Robust Skin-Feature Tracking in Free-Hand Video from Smartphone or Robot-Held Camera, to Enable Clinical-Tool Localization and Guidance

Chun-Yin Huang[1] and John Galeotti[2]

***Abstract*— Our novel skin-feature visual-tracking algorithm enables anatomic vSLAM and (by extension) localization of clinical tools relative to the patient's body. Tracking naturally occurring features is challenging due to patient uniqueness, deformability, and lack of an accurate a-priori 3D geometric model. Our method (i) tracks skin features in a smartphone-camera video sequence, (ii) performs anatomic Simultaneous Localization And Mapping (SLAM) of camera motion relative to the patient's 3D skin surface, and (iii) utilizes existing visual methods to track clinical tool(s) relative to the patient's reconstructed 3D skin surface. (We demonstrate tracking of a simulated ultrasound probe relative to the patient by using an Apriltag visual fiducial). Our skin-feature tracking method utilizes the Fourier-Mellin Transform for robust performance, which we incorporated and extend an existing Phase Only Correlation (POC) based algorithm to be suitable for our application of free-hand smartphone video, wherein the distance of the camera fluctuates relative to the patient. Our SLAM approach further utilizes Structure from Motion and Bundle Adjustment to achieve an accurate 3D model of the human body with minimal drift-error in camera trajectory. We believe this to be the first freehand smartphone-camera tracking of natural skin features for anatomic tracking of surgical tools, ultrasound probe, etc.**

I. Introduction

Medical image-guided interventions have benefited from the rapid evolution of computer vision algorithms and medical imaging methodologies. There is a never-ending need to perform surgery more accurately using less harmful or lower cost clinical tools. For example, Ultrasound (US) combines several advantages including low-cost, real-time operation, a small size that is easy to use and transport, and a lack of ionizing radiation. However, US suffers from a lack of contextual correlates due to changing and un-recorded probe location, which makes it challenging to be applied in certain clinical uses. Therefore, we and others before us have sought to make a 3D tracking and visualization systems that connect the coordinates of clinical tools with the human body, via camera-based computer vision. Such systems could be used for image-guided therapy, 3D US image reconstruction, recording and replaying clinical imaging/interventions, etc.

A key challenge for 3D tracking on the human body stems from the lack of a stable anatomic coordinate system, which typically requires a pre-built 3D model. There are several ways to achieve this. Sun et al. [11] used Scale-Invariant Feature Transform (SIFT) feature tracking [6] on images that were taken by a low-cost camera mounted on a US probe as input for simultaneous localization and mapping (SLAM) for 3D reconstruction. While the method is cost effective, SIFT often fails to track natural skin features in our application, resulting in high cumulative error. Others manually attach known markers on the body such as the work by Lange et al. [5]. However, tracking tissue deformation requires a dense set of tracked points, and natural features are more convenient than attaching or inking large numbers of artificial markers, and may be preferred by patients, especially for publicly visible skin, e.g. on forearms or faces. Many artificial markers protrude or cover the skin in a manner that can get in the way of the clinician, and in any case artificial markers do not usually persist across months or years, as would be desirable for longitudinal patient monitoring. Some prior work utilized traditional feature tracking methods to either create an initial guess of sparse depth estimation or perform initial 3D reconstruction on monocular endoscopy images, and these approaches then further computed the dense results with either zero-mean cross correlation [15] or deep learning methods [14]. Nevertheless, as mentioned above, traditional feature tracking often fails to accurately locate natural features on the human body, so that in practice their initial guess may not be stable enough for consistent further dense estimation. On the other hand, Wang et al. [13] used a commercial clinical 3D scanning system to acquire a preoperative 3D patient model, which aided in determining the location and orientation of the probe and the patient. However, we want to achieve the task using only a smartphone camera, without the time or cost of obtaining an accurate preoperative 3D scan. More recently, Ito et al. [4] utilized phase only correlation (POC) tracking as previously proposed by Takita et al. [12] to robustly find subtle features with sub-pixel precision on the human body. However, their POC method was unable to track skin features when the camera was held free-hand because their phase correlation was not invariant to scale and rotation. They mounted their camera directly on the probe and avoided tilting the probe to maintain a fixed camera distance.

