DIGITAL ACCESS TO
SCHOLARSHIP AT HARVARD
DASH.HARVARD.EDU

HARVARD LIBRARY
Office for Scholarly Communication

High-speed surface reconstruction of a flying bird using structured light

The Harvard community has made this article openly available. **Please share** how this access benefits you. Your story matters

Citation	Deetjen, Marc E., Andrew A. Biewener, and David Lentink. 2017. "High-Speed Surface Reconstruction of a Flying Bird Using Structured Light." The Journal of Experimental Biology 220 (11) (March 27): 1956–1961. doi:10.1242/jeb.149708.
Published Version	10.1242/jeb.149708
Citable link	http://nrs.harvard.edu/urn-3:HUL.InstRepos:33086350
Terms of Use	This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Open Access Policy Articles, as set forth at http://nrs.harvard.edu/urn-3:HUL.InstRepos:dash.current.terms-of-use#OAP

High-Speed Surface Reconstruction of a Flying Bird Using Structured-Light

Marc E. Deetjen[1*], Andrew A. Biewener[2], David Lentink[1]

[1]*Department of Mechanical Engineering, Stanford University, Palo Alto, CA, USA.*

[2]*Harvard University, Department of Organismic and Evolutionary Biology, Cambridge, MA, USA.*

[*]Corresponding author: Marc E. Deetjen mdeetjen@stanford.edu

Keywords: animal locomotion, high-speed, single-shot, structured-light, surface reconstruction, wing morphing

Abstract

Birds fly effectively and maneuver nimbly by dynamically changing the shape of their wings during each wingbeat. These shape changes have yet to be quantified automatically at high temporal and spatial resolution. Therefore, we developed a custom 3D surface reconstruction method, which uses a high-speed camera to identify spatially encoded binary striped patterns that are projected on a flying bird. This non-invasive structured-light method allows automated 3D reconstruction of each stand-alone frame and can be extended to multiple views. We demonstrate this new technique by automatically reconstructing the dorsal surface of a parrotlet wing at 3200 fps during flapping flight. From this shape we analyze key parameters such as wing twist and angle of attack distribution. While our binary 'single-shot' algorithm is demonstrated by quantifying dynamic shape changes of a flying bird, it is generally applicable to moving animals, plants and deforming objects.

Introduction

All flying animals rely to some extent on dynamic body-shape changes to propel themselves. Insects rely predominantly on passive wing morphing through aero-elastic wing deformation (Combes and Daniel, 2003; Wootton, 1992). Bats can actively change the shape of their wings through musculoskeletal control and muscle fibers in their membrane (Cheney et al., 2014). Amongst active flyers, birds can morph their wings to the greatest extent, from fully extended to completely folded in flight (Pennycuick, 2008; Williams and Biewener, 2015), but how they utilize morphing during flapping and maneuvering flight is not fully understood. Such questions have traditionally been addressed by measuring the 3D body kinematics of flying animals using semi-automated marker tracking (Hedrick, 2008; Hedrick et al., 2002; Ros et al., 2015; Tobalske et al., 2007), feature tracking (Biesel et al., 1985; Carruthers et al., 2010; Walker et al., 2009), or visual-hull based reconstruction methods (Fontaine et al., 2009; Muijres et al., 2014; Ristroph et al., 2009). None of these methods, however, can directly and automatically reconstruct the wing surface at high resolution.

