Tracking people in space for augmented reality

The data we want :

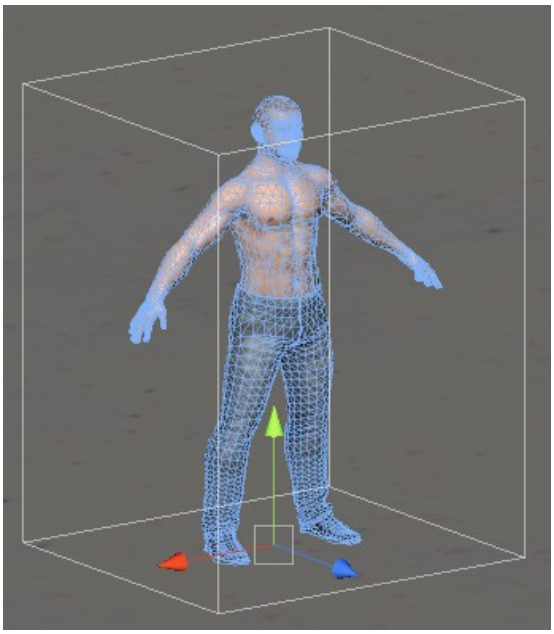
- depth : How far they are from the camera

In a rotated cartesian coordinates system with axis parallel to the ground (Top view)

- 3D bounding box
- centroid (center of mass)
- centroid projected on the ground
- highest point projected on the ground

The projection of the ground is useful in the particular case when lowest part of the bounding box does not touch the ground.

We also need this data normalized, for an easy inter process communication.



3D bounding box example

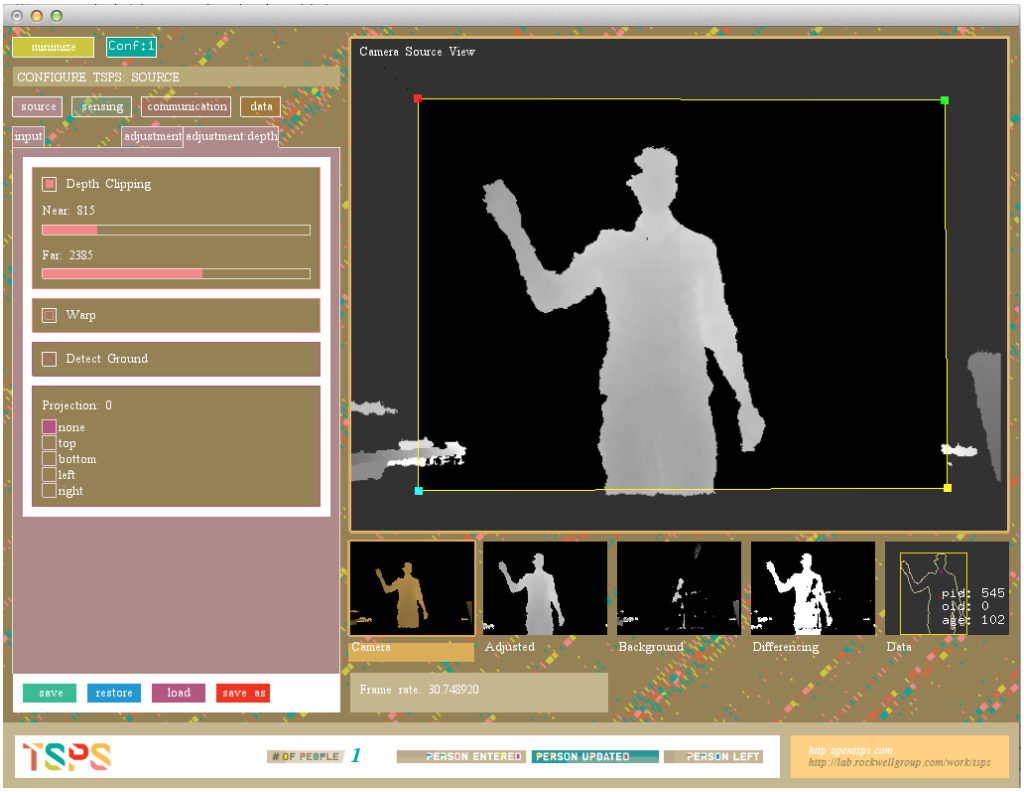
OpenTSPS fork depth video processing pipe dataflow

Step 1 : Choosing a 3D ROI (Region of interest)

We want to choose a ROI because :

- We want to be sure not avoid having false positive detection outside our ROI
- We want to ignore some part of the scene
- The detection pipe is implemented using 8bits buffers (for now) and depth precision is 11bits. By clipping the value in depth and remapping to an 8bit scale, we loose less precision.

Here we did set a 2D ROI with the same aspect ratio than the input image, the resulted ROI is upscaled and then processed in the next step.



To set the ROI in depth, we tick the *Depth Clipping* box, and set the near and far clipping values.

Theses values are depth values in mm. After clipping the values, the values are remapped to an 8 bit scale, and displayed.