We desire camera tracking of both skin features and tools, for which the camera must be further from the patient to have a wider field of view. With patient motion and free-hand operator motion, the distance from camera to patient will naturally vary during the course of a procedure. *Our chief contribution is a flexible approach that allows a handheld smartphone camera to robustly identify and track sub-pixel skin features, builds a 3D reconstruction model, and monitors*

*This work was not supported by any external organization

[1]Chun-Yin Huang is with the Biomedical Engineering Department, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213, USA chunyinh@alumni.cmu.edu

[2]John Galeotti is with the Robotics Institute, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213, USA jgaleotti@cmu.edu

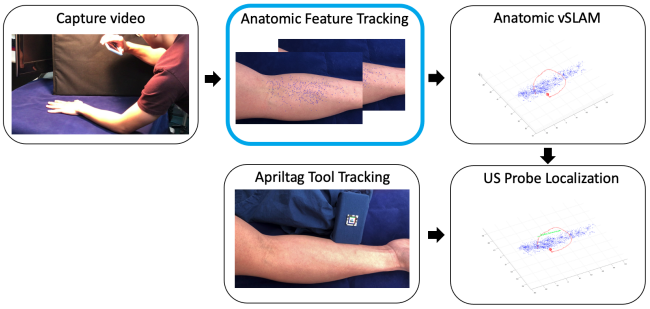


Fig. 1: We incorporate our novel anatomic feature tracking algorithm into an end-to-end 3D tracking and reconstruction system. The blue box indicates our chief contribution to the system. The reconstructed 3D arm features, tracked tool (i.e. AprilTag), and computed smartphone camera trajectory are depicted as blue, red, and green dots, respectively.

the relative movement between visually tracked tools (such as an US probe) and a patient. We demonstrate that our unique combination of existing approaches, including our scale- and rotation-invariant Fourier-based feature tracking, enables smartphone-single-camera anatomic vSLAM.

The long-term goal for this work is to enable a readily accessible smartphone camera to achieve free-hand image-guided clinical intervention while relegating the use of fiducial tags to tools rather than patients. The overall process is shown in Fig. 1, and the remaining paper is organized as follows. In Section II, we present our Fourier-based human skin feature tracking, our unique setup for anatomic SLAM, and the combination of clinical tool tracking into the integrated system. Section III contains our evaluation experiments and discussion. Section IV is our conclusion and future work. We used the human forearm and US probe as our anatomical target in this paper. Due to COVID-19 circumstances, we used a dummy US probe consisting of a similarly sized box, which we refer to as the "US probe" for simplicity in the following contents.

II. Approach

A. Fourier-Based Feature Tracking

Our novel contribution is our robust scale- and rotation-invariant Fourier-based feature tracking, which enables the augmentation of an existing system into our flexible low cost 3D reconstruction and localization system.

1) Prior POC method: Our methodology is closely inspired by [4], which we briefly describe in the next few equations to lay a common notation framework. Their POC method starts by detecting feature points on the first frame using *Good Features to Track* (GFtT) [10], and subsequently tracks the features along the video sequence using Phase-Only Correlation (POC), which relies on the Fourier shift theorem as we will show. Let f_1 and f_2 be two small image patches centered on the corresponding feature points in two consecutive frames that differ only by a shift (x_0, y_0):

$$f_2(x, y) = f_1(x - x_0, y - y_0). \tag{1}$$

From the Fourier shift theorem, the corresponding Fourier Transformed images F_1 and F_2 will be:

$$F_2(\xi, \eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} * F_1(\xi, \eta). \tag{2}$$

The cross-power spectrum of two image patches with Fourier Transform is then defined as

$$\frac{F_1(\xi, \eta) F_2^*(\xi, \eta)}{|F_1(\xi, \eta) F_2^*(\xi, \eta)|} = e^{j2\pi(\xi x_0 + \eta y_0)} \tag{3}$$

and the inverse Fourier Transform of the right hand side of (3) results in an impulse function:

$$F^{-1}\{e^{j2\pi(\xi x_0 + \eta y_0)}\} = \delta(x - x_0, y - y_0) \tag{4}$$

Eq. (4) can be used to estimate sub-pixel translation displacements by fitting an analytical peak model of the POC function as described in [12]. If the similarity score is too low, the feature point will be eliminated as an outlier. When visible regions become too sparse with too few tracked feature points, new features are again identified in the sparse regions using GFtT. By repeating the process, each sequence in the video contains an acceptable number of well distributed tracked feature points.

2) Our Fourier-Mellin enhancements over POC: Although the POC feature-tracking method can accurately track features from skin, the algorithm may rapidly lose track of feature points through a video sequence when the distance between the camera and the object is not fixed or the camera is rotated during recording. Since we want to build a free-hand system, we have refined the algorithm by utilizing the image registration method proposed by *Srinivasa et al.* [16], which utilized the Fourier Mellin Transform to first rectify the scale and rotation between two image patches and then calculated the shift. Their math is described next.