Structured-light based methods record the deformation of a projected light pattern due to the animal's surface geometry for offline 3D reconstruction (Fig. 1A), generally by using one of two different pattern encoding techniques. *Temporally encoded* projection patterns require comparison of consecutive frames. Previous studies have shown that slowly moving human body parts and internal organs can be reconstructed using binary (Ackerman et al., 2002; McKeon and Flynn, 2010) and phase shifted (Lohry and Zhang, 2014; Wang et al., 2013) temporal coding. During pilot experiments, we determined this method is too slow to be automated for bird flight. *Spatially encoded* projection patterns can reconstruct a sequence of stand-alone frames and are hence called 'single-shot' (Salvi et al., 2010; Zhang, 2012), which gives the advantage of being robust to inter-frame movement. Some existing spatially encoded structured-

light methods rely on binary pseudo-random dots but either have relatively low frame-rate and accuracy (Saberioon and Cisar, 2016; Sarbolandi et al., 2015) or require manual digitizing of numerous points per frame (Wolf and Konrath, 2015; Zhang et al., 2008). Other existing spatial methods use grayscale patterns which can't be projected at high frame rates (Guan et al., 2003; Lenar et al., 2013; Sagawa et al., 2012; Su and Liu, 2006). Because we found that no existing system can automatically measure dynamic shape changes at sufficiently high speeds, we developed a custom method. This new single-shot structured-light technique can automatically resolve body shape changes at high temporal and spatial resolution.

Materials and methods

1. High-speed 3D surface reconstruction experimental setup

The experimental setup (Fig. 1A) consisted of a 3D calibrated and synchronized high-speed camera (Phantom Miro M310; Vision Research, Wayne, New Jersey, USA) and high-speed projector (DLP® LightCrafter™ E4500MKII™; EKB Technologies Ltd., Bat-Yam, Israel) operating at 3200 fps. Calibration of the system was achieved using a modified version of the camera calibration toolbox for Matlab® (Bouguet, 2000). All data processing was conducted in Matlab® R2015b. We analyzed the first two wingbeats after takeoff of a 4-year-old female near-white Pacific parrotlet (*Forpus coelestis*, 27-29 grams, 0.23 meter wingspan), which was trained using positive reinforcement to fly between perches 0.5 meters apart. All experiments were in accordance with Stanford University's Institutional Animal Care and Use Committee.

2. Design of the single-shot structured-light pattern

To achieve a binary single-shot light pattern for high-speed surface reconstruction, we modified a single-shot structured-light technique (Kawasaki et al., 2008). The projected pattern consists of horizontal and vertical stripes that form a grid, and in the original method, different colors were used to distinguish the horizontal and vertical stripes. We simplified this approach for use at high-speed by removing the color coding, instead relying on image processing to distinguish binary stripes (Fig. 1B). Vertical stripes are equally spaced and densely packed for high spatial resolution while horizontal stripes are unequally spaced to ensure local uniqueness. To enhance robustness, the unequally spaced horizontal stripes can be spaced such that the spacing between every set of four consecutive stripes is unique.

There are four key advantages of using this scheme. First, it is designed for full automation, which allows for high throughput of data. Second, it is single-shot, which is robust for rapidly deforming objects. Third, the scheme uses binary light patterns, which allow the projectors to use their maximum frame rate. Fourth, it uses a single color that allows maximum frame rate and multiple view angles.

Interference can be avoided by using different color channels, light polarization or slightly out-of-phase light pulses.

3. Image processing for identifying and ordering stripes

Before 3D reconstructing the surface, image processing was required to separate the vertical and horizontal stripes in the camera image, order these stripes, and find their intersections. We applied the following fully automated steps (Fig. S1): The camera image was rotated to align the equally spaced stripes vertically. Next, the Laplacian of a directional Gaussian filter was applied in horizontal and vertical directions. Adaptive thresholding was used to generate a noisy approximation of the horizontal and vertical stripes. The noise was filtered out by adding a minimum length requirement for each connected white region. Extension and closure of stripes with gaps was accomplished by choosing paths that best combine attributes of high pixel brightness and correct stripe direction. These factors were weighted, and stripes were only extended if a preset cutoff value was satisfied.

　　　After all stripes were identified, their subpixel center was determined using the original image by quadratically fitting brightness levels perpendicular to the stripe, based on which intersection points between horizontal and vertical stripes were located. Regions near intersections produced inaccurate center lines, so these regions were interpolated and the intersections recomputed. Finally, all stripes were ordered based on connections between stripes, and discontinuous grids were ordered separately.