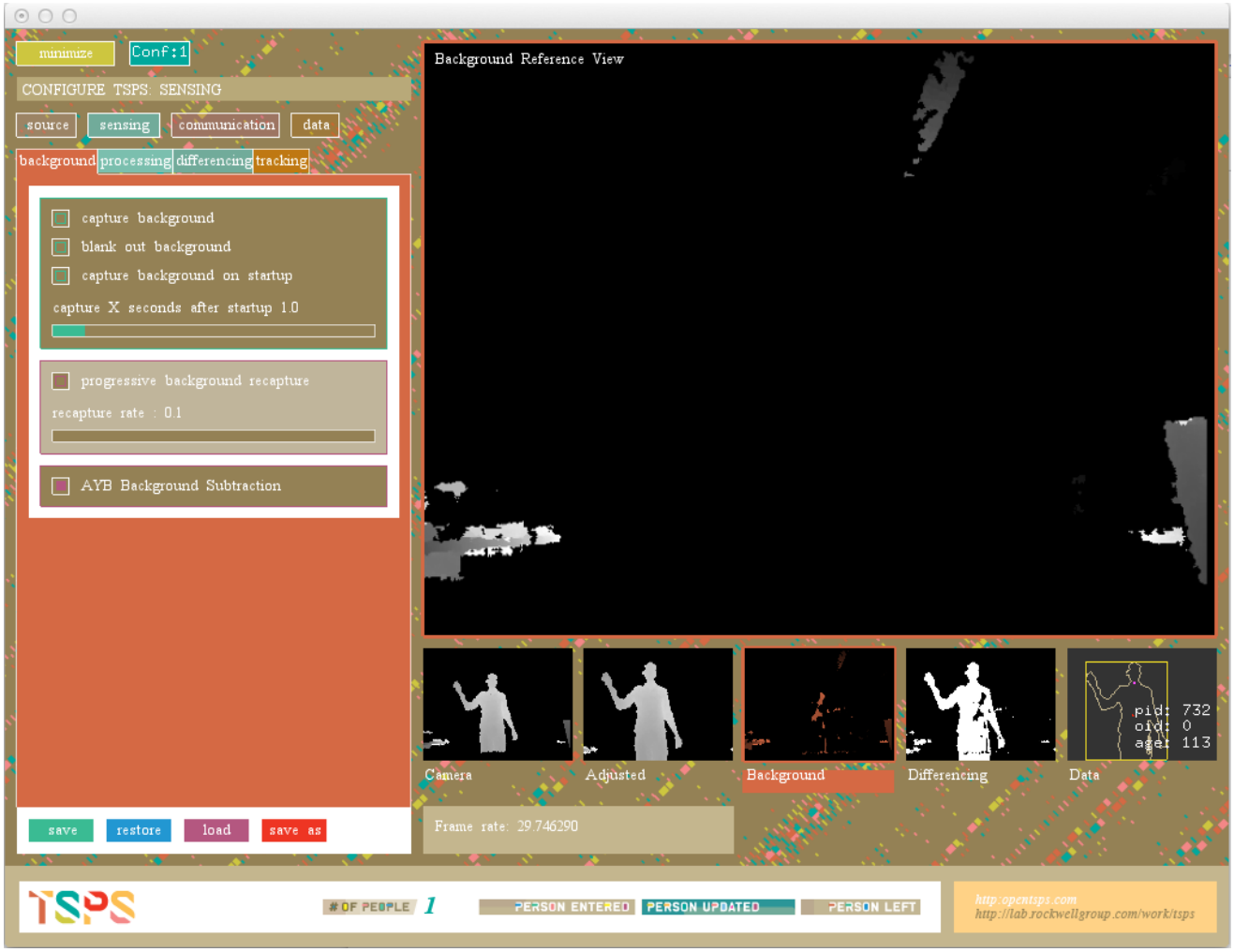
The effect in this example is that some of the background has disappeared because of the clipping, and we see more details on the person because of the remapped values.

In this step we can also detect the ground by ticking the box.

Step 2 : Background extraction

There is several methods to extract the background. We are using the automatic background extraction, which does not need calibration step.

This algorithms does constantly update the background image with the furthest value, working on the 16bits buffer and remapping the value after.