Assuming there are scale s, rotation θ, and shift (x_0, y_0) variants between f_1 and f_2, we can augment (1) into

$$\begin{aligned} f_2(x, y) = & f_1(s * x \cos\theta + s * y \sin\theta - x_0, \\ & - s * x \sin\theta + s * y \cos\theta - y_0). \end{aligned} \tag{5}$$

and then the Fourier Transformed F_1 and F_2 will become

$$\begin{aligned} F_2(\xi, \eta) = & \frac{1}{|s|} e^{-j2\pi(\xi x_0 + \eta y_0)} * \\ & F_1(\frac{\xi}{s}\cos\theta + \frac{\eta}{s}\sin\theta, -\frac{\xi}{s}\sin\theta + \frac{\eta}{s}\cos\theta). \end{aligned} \tag{6}$$

Take the magnitude on both sides and remove the constant $\frac{1}{|s|}$ because it will be cancelled in the division of Eq. (3)

$$G_2(\xi, \eta) = G_1(\frac{\xi}{s}\cos\theta + \frac{\eta}{s}\sin\theta, -\frac{\xi}{s}\sin\theta + \frac{\eta}{s}\cos\theta). \tag{7}$$

where $G_i = |F_i|$. We can denote (7) in RHS Polar coordinates as

$$\begin{aligned} G_2(\xi, \eta) = G_1(& \frac{r}{s}\cos\theta\cos\psi + \frac{r}{s}\sin\theta\sin\psi, \\ & - \frac{r}{s}\sin\theta\cos\psi + \frac{r}{s}\cos\theta\sin\psi). \end{aligned} \tag{8}$$

where $\xi = r\cos\psi, \eta = r\sin\psi$, and then apply the product-to-sum trigonometric identities on (8) to get

$$G_2(\xi, \eta) = G_1(\frac{r}{s}\cos(\psi - \theta), \frac{r}{s}\sin(\psi - \theta)). \tag{9}$$

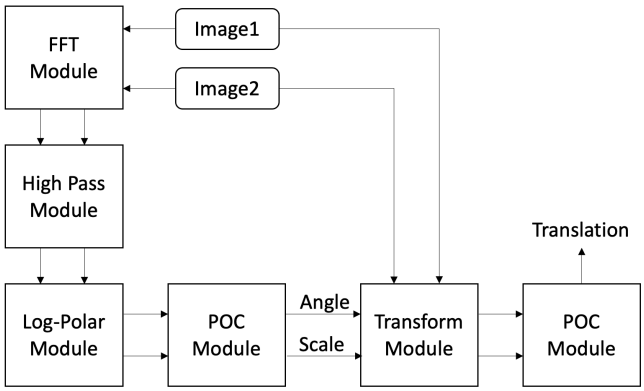


Fig. 2: Overview of the registration algorithm proposed by [16]. The POC modules contain our novel algorithm described in Section II-A.

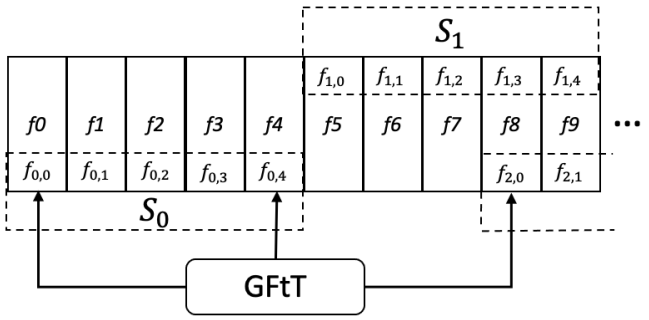


Fig. 3: Anatomic vSLAM setup. Notice that we assign every 5 frames into a set, with one overlapped frame between sets.

The transformations of (9) from Cartesian coordinates to Polar coordinates (10) and Log-Polar coordinates (11) are:

$$G_2(r, \psi) = G_1(\frac{r}{s}, \psi - \theta). \tag{10}$$

$$G_2(\log r, \psi) = G_1(\log r - \log s, \psi - \theta). \tag{11}$$

We can apply the POC approach to find the "shift" of (11), which indicates the scale and rotation of the image patches.

As shown in Fig. 2, before we apply the POC-Fourier-Mellin methods, the image patches are passed through a high pass filter to enhance the skin features. This helps match skin creases, freckles, etc. rather than matching shading.

The Fourier-based feature tracking is more robust on natural skin features than POC feature tracking and traditional feature tracking methods. The discussion is in Section III-A.

B. Experimental Setup: Anatomic vSLAM and US probe localization

Our current vSLAM system processes images according to small overlapping batches and the nature of Fourier-based feature tracking as shown in Fig. 3. Since the features are tracked frame by frame, we define a set S_i which contains 5 consecutive captured frames $f_{i,j}$, i.e. $S_i = \{f_{i,0}, f_{i,1}, f_{i,2}, f_{i,3}, f_{i,4}\}$. In $f_{i,0}$, GFtT is used to find initial features with a constraint that the features are at least 5 pixels away from each other (POC requires separation between features to operate reliably). From $f_{i,1}$ to $f_{i,4}$, we use Fourier-based feature tracking to track the corresponding features along the frames. After tracking features from this set S_i, the tracked good features of $f_{i,4}$ will be inherited by $f_{i+1,0}$ in the new set S_{i+1}. After we set $f_{i+1,0}$ = $f_{i,4}$, we then repeat the process for S_{i+1} by finding new features from the areas of $f_{i+1,0}$ that lack features, while the inherited features provide necessary overlap to maintain correspondence between the 2 sets. After that we perform feature tracking again in S_{i+1}. The process will go on until we reach the end of the video sequence we are processing.