4. Stripe matching algorithm

To triangulate light on the bird's surface, the *unknown* 3D planes visible from the camera needed to be matched with *known* 3D light planes generated by the projector (Fig. 1B,C). The algorithm described here used one camera and projector, but the same steps can be followed for multiple calibrated cameras and projectors. After completing the image processing steps, variables were organized with lower case letters referring to unknown planes and upper case letters referring to known planes. In the projected pattern, there was an ordered set of M known vertical planes with the unit normal $<A_K,B_K,C_K>$ and an ordered set of N known horizontal planes with the unit normal $<D_L,E_L,F_L>$. From the camera image, there was a set of m unknown vertical planes with unit normal $<a_k,b_k,c_k>$ and n unknown horizontal planes with unit normal $<d_l,e_l,f_l>$. The order of the unknown planes and their 2D intersection points (x_{kl}, y_{kl}), however, was known as long as the grid connected the stripes (minimum four connections required). If there were discontinuities in the grid, separate portions were computed separately. Calibration of the camera and projector produced Eqns 1-3,

$$P_p = K_p \left(RP_w + T \right) \tag{1}$$

$$P_c = K_c P_w \tag{2}$$

$$K = \begin{bmatrix} \alpha & 0 & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

where $P_{c/p}$ are 2D homogeneous camera or projector (c/p) coordinates in pixels, P_w are 3D coordinates, $K_{c/p}$ is the internal calibration matrix defined by Eqn 3 (with constants α, β, u_0 and v_0), R is the rotation matrix from the camera to the projector, and T is the translation matrix from the camera to the projector (where the optical center of the camera lies at the origin). While it was unknown to which projection plane each plane in the camera image corresponded to, two equations could be written per camera plane based on the calibration above, Eqns 4,5. They were derived based on the principle that all the planes intersected at the optical center of the projector. Eqn 6 then followed from all vertical planes intersecting at a vector in space, while Eqn 7 is the equivalent for all horizontal planes.

$$a_k T[1] + b_k T[2] + c_k T[3] = -1 \tag{4}$$

$$d_l T[1] + e_l T[2] + f_l T[3] = -1 \tag{5}$$

$$a_k R[1,2] + b_k R[2,2] + c_k R[3,2] = 0 \tag{6}$$

$$d_l R[1,1] + e_l R[2,1] + f_l R[3,1] = 0 \tag{7}$$

Brackets used in Eqns 4-7 indicate the selection of a specific [Row, Column] value in matrices.

For each intersection of horizontal plane, vertical plane, and the bird, the calibration was used to write Eqns 8,9 where x, y, and z defined the unknown 3D location of the intersection point. Further, we knew which two planes intersected at this point, so we wrote two more equations defining each plane. These four equations combined into a single equation by eliminating x, y, and z (Eqn 10).

$$x = \frac{(x_{kl} - u_{0c})z}{\alpha_c} \tag{8}$$

$$y = \frac{(y_{kl} - v_{0c})z}{\beta_c} \tag{9}$$

$$\frac{(x_{kl} - u_{0c})}{\alpha_c}(a_k - d_l) + \frac{(y_{kl} - v_{0c})}{\beta_c}(b_k - e_l) + (c_k - f_l) = 0 \tag{10}$$

With all known and unknown variables defined and constrained, known and unknown planes could be matched. This was done by 'sliding' the unknown planes onto different known plane positions to determine which position results in a minimal matching error, as visualized in Figs 1D-F. The mathematical analog to this combines Eqns 4-7,10 into one large matrix as seen in Eqn 11,

$$MX = B \tag{11}$$

where *M* and *B* are constant matrices and *X* contains the sought-after unit normal vectors for all ordered horizontal and vertical unknown planes. Because *X* has one degree of freedom, it can be rewritten as Eqn 12,

$$X = X_p + pX_h \tag{12}$$

where X_p is a particular solution of *X*, X_h is the homogenous solution of *X*, and *p* is a variable. The value of *p* is critical as it determines the particular solution of *X* and can be tuned to slide the unknown planes to different positions to reduce matching errors. Singular Value Decomposition was then used as defined in Eqn 13, where σ are the singular values and *U* and *V* are square matrices, to find X_p in Eqn 14 and X_h which is the rightmost column of *V*.

