Capturing Hair Assemblies Fiber by Fiber

Pramook Khungurn

December 11, 2019

This is the "untangled" version of "Capturing Hair Assemblies Fiber by Fiber" by Wenzel, John Moon, and Steve. I wrote this because I was so confused when I read the article.

1 Measurement

- The hair assembly is mounted on a turntable whose rotation axis is vertical.
  The hair assembly is hung vertically, and the volume of hair is approximately centered around the center of rotation.
- The camera is mounted on a linear translation stage.
  The direction of translation is parallel to the optical axis of the camera.
  Moreover, the optical axis is perpendicular to the rotation axis of the turntable.
- The lens used was Cannon 100mm $f/2.8$ used.
  The aperture was opened fully to create small depth of field.
  The paper reported the depth of field was about 1.2mm.
- There's a blue background behind the hair.
  As the background is blue, the paper uses only the red pixels from Bayer pattern to do hair reconstrution.
- The system rotates the table around its axis, simulating the camera movement around the hair.
  For each rotation angle of the table, the system translates the camera
  along the translation. For each distance along the translation stage, it takes a photograph of the hair assembly.

2 Interpreting the Data

- The positions in the hair volume is described in coordinate system (x, y, z) such that the y-axis is the axis of the rotation. The coordinate system moves with the hair as the table is rotated.
- The focal plane is parameterized by coordinate system (u, v) where the v-axis is vertical and coincides with the axis of rotation of the turntable.
- Let d be the distance from the camera's translation from the position where the focal plane passes the rotation axis. We have that u, v, and d constitute another coordinate system for 3D points.
- For any point in the hair volume, it's (x, y, z) coordinate is related to its (u, v, d) coordinate by a simple rotation around the y-axis. That is:

$$\begin{bmatrix} u \\ d \end{bmatrix} = R(\theta) \begin{bmatrix} x \\ z \end{bmatrix}$$

where θ is the angle of rotation of the turntable, and $R(\theta)$ is the corresponding rotation matrix. Since the rotation do not affect the y-coordinate, we have that

$$v = y.$$

- When a point on a fiber appears in focus, it's u-position can be determined very precisely because its footprint on the image is small due to the fact that the camera is of high resolution.
- However, its d-position is more fuzzy. The paper claims there's an error about 1mm in the d-direction.
- As such the point that appears in focus is interpreted as begin a short strip in the d-direction.

  A hair strand is therefore a ribbon strip that is always edge-on to the camera's view direction.
- "Finding hair in the volume amounts to finding sets of intersecting ribbons that are mutally consistent."

3 Image Filtering

- For each of the photograph capture, we need to apply a ridge detector to detect hair fibers.
- The paper uses a modified version of the Canny operator for edge detection.
- Before applying the filtering, the paper chooses a scale which will corresponds to the width of the filter to be applied to the image. This filter should corresponds to the width of the hair fiber.
- The filter applied is the *two-dimensional Gabor filter.*

  $$G(u,v) = \exp\left(-\frac{1}{2}\left[\frac{\tilde{u}^2}{\sigma_u^2} + \frac{\tilde{v}^2}{\sigma_v^2}\right]\right)\cos\left(\frac{2\pi\tilde{u}}{\lambda}\right)$$

  where

  $$\begin{aligned}\tilde{u} &= u\cos\varphi + v\sin\varphi, \text{ and}\\ \tilde{v} &= -u\cos\varphi + v\cos\varphi.\end{aligned}$$

  The parameter φ and λ determines the orientation and the period of the sinusoid plane wave. The σ_u and σ_v parameter control the standard deviation of the Gaussian envelope.
- The "scale" that we determine in the last item is the values of λ, σ_u and σ_v.

  The paper uses $\sigma_u = \sigma_v = 1$ and $\lambda = 3$.
- The uses 32 values of φ to get 32 directions. This results in 32 Gabor filters to convolve with each photograph.
- To detect a ridge, we go to the pixels of the 32 convolved images. We do non-maximum suppression on it.

  That is, for a pixel with intensity M, let M_1 and M_2 be bilinearly interpolated values resulting from a lookup normal to the local orientation of the Gabor filter. We create a new image with intensity given by:

  $$M' = \frac{M - M_{\max}}{M_{\max} + c}$$

  where $M_{\max} = \max\{M_1, M_2\}$ and c is a constant.
- Hysteresis thresholding is applied to the output of the non-maximum suppresion step.