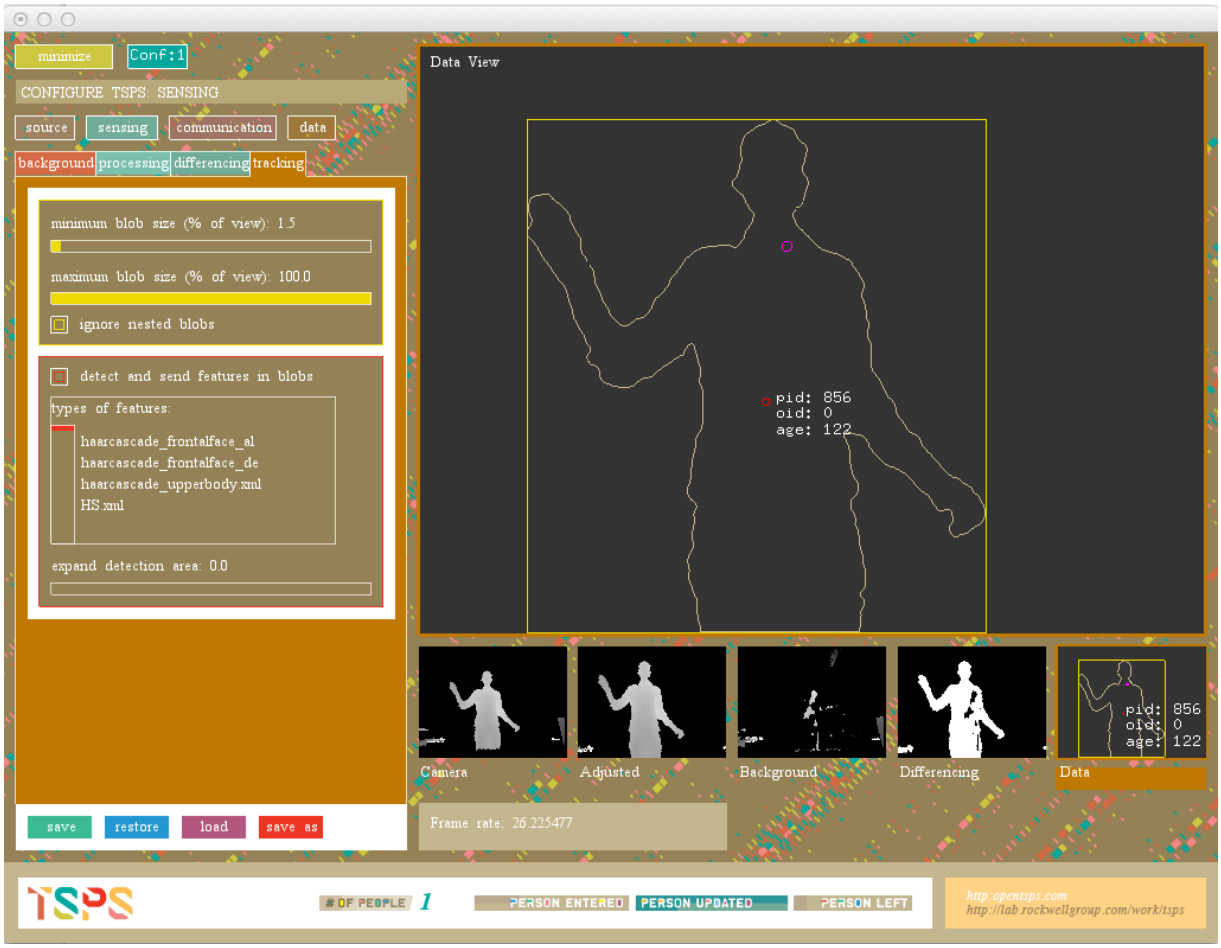
Step 3 : Differencing

Now we substract the adjusted image with the background and then apply a thresholding.

This step does not contain specific depth processing. We are smoothing the regions to fill some holes. Some morphological operators would be also useful in this step for depth image.

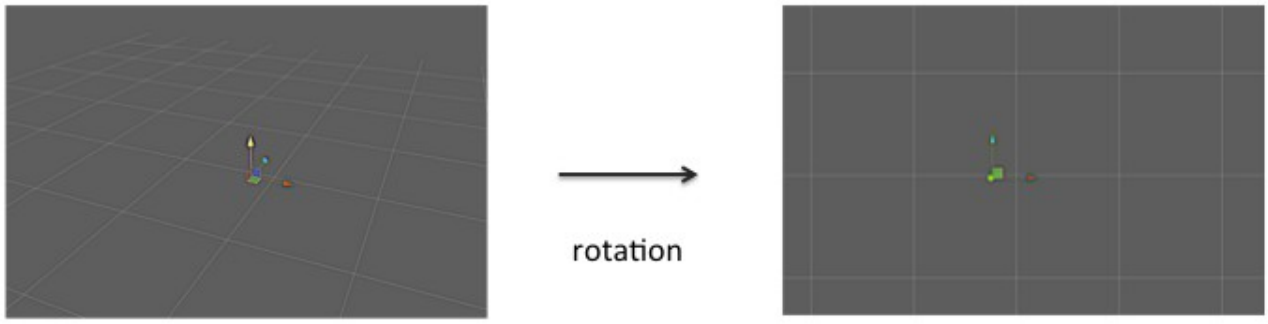
Step 4 : Blob detection and tracking

The blob detection and tracking is not specifically depth oriented. We now have to extract all the parameters that we want, providing that we know the ground equation.



Step 5 : Projection and rotated data extraction

We need to rotate our coordinate system to the ground detected at step 1 to achieve a top view.



Ground detection

Top view

Step-by-step process :

- Find ground equation
- Find the correct new coordinate system (or) rotation matrix
- Iterate over the blobs
- Convert each blob to a point cloud (using absolute coordinates before converting)
- Project on the new coordinate system (or) apply rotation matrix
- Compute bounding rect of the point cloud
- Compute centroid
- Project centroid to the ground
- Project highest point to the ground
- Normalize values