$$M = U * \mathrm{diag}\left(\sigma_1, \sigma_2, \cdots, \sigma_{3n}\right)V^T \tag{13}$$

$$X_p = V * \mathrm{diag}\left(\frac{1}{\sigma_1}, \frac{1}{\sigma_2}, \cdots, \frac{1}{\sigma_{3n-1}}, 0\right)U^T B \tag{14}$$

For each potential match between known and unknown planes, the error was computed as the mean squared angle between the known and unknown planes. The correct matching sequence for the horizontal planes gave a much lower error than other possible matches due to the unequal spacing of the stripes. When the correct matching planes were found for the horizontal planes, the value of *p* used in Eqn 12 was then used to match the vertical planes as well.

5. 3D surface reconstruction and system accuracy

After the unknown planes were matched with the projected planes, 3D reconstruction of the surface was straight-forward. For each stripe seen on the bird, its light plane was defined. Additionally, for each point at the center of a stripe, the 3D vector along which that point must lie was specified. The intersection of the vector and plane lies on the bird's surface. We then fit a surface (average 26907 points) to the point cloud of data (average 285 intersections and 11405 total points) using the Gridfit toolbox in Matlab® (D'Errico, 2005), which uses a modified ridge estimator with tunable smoothing. The result is shown in Fig. 2E for different wingbeat phases reconstructed with a single camera and projector.

The reconstruction accuracy achievable was estimated in a separate test with similar equipment settings. Using a sphere of known radius (22.23 +/- 0.05 s.d. mm), we found an accuracy of 0.31 mm (error) and precision of +/- 1.03 mm (s.d.), see Fig. 2A,B. Errors were largest (double) in areas outside the calibration volume (the image corner regions). Additionally, occasional stripe mismatching occurred in the image processing steps, which accounts for other larger errors (Fig. S3). When processing both the

sphere and bird, no manual manipulation was used and bird reconstruction was successful for 98% of the frames over four separate downstroke segments (two wingbeats of two flights).

6. Calculation of dorsal surface parameters

To calculate bird-specific surface parameters, a reference frame must be defined for each frame. To accomplish this automatically, we first identified the body of the bird by identifying surface points that move less than a preset threshold distance (see Fig. S2). To compute the z axis, the body points were fit to a plane, after which the x axis was computed by finding the line of symmetry of the body points projected on that plane. In order to find a repeatable origin point, the top of the bird's head was found by fitting a paraboloid to points on the head. For the frames we analyze here, the orientation of the axis did not change significantly and was thus set constant for all frames, while the origin point was fit linearly over all relevant frames. This computed body reference frame was labeled subscript b. Another reference frame, the wing reference frame, was used for measuring wing shape and labeled subscript w. It was found by rotating the body reference frame about the x_b axis in order to best fit the (right) wing of the bird. The reference frames and the corresponding surfaces are shown in Fig. 3A,B.

Using these reference frames and the 3D data, surface metrics were computed. In Fig. 3D, the shape of the bird at its midline is tracked while in Fig. 3E-F, the shapes of dorsal airfoil slices of the wing are tracked at different spanwise positions. We determined the angles of attack spanwise along the wing (see Fig. 3G-I) based on these airfoils. The geometric angle of attack was found with a linear fit to each airfoil, while the induced angle of attack was found by computing the direction of the velocity of the wing from root to tip by using the bird velocity and angular velocity of the wing. To reduce noise in these linear fits, outliers were detected using the RANSAC method (Fischler and Bolles, 1981). The effective angle of attack is the difference of the induced and geometric angles of attack. Due to the angle of the bird's wing relative to the camera and projector positions, angle of attack measurements beyond a spanwise position halfway to the wingtip are less reliable. This could be resolved in future setups by adding cameras and projectors to better image the wing under these angles.

