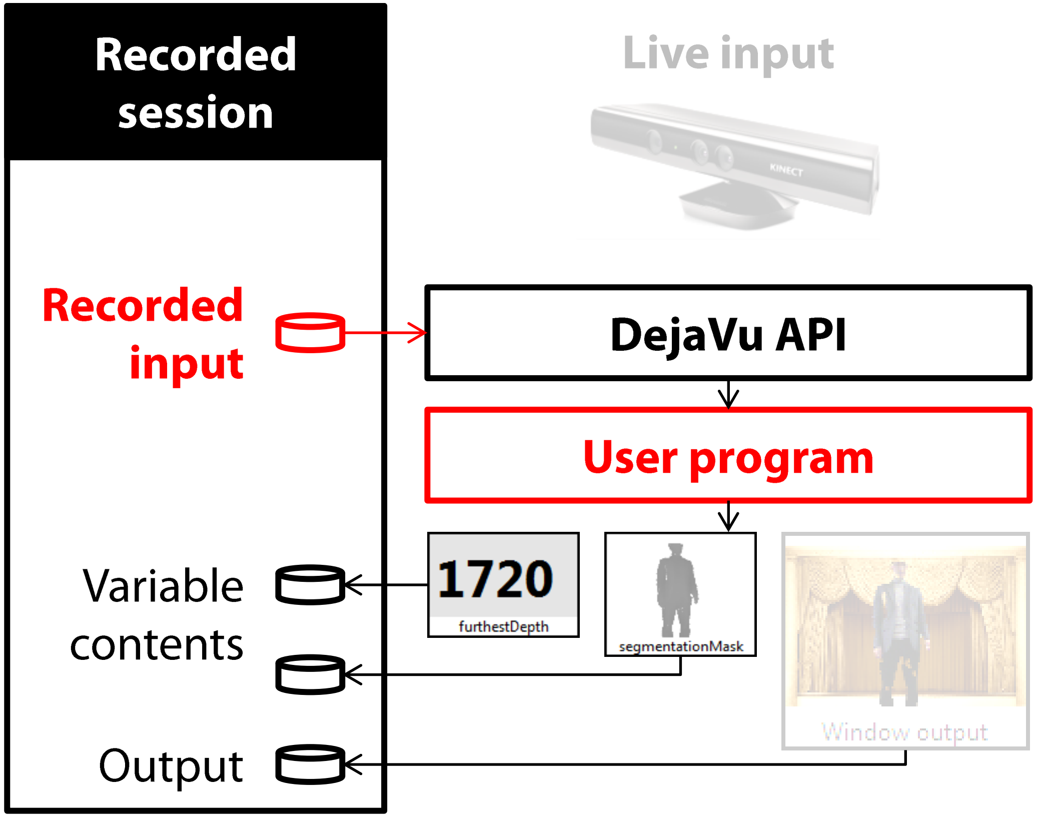
User program

EXECUTION MODES

Live execution

Replay

Refresh



USER FEEDBACK

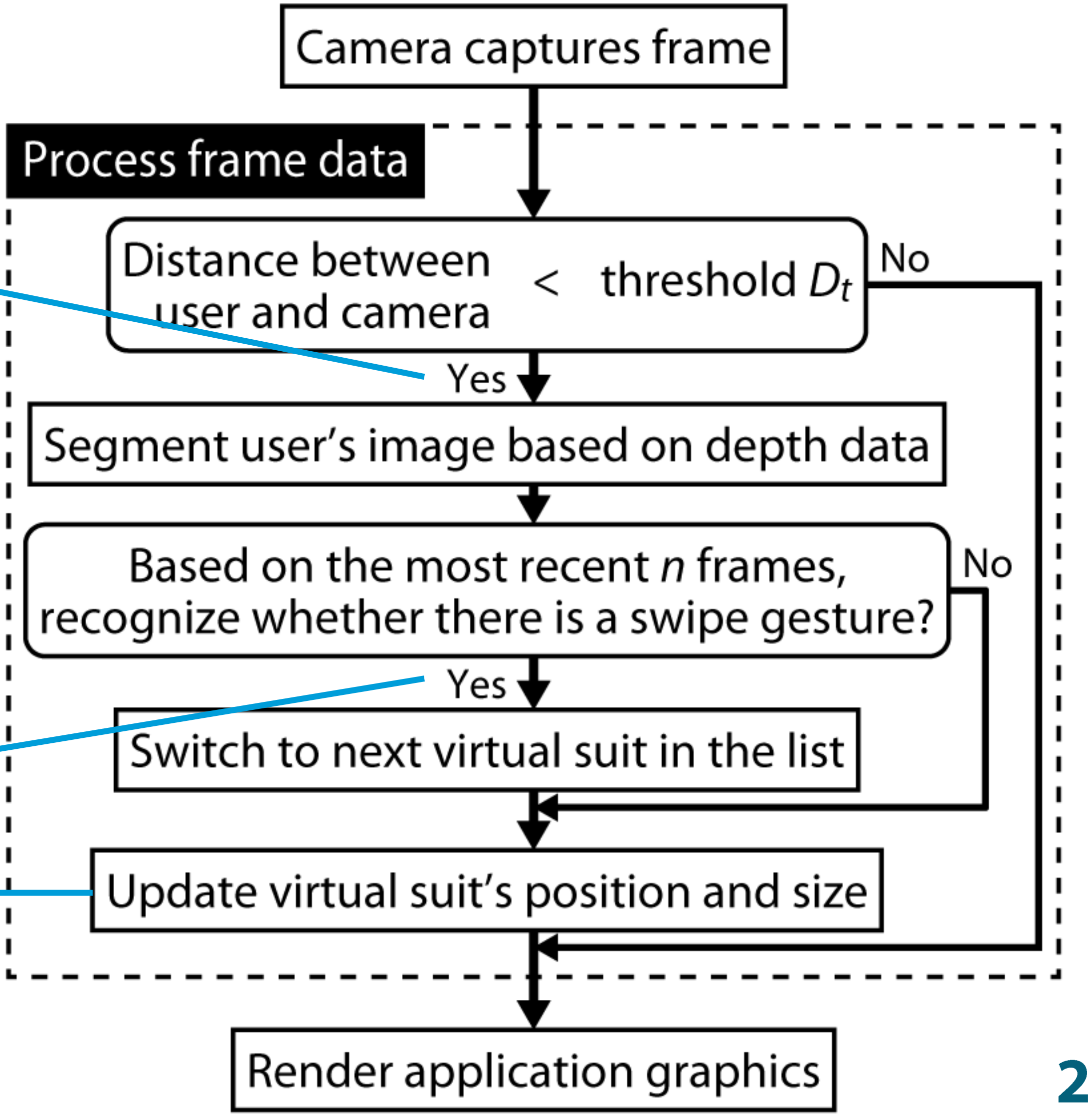
3 experienced programmers

1 hour trial + informal interview

Positive comments and suggestions

- Simulating and manipulating input
- Visualizing generic arrays
- Composite visualization

EXAMPLE APPLICATION (KINECTDRESS)



Camera captures frame

Process frame data

Distance between user and camera < threshold D_t