- **Hysteresis thresholding** uses two thresholds: T_{high} and T_{low}. With this, each pixel is classified into three classes:
  - A pixel is *strong* if its intensity is greater than T_{high}.
  - A pixel is *weak* if its intensity is less than or equal to T_{low}.
  - A pixel is a *candidate* otherwise.

  Then:
  - The algorithm discards a pixel if it is weak.
  - The algorithm retains a pixel if it is strong.
  - For a candidate pixel, find the transitive closure of candidate and strong pixels to that pixel. It the connected component has a strong pixel, then retain the pixel. If not, discard the pixel.
- The output of the hysteresis thresholding is a binary image. If a pixel is turned on, it indicates that there's a ridge in that pixel.
- If a pixel is turned on after the original image is convolved with a Gabor filter of a certain orientation, then the orientation of the ridege is given by the orientation of the Gabor filter.
- Hence, the output of the filtering stage are pixel positions and the associated orientation of the ridge at that pixel.

4 Growing Hair

- The gist of the hair growing algorithm is to start with an initial position, estimate the local direction at that position, take a step in that direction, and then do the same for the new position until we cannot do so. This should trace a path through one of the hair strand.
- However, this process outline above is not very robust spurious ridge detection. We need a more robust process.
- The paper proposes correcting the position after following a step in the local direction so that the new position is still located in the ribbon corresponding to the ridges.

  This correction step can be described as a least-square problem.

4.1 Direction Estimation

- The pixel position and the depth associated with the image it comes from gives a (u, v, d) coordinate of a 3D point. We take all the (u, v, d) and its corresponding orientation and transform them into the (x, y, z) coordinate system by the simple rotation outlined earlier.
- The algorithm relies on the ability to find near 3D point to a given query point. So, the (x, y, z)-positions of the ridge points should be put into a kd-tree to enable this capability.
- Given a point $\mathbf{x} = (x, y, z)^T$, let there be m near ridge points.
  Let these points be viewed from camera angles $\theta_{i_1}, \theta_{i_2}, \dots, \theta_{i_m}$.
  Let the image space coordinates of the points be $\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_m$.
  Let the translation of the camera for these points be $d_1, d_2, \dots, d_m$.
  Let the 2D orientation of these points be $\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_m$.
- We shall use the information in the last item to deduce the fiber orientation near $\mathbf{x}$.

- Now, we assume the perfect camera model.
  Let l be the distance from the camera's "eye" to the focal plane.
  Let f be the distance from the "eye" to the image plane.
  Let $\mathbf{s}_k = (s_k, t_k, 1)^T$ in homogeneous coordiate.
  Let $\mathbf{x} = (x, y, z, 1)$ in homogeneous coordinate as well.

  Expressing $\mathbf{x}$ in the (u, v, d)-coordinate system, we have that

$$\begin{bmatrix} u_k \\ v_k \\ d_k \end{bmatrix} = \begin{bmatrix} \cos\theta_{i_k} & 0 & \sin\theta_{i_k} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta_{i_k} & 0 & \cos\theta_{i_k} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

  Simple proportionality tells us that

$$\frac{u_k}{l} = \frac{s_k}{f}, \text{ and } \frac{v_k}{l} = \frac{t_k}{f}.$$

  In other words, in homogeneous coordinates,

$$\begin{bmatrix} s_k \\ t_k \\ 1 \end{bmatrix} = \begin{bmatrix} u_k \\ v_k \\ \frac{l}{f} \end{bmatrix} = \begin{bmatrix} \cos\theta_{i_k} & 0 & \sin\theta_{i_k} & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{\sin\theta_{i_k}}{f} & 0 & \frac{\cos\theta_{i_k}}{f} & \frac{l-d_k}{f} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

  As such

$$\mathbf{u}_k = \begin{bmatrix} \cos\theta_{i_k} & 0 & \sin\theta_{i_k} & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{\sin\theta_{i_k}}{f} & 0 & \frac{\cos\theta_{i_k}}{f} & \frac{l-d_k}{f} \end{bmatrix} \mathbf{x}.$$

  For convenience, let the mapping from $\mathbf{x}$ to $\mathbf{u}_k$ be denoted by P_k

References