Results and discussion

We quantified and analyzed the 3D wing and body shape of a flying bird during a portion of four downstrokes (41% to 64%) using a novel high-speed, automated, 3D structured-light method. While our results are for a single projector and camera combination imaging the dorsal side of the bird, more cameras and projectors can be added to get full body volumetric reconstruction and provide a more complete analysis of wing morphing over an entire wingbeat cycle. In our analysis of the dorsal data, we find that the bird's tail rotates down significantly with respect to the body ($1720° \text{ s}^{-1}$ in video 1, $1550° \text{ s}^{-1}$

in video 2) during the first downstroke after takeoff (Fig. 3D) but not the second (-290° s^{-1} in video 1, 270° s^{-1} in video 2) (Fig. S3; all angular velocities are averaged over the downstroke phase analyzed). The wings rotate down at an average of 5700° s^{-1} in this same portion of the downstroke. Further, the wings are tracked at different spanwise positions (Fig. 3E-F), and we see that the wing twists relative to the body through the downstroke as confirmed by the geometric angle of attack plot (Fig. 3G). Using these data, we computed the effective aerodynamic angle of attack (Fig. 3I) which remains relatively constant between 55° and 75° in the first downstroke and 45° to 60° in the second downstroke (Fig. S3C,F). These high angles during the downstroke at takeoff enable the bird to support its body weight with both lift and drag, while simultaneously generating significant thrust by tilting lift forward. This is facilitated by the close to horizontal orientation of the wing surface moving predominantly downward (Fig. 3A,B) combined with the high lift and drag coefficients of a bird wing at these high angles of attack (Kruyt et al., 2014). The measurements illustrate how the system provides insight into how birds morph their wings to generate aerodynamic forces and could give insight into maneuvering flight in future studies.

Beyond studying birds, this new structured-light method has many benefits for capturing 3D data in experimental biology and engineering. It is automated to allow for a high throughput of data, single-shot to track deforming objects, binary to allow for high-speed tracking, and uses a single color to allow for multiple view angles. While the current application of this technique is for studying bird flight, it is broadly adaptable for tracking shapes of other dynamically morphing animals, plants and objects of interest.

References

Ackerman, J. D., Keller, K. and Fuchs, H. (2002). Surface reconstruction of abdominal organs using laparoscopic structured light for augmented reality. *Three-Dimensional Image Capture Appl.* **4661**, 39–46.

Biesel, W., Butz, H. and Nachtigall, W. (1985). Erste Messungen Der Flogelgeometrie Bei Frei Gleitflie-Genden Haustauben (Columba Livia Var. Domestica) Unter Benutzung Neu Ausgearbeiteter Verfahren Der Windkanal-Technik Und Der Stereophotogrammetrie. *Biona-Report 3* **3**, 138–160.

Bouguet, J. Y. (2000). Camera Calibration Toolbox for Matlab.

Carruthers, A. C., Walker, S. M., Thomas, A. L. R. and Taylor, G. K. (2010). Aerodynamics of aerofoil sections measured on a free-flying bird. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **224**, 855–864.

Cheney, J. A., Konow, N., Middleton, K. M., Breuer, K. S., Roberts, T. J., Giblin, E. L. and Swartz, S. M. (2014). Membrane muscle function in the compliant wings of bats. *Bioinspir. Biomim.* **9**, 25007.

Combes, S. A. and Daniel, T. L. (2003). Into thin air: Contributions of aerodynamic and inertial-elastic forces to wing bending in the hawkmoth Manduca sexta. *J. Exp. Biol.* **206**, 2999–3006.

D'Errico, J. (2005). Surface Fitting using gridfit.