1) Anatomic SLAM: An Intrinsic Matrix is obtained by calibration of the camera before video acquisition. Following standard procedure, we employ the camera calibration toolbox provided by Matlab [8] for this part after which we keep the Intrinsic parameters fixed in our system during vSLAM.

Next, we convert color images to grayscale and then enhance the appearance of skin features using contrast limited adaptive histogram equalization (CLAHE) [20] to find better spatial frequency components for the feature detection and our feature tracking.

While operating on a video sequence, we first do Structure from Motion (SfM) [3] locally on every newly obtained set S_i, and then use the locally computed 3D positions as an initialization for the global set $\mathbf{S} = \{S_0, S_1, S_2...\}$. Once a new set is obtained , we use Bundle Adjustment [3] to refine the overall 3D scheme. By doing this, we can simultaneously update and refine the 3D feature points and camera motions while reading in new frames from the video.

Since we want to compute SfM for every 5 frames, we decide not to use the traditional pipeline which includes the normalized five-point algorithm and random sample consensus (RANSAC) [2]. Instead, we perform SfM in a manner similar to [1]. First, we use *reprojection error* which is defined by the Euclidean distance $||x - x_{rep}||^2$, where x is a tracked feature point and x_{rep} is a point obtained by projecting a 3D point back to the image using the calculated projection matrix. After we obtained the initialized 3D points, camera projection matrices, and corresponding 2D features in a set, we minimize the *reprojection error* (using *Ceres* to solve the numerical problem). In this latter stage, we fix the Intrinsic Matrix and let the system update the 3D points and camera Extrinsic Matrix repeatedly.

For higher robustness, we set an additional constraint that new feature points must persist across at least 2 consecutive sets before they are added to our point cloud of the patient. Higher reconstruction quality can be achieved by setting larger constraint thresholds.

As is typical of SfM, the resulting 3D point cloud of the arm and the camera trajectory are only recovered up to a scale factor. Currently, we manually adjust the 3D positions to fit into real world coordinates, and in the future we envision briefly placing a calibrated object (such as a ruler or a small, flat *AprilTag*) on the patient's skin during the first few frames of the video.

2) US Probe Localization: Unlike patients, it is relatively easy to place fiducial markers on clinical tools, which tend to be (piece-wise) rigid. We can accurately track the 3D position and orientation of our US probe by attaching an *Apriltag* marker and using the associated tracking software by [9].

After we reconstruct a patient's skin surface (e.g. forearm in this paper) during the first several seconds of video, our system is ready for tools (e.g. Ultrasound) to be introduced. Our system continues to run SfM and Bundle Adjustment algorithms while the US probe is scanning the patient, as necessary to accommodate (1) the hand-held movement of the smartphone camera and (2) possible skin deformation or patient motion. Continuous tracking of both skin and tools relative to the moving camera allows consistent tracking of tools relative to the skin.

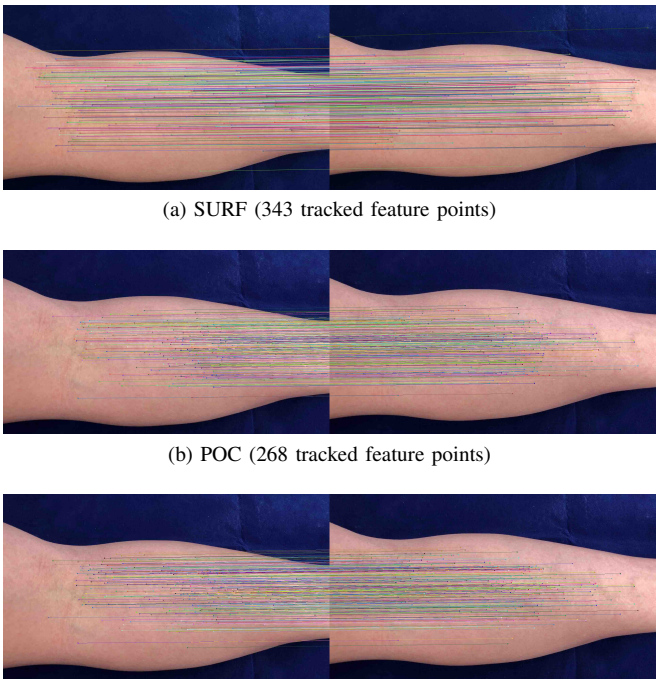
Note that our feature tracking method may also find features on US probes, which might confuse 3D reconstruction of the skin surface. We handle this problem by first detecting the US Apriltag, and then mask-out the US probe from our video images *before* we run our feature detection algorithms.