No

Yes

Segment user's image based on depth data

Based on the most recent *n* frames, recognize whether there is a swipe gesture?

No

Yes

Switch to next virtual suit in the list

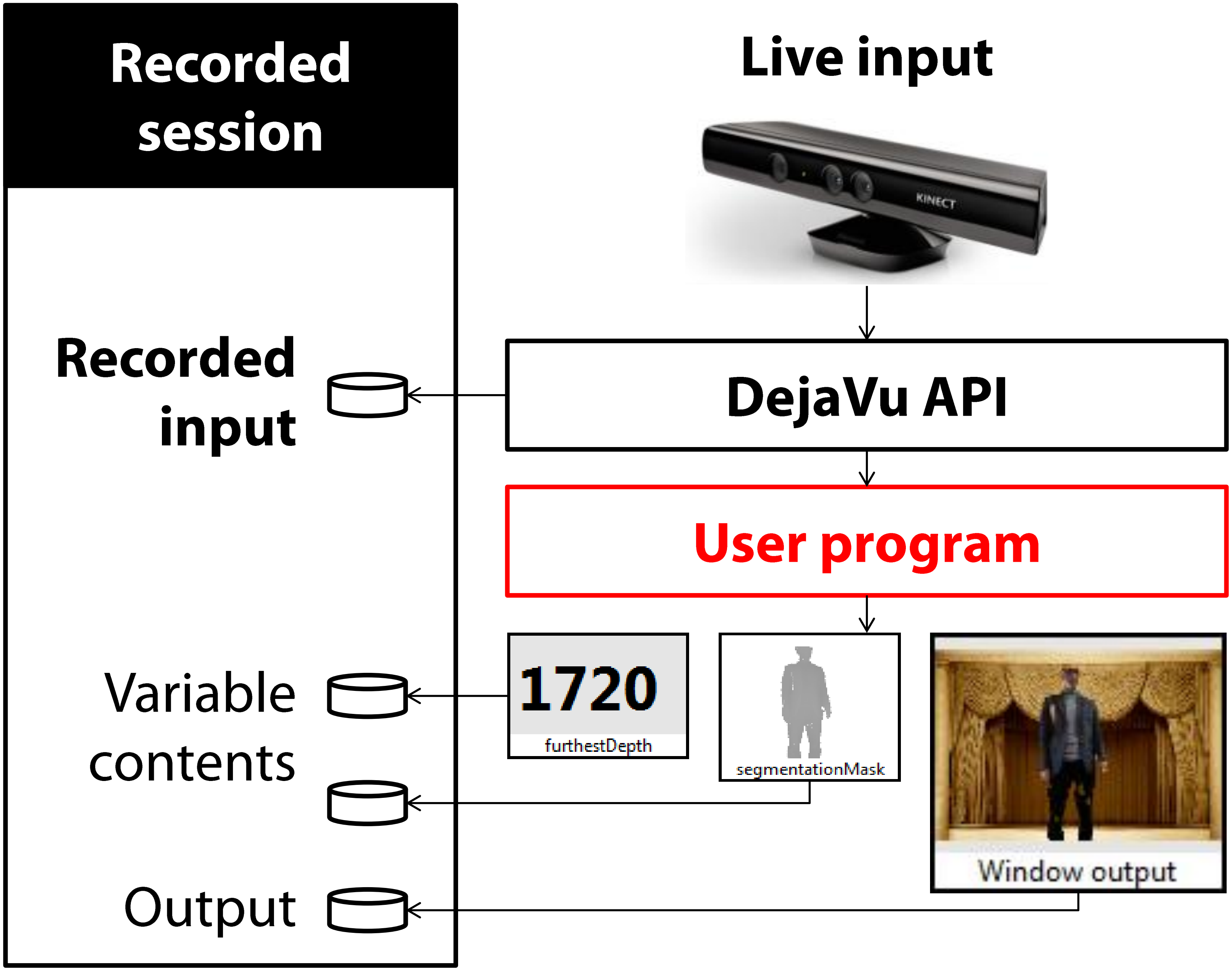
Update virtual suit's position and size

Render application graphics

Go close to start

Swipe to change cloths

Move to see fitting



Recorded session

Live input

Recorded input

DejaVu API

User program

Variable contents

Output