Fischler, M. A. and Bolles, R. C. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applicatlons to Image Analysis and Automated Cartography. *Commun. ACM* **24**, 381–395.

Fontaine, E. I., Zabala, F., Dickinson, M. H. and Burdick, J. W. (2009). Wing and body motion during flight initiation in Drosophila revealed by automated visual tracking. *J. Exp. Biol.* **212**, 1307–23.

Guan, C., Hassebrook, L. and Lau, D. L. (2003). Composite structured light pattern for three-dimensional video. *Opt. Express* **11**, 406–417.

Hedrick, T. L. (2008). Software techniques for two- and three-dimensional kinematic measurements of biological and biomimetic systems. *Bioinspir. Biomim.* **3**, 34001.

Hedrick, T. L., Tobalske, B. W. and Biewener, A. A. (2002). Estimates of circulation and gait change based on a three-dimensional kinematic analysis of flight in cockatiels (Nymphicus hollandicus) and ringed turtle-doves (Streptopelia risoria). *J. Exp. Biol.* **205**, 1389–1409.

Kawasaki, H., Furukawa, R., Sagawa, R. and Yagi, Y. (2008). Dynamic scene shape reconstruction using a single structured light pattern. *IEEE Conf. Comput. Vis. Pattern Recognit.* 2806–2813.

Kruyt, J. W., Quicazán-Rubio, E. M., van Heijst, G. F., Altshuler, D. L. and Lentink, D. (2014). Hummingbird wing efficacy depends on aspect ratio and compares with helicopter rotors. *J. R. Soc. Interface* **11**, 570–81.

Lenar, J., Witkowski, M., Carbone, V., Kolk, S., Adamczyk, M., Sitnik, R., van der Krogt, M. and Verdonschot, N. (2013). Lower body kinematics evaluation based on a multidirectional four-dimensional structured light measurement. *J. Biomed. Opt.* **18**, 56014.

Lohry, W. and Zhang, S. (2014). High-speed absolute three-dimensional shape measurement using three binary dithered patterns. *Opt. Express* **22**, 26752.

McKeon, R. T. and Flynn, P. J. (2010). Three-dimensional facial imaging using a static light screen (SLS) and a dynamic subject. *IEEE Trans. Instrum. Meas.* **59**, 774–783.

Muijres, F. T., Elzinga, M. J., Melis, J. M. and Dickinson, M. H. (2014). Flies Evade Looming Targets by Executing Rapid Visually Directed Banked Turns. *Science (80-.).* **344**, 172–177.

Pennycuick, C. J. (2008). *Modeling the Flying Bird*. Academic Press.

Ristroph, L., Berman, G. J., Bergou, A. J., Wang, Z. . J. and Cohen, I. (2009). Automated hull reconstruction motion tracking (HRMT) applied to sideways maneuvers of free-flying insects. *J. Exp. Biol.* **212**, 1324–1335.

Ros, I. G., Badger, M. A., Pierson, A. N., Bassman, L. C. and Biewener, A. A. (2015). Pigeons produce aerodynamic torques through changes in wing trajectory during low speed aerial turns. *J. Exp. Biol.* **218**, 480–490.

Saberioon, M. M. and Cisar, P. (2016). Automated multiple fish tracking in three-Dimension using a Structured Light Sensor. *Comput. Electron. Agric.* **121**, 215–221.

Sagawa, R., Sakashita, K., Kasuya, N., Kawasaki, H., Furukawa, R. and Yagi, Y. (2012). Grid-based active stereo with single-colored wave pattern for dense one-shot 3D scan. *IEEE Int. Conf. 3D Imaging, Model. Process. Vis. Transm.* 363–370.

Salvi, J., Fernandez, S., Pribanic, T. and Llado, X. (2010). A state of the art in structured light patterns for surface profilometry. *Pattern Recognit.* **43**, 2666–2680.