III. Experiments and Discussion

In the experiments in this section, we use a freehand iPhone 8 camera to do scanning on a real arm with 1080p resolution. Freehand smartphone video tracking requires a wide field of view (FoV) to ensure the operating area remains in view, for instance, the validated working range in the paper is between 27 and 54 $pixels/cm^2$, and we would like to extend to other cameras at different distances in future experiments. However, the wide FoV also introduces many spurious objects that we do not want to track. In the future, we would need to automatically identify which pixels correspond to the patient (perhaps using human pose tracking followed by semantic segmentation), but for now we simply used a blue screen background to isolate the skin. Likewise, we covered our dummy US probe with a blue cloth so that only the AprilTag would require masking, but in the future we would make use of an accurate 3D CAD model of the real US probe to create an accurate 2D protective-view pixel mask (with a few-pixels of safety margin). After masking the color image, we then proceed with our vSLAM pipeline.

We minimize motion blur by forcing a short shutter speed, i.e. 120 fps, of which we only preserve every 20th frame to end up at our target frame rate of 6 fps (SfM requires some degree of motion within each of our sets S_i). Due to the high fps, we carefully avoid flickering electric light sources. Multiple soft LED lights (DC powered or else $>$ 12kHz PWM) from different angles is recommended. However, due to COVID-19 circumstances, we instead used natural sun light on a cloudy day as our light source. As aforementioned, we update the 3D reconstruction every 4 captured frames (the 5th and 1st frames of consecutive sets overlap), so we can update our 3D skin tracking every 2/3 second (4 frames/6 fps) (fiducial-based tool tracking could independently run much faster).

There are 4 experiments in this section which includes III-A Feature Matching Comparison, III-B 3D Reconstruction Evaluation, III-C Systematic Cumulative Error Evaluation, and III-D Overall vSLAM and US Probe Tracking.



(a) SURF (343 tracked feature points)

(b) POC (268 tracked feature points)

(c) OURS (308 tracked feature points)

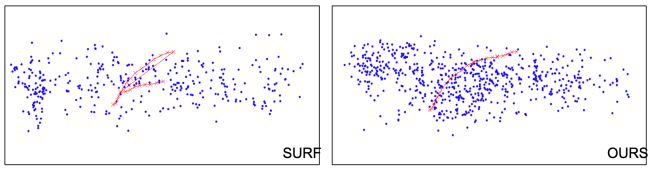
Fig. 4: The feature tracking results from different methods.

A. Feature Matching Comparison

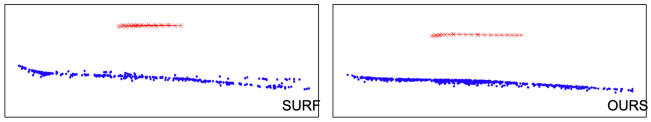
In the first experiment, we evaluated the robustness of our Fourier-based feature matching algorithm against other feature-trackers on natural human skin features. In this subsection, we will first show the real human skin feature tracking results from our Fourier-based method, POC method, and other traditional methods(ORB [21], SIFT [6], SURF [7]), and then do 3D reconstruction on the tracked features if the method shows compelling accuracy. We use the same preprocessing module for every feature tracking method. The initial number of GFtT features for POC and Fourier-based tracking is set to be 500. We empirically evaluate the size of image patches from 51-91 pixels, and find patches with size 71 pixels resulted in the most tracked features, which is similar to the POC choice of [4].

Fig. 4 shows the number of tracked features on the first and the last frame from a set. Since ORB (found 75 features) and SIFT (found 129 features) lose track of features rapidly and there are obvious mismatches, we show only the features found by SURF, POC, and our Fourier-based tracking. Only SURF can find similar number of features as we do. Thus, we decide to do 3D reconstruction on a few sets using SURF and our method in Fig. 5.

Although SURF may find more skin features through a set, it fails to keep track of features when the number of frames is increasing, which results in unstable skin surface and camera trajectory reconstruction. Fig. 5(a) shows that not only the camera trajectory is incorrect (which we conjecture that it's due to feature mismatching) but the 3D points are



(a) Viewing the 3D reconstructions from top



(b) Viewing the 3D reconstructions from side

Fig. 5: The 3D reconstructions using SURF and Fourier-based tracking.