Sarbolandi, H., Lefloch, D. and Kolb, A. (2015). Kinect range sensing: Structured-light versus Time-of-Flight Kinect. *Comput. Vis. Image Underst.* **139**, 1–20.

Su, W.-H. and Liu, H. (2006). Calibration-based two-frequency projected fringe profilometry: a robust, accurate, and single-shot measurement for objects with large depth discontinuities. *Opt. Express* **14**, 9178–9187.

Tobalske, B. W., Warrick, D. R., Clark, C. J., Powers, D. R., Hedrick, T. L., Hyder, G. A. and

Biewener, A. A. (2007). Three-dimensional kinematics of hummingbird flight. *J. Exp. Biol.* **210**, 2368–2382.

Walker, S. M., Thomas, A. L. R. and Taylor, G. K. (2009). Deformable wing kinematics in the desert locust: how and why do camber, twist and topography vary through the stroke? *J. R. Soc. Interface* **6**, 735–47.

Wang, Y., Laughner, J. I., Efimov, I. R. and Zhang, S. (2013). 3D absolute shape measurement of live rabbit hearts with a superfast two-frequency phase-shifting technique. *Opt. Express* **21**, 6631–6636.

Williams, C. D. and Biewener, A. A. (2015). Pigeons trade efficiency for stability in response to level of challenge during confined flight. *Proc. Natl. Acad. Sci. U. S. A.* **2015**, 201407298.

Wolf, T. and Konrath, R. (2015). Avian wing geometry and kinematics of a free-flying barn owl in flapping flight. *Exp. Fluids* **56**,.

Wootton, R. J. (1992). Functional Morphology of Insect Wings. *Entomol* 113–40.

Zhang, Z. H. (2012). Review of single-shot 3D shape measurement by phase calculation-based fringe projection techniques. *Opt. Lasers Eng.* **50**, 1097–1106.

Zhang, G., Sun, J., Chen, D. and Wang, Y. (2008). Flapping motion measurement of honeybee bilateral wings using four virtual structured-light sensors. *Sensors Actuators, A Phys.* **148**, 19–27.
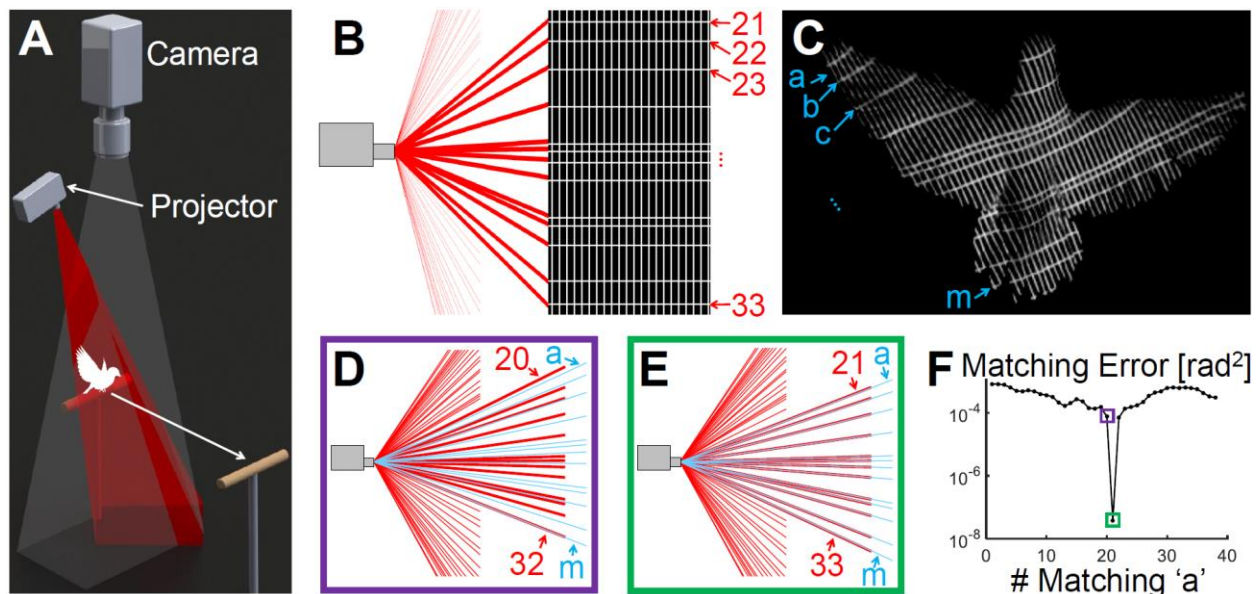
Figure legends