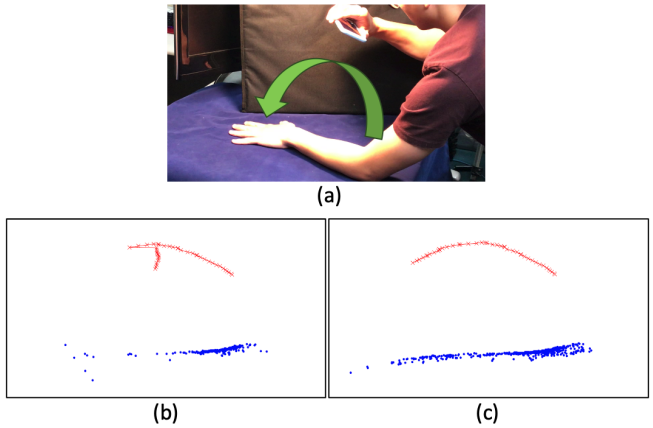


Fig. 6: 3D reconstruction results using POC (left) vs our method (right), viewing the side of the arm-surface point cloud. The skin surface (blue) and camera trajectory (red) are also depicted. Our Fourier-based feature tracking solves for the physical camera trajectory and tracks more feature points. Supplementary video demonstrates another scan, showing live 3D reconstruction and camera trajectory updates.

sparser. In Fig. 5(b) we can see that the reconstructed skin surface from SURF is jittering.

To compare the performance differences (especially scale changes) between POC and Fourier-based feature tracking for freehand 3D anatomic reconstruction, we captured a smartphone video sequence in which the distance between the camera and the arm varies during recording as shown in Fig. 6(a). The green arrow indicates the camera motion. Observe that our method presents higher accuracy by showing a more representative smooth arc of the camera motion in Fig. 6(c). The shape of the blue reconstructed skin feature points lie on the arm where the right hand side is the elbow and the left hand side is the wrist. The red camera trajectory also recovers our intended smooth camera motion. The reason for the poor reconstruction based on the original POC tracking in this case is that it failed to track enough feature points between frames, which leads to high reconstruction error, while our proposed refinement solves this problem by finding more consecutive feature points.

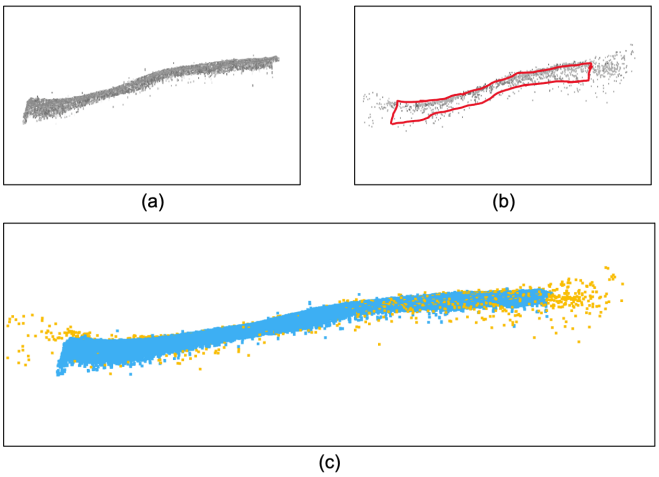


Fig. 7: Comparison of 3D models computed by *Blaser* and our system. Our system scans a larger area, and the red lines in (b) indicate the approximate region of overlap with (a).

B. 3D Reconstruction Evaluation

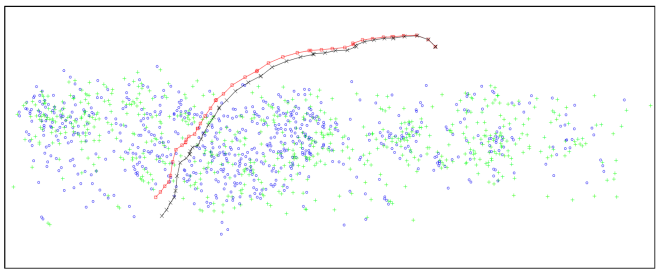
To evaluate the robustness of our system, we applied *Blaser* 3D scanner [18] to obtain the 3D model of the patient's arm and used Iterative Closest Points (ICP) [17] algorithm to evaluate our point cloud.

Fig. 7 shows the 3D models where Fig. 7(a) is the 3D model generated by *Blaser* and Fig. 7(b) is the 3D reconstruction found by our system. Fig. 7(c) is the registration result where *Blaser* 3D model is marked as blue and our 3D reconstruction result is marked as yellow. One can see that the overlapping regions show descent alignments, and the RMSE of the inliers (overlapping regions) is $0.80mm$.

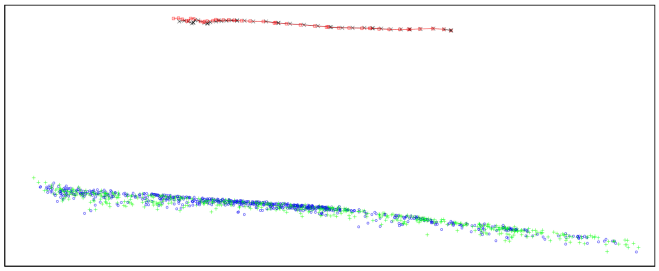
C. Systematic Cumulative Error Evaluation

For this experiment, we sought to test the stability of our tracking system. We design a system similar to [19] by capturing a video sequence containing N frames, and then we play the sequence in order from beginning to end and then from end to beginning, producing a single long sequence of $2N$ frames to input into our algorithm. This way, for each frame in the first half of the video f_i, we know of a perfectly corresponding frame in the second half of the video $f_{2N-(i+1)}$. We treat this composite video as if it were a single freehand video, having our algorithm perform 3D reconstruction across its entirety. By checking whether the reconstructed 3D points are on the same skin surface and the forward and backward camera trajectories are on the same path, we can estimate the accumulated divergence error.

The total accumulated error includes both feature tracking error and sequential reconstruction error, since when playing the video reversely, different points will end up being selected for tracking and will be tracked in the reversed order. As the algorithm analyzes the second half of the video, it becomes progressively more difficult from frame f_N to frame f_{2N-1} to achieve perfect correspondence between the outputs for the corresponding frames f_i and f_{2N-i}.



(a) Viewing from top of the 3D reconstruction



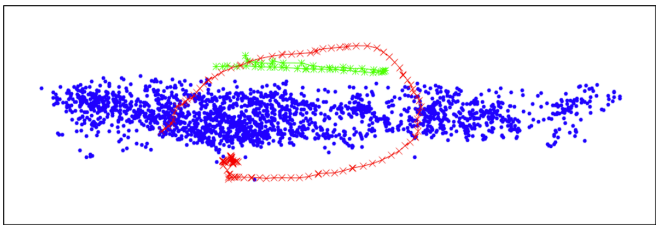
(b) Viewing from side of the 3D reconstruction

Fig. 8: Accumulated divergence error. The skin surface (blue and green) and camera trajectory (red and black) are depicted by their corresponding colors during forward and reverse playback, respectively. The point clouds visually appear to be on the same surface, and the computed camera trajectory ends close to where it began, indicating that our system appears to be performing well with modest error accumulation.

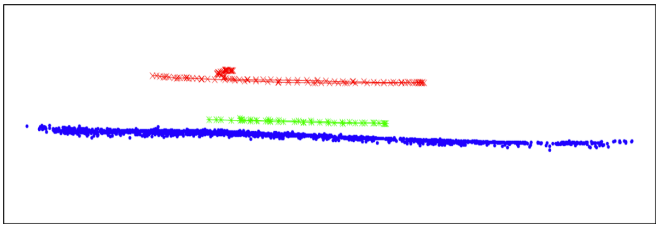
The result is shown in Fig. 8. One can see from Fig. 8(a) the computed camera trajectories from the froward pass (marked as red) and the reverse pass (marked as black) suffer low accumulated divergence error. Also from Fig. 8(b) the reconstructed arm surface from forward pass (marked as blue) and reverse pass (marked as green) lie in the same surface and have the same shape. The reason the blue and green 3D points are distributed differently in Fig. 8(a) is that the initial feature points found in the first frame f_1 of the sets S is now different due to finding different Good Features to Track during the reverse pass.

D. Overall vSLAM and US Probe Tracking

For our last experiment, we moved the smartphone camera in a circular motion above the target arm skin, to obtain a sufficient diversity of viewpoints to perform better 3D reconstruction. We then moved the camera further away from the patient so that the whole arm becomes visible to the camera. The dummy US probe is then introduced and moves as if scanning the body, with motion toward the right and then back to the left. Our anatomic vSLAM tracking results are shown in Fig. 9. One can see how the arm surface (marked in blue), the camera trajectory (marked in red), and the US probe scanning path (marked in green) are reconstructed and tracked by our system. The small, dense cluster of camera-trajectory points occurs just after moving the camera away from the arm. The spread of this point



(a) Viewing from top of the 3D reconstruction



(b) Viewing from side of the 3D reconstruction

Fig. 9: Reconstructed skin surface(blue), camera trajectory(red), and US probe scanning path(green).

cluster is indicative of the freehand jittering of the camera's position when attempting to hold the camera still.

Our supplementary video contains the process of our vSLAM system (before US probe tracking), in which the smartphone also follows a circular trajectory. The video shows frame-by-frame how the skin surface is reconstructed in 3D, using sets of five images as previously described. The camera trajectory is also shown, with both the primary trajectory computed from the image sets, as well as with intermediary (every-frame) estimates of current camera pose. These intermediary pose estimates are not used to update the point cloud, but are useful for continuously tracking tools relative to the patient via the camera position.

IV. Conclusion and Future Work

We presented what we believe to be the first anatomic vSLAM system for use by a freehand smartphone imaging natural human skin. We achieve3d sub-mm RMS error in skin-surface reconstruction, and our free-hand smartphone motion captures a more complete surface that wraps around the patient than would be possible from a single perspective. We demonstrated the robustness of our system used in combination with *Apriltag* tool tracking. In the future, we will use GPU acceleration to make our system fully real time and validate its tracking of a real US probe on real tissue that is partially covered with US gel. We will also place a physical calibration object on the patient to correctly calibrate scale for the patient's size. Lastly, we plan to incorporate recent smartphone 3D scanners for improved shape scanning while using our Fourier-based method to track rotating/translating skin features, e.g. to track arm axial arm rotation for which 3D geometry would not appreciably change.

V. Acknowledgement

We wish to thank Daqian Cheng and Haowen Shi for their assistance using Blaser to acquire a 3D scan of our arm.

REFERENCES

[1] S. Choi, An Invitation to 3D Vision: A Tutorial for Everyone, (2017), *GitHub repository*, https://github.com/sunglok/3dv_tutorial.

[2] M.A. Fischler and R.C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comm. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[3] R. Hartley and A. Zisserman, *Multiple View Geometry*, Cambridge University Press, 2004.

[4] S. Ito, K. Ito, T. Aoki, J. Ohmiya, and S. Kondo, "Probe localization using structure from motion for 3D ultrasound image reconstruction," Proc. - *Int. Symp. Biomed. Imaging*, pp. 68–71, 2017, doi: 10.1109/ISBI.2017.7950470.

[5] T. Lange, S. Kraft, S. Eulenstein, H. Lamecker, and P.M. Schlag, "Automatic calibration of 3D ultrasound probes," *Proc. Bildverarbeitung fur die Medizin 2011*, pp. 169–173, Mar. 2011.

[6] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, pp. 91–110, 2004, [Online]. Available: https://www.cs.ubc.ca/ lowe/papers/ijcv04.pdf.

[7] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Comput. Vis. Image Underst.*, vol. 110, no. 3, pp. 346–359, 2008, doi: 10.1016/j.cviu.2007.09.014.

[8] Matlab, Camera Calibration Toolbox for Matlab: http://www.vision.caltech.edu/bouguetj/calib_doc/.

[9] E. Olson, "AprilTag: A robust and flexible visual fiducial system," *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 3400–3407, 2011, doi: 10.1109/ICRA.2011.5979561.

[10] J. Shi and C. Tomasi, "Good features to track," *Proc. Int'l Conf. Computer Vision and Pattern Recognition*, pp. 593–600, 1994.

[11] S.Y. Sun, M. Gilbertson, and B.W. Anthony, "Probe localization for freehand 3D ultrasound by tracking skin features," *Med. Image Comput. Comput. Assist. Interv.*, vol. 17, pp. 365–372, 2014.

[12] K. Takita, M.A. Muquit, T. Aoki, and T. Higuchi, "A sub-pixel correspondence search technique for computer vision applications," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E87-A, no. 8, pp. 1913–1923, 2004.

[13] J. Wang, V. Shivaprabhu, J. Galeotti, S. Horvath, V. Gorantla, and G. Stetten, "Towards video guidance for ultrasound, using a prior high-resolution 3D surface map of the external anatomy," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8678, pp. 51–59, 2014.

[14] X. Liu et al., "Dense Depth Estimation in Monocular Endoscopy with Self-Supervised Learning Methods," *IEEE Trans. Med. Imaging*, vol. 39, no. 5, pp. 1438–1447, 2020, doi: 10.1109/TMI.2019.2950936.

[15] N. Mahmoud, T. Collins, A. Hostettler, L. Soler, C. Doignon, and J. M. M. Montiel, "Live tracking and dense reconstruction for handheld monocular endoscopy," *IEEE Trans. Med. Imaging*, vol. 38, no. 1, pp. 79–89, 2019, doi: 10.1109/TMI.2018.2856109.

[16] B. Srinivasa Reddy and B. N. Chatterji, "An FFT-based technique for translation, rotation, and scale-invariant image registration," *IEEE Trans. Image Process.*, vol. 5, no. 8, pp. 1266–1271, 1996, doi: 10.1109/83.506761

[17] Z. Zhang, "Iterative point matching for registration of freeform curves and surfaces," *Int'l J. Computer Vision*, vol. 13, no. 2, pp. 119–152, Oct. 1994.

[18] D. Cheng, H. Shi, M. Schwerin, L. Li and H. Choset, "A Compact and Infrastructure-free Confined Space Sensor for 3D Scanning and SLAM", *IEEE Sensors*, 2020.

[19] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," *Proc. - Int. Conf. Pattern Recognit.*, pp. 2756–2759, 2010, doi: 10.1109/ICPR.2010.675.

[20] S. M. Pizer, E. P. Amburn, J. D. Austin, et al.: Adaptive Histogram Equalization and Its Variations. *Computer Vision, Graphics, and Image Processing* 39 (1987) 355-368.

[21] E. Rublee, V. Rabaud, K. Konolige, G. R. Bradski: ORB: An efficient alternative to SIFT or SURF. *International Conference on Computer Vision* 2011: 2564-2571.