Fig. 1. We developed a new high-speed 3D surface reconstruction technique for rapidly locomoting animals based on binary spatially-encoded structured-light. (A) The 3D calibrated projection of a structured-light pattern on a flying bird is imaged by a high-speed camera. (B) The projected pattern is

shown in black and white, while the color red is used to indicate the known lines and numbers of horizontally projected stripes (we show a side view of horizontal projection planes). Horizontal stripes are spaced unequally to ensure local uniqueness of the pattern, while vertical stripes are equally spaced for high spatial resolution. (C) The images captured by the camera with unknown horizontal stripes are labeled in blue. (D,E) Ordered but unidentified horizontal stripes (blue letters and lines) are matched with an ordered subset of known projection stripes (red numbers & lines). In (D), the blue and red lines do not match, whereas in (E) they match. (F) Horizontal stripe matching error as a function of matching permutation, including the two boxed examples shown in (D) and (E). The error is computed as the mean squared angle in radians between matching stripes. Note that the stripe numbering and lettering convention in this figure do not match the equations given in the text, they serve illustrative purpose only.
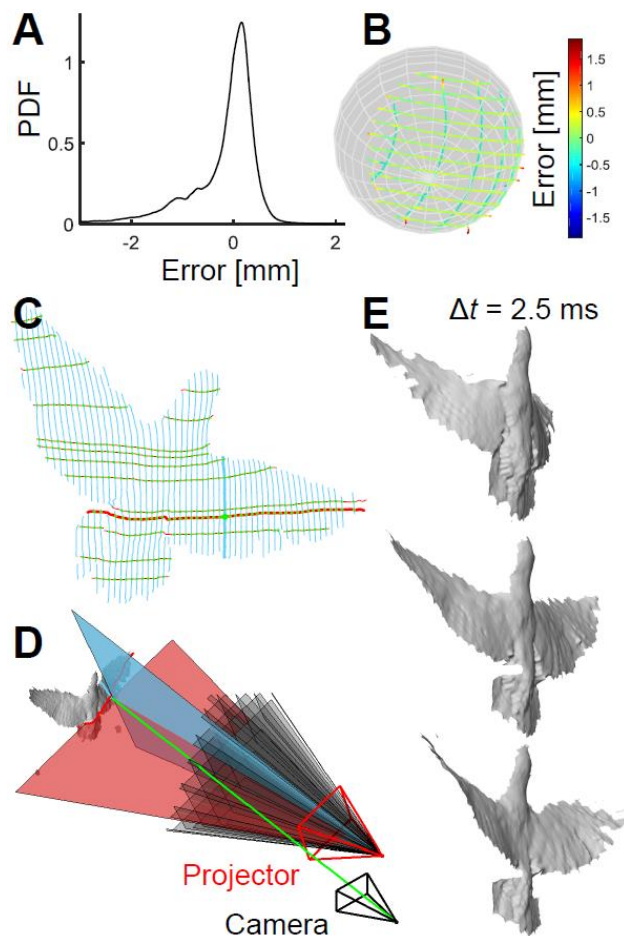

Fig. 2. Single-shot 3D surface reconstruction accuracy and results. (A-B) Method verification tests using a 22.23 +/- 0.05 (s.d.) mm radius sphere results in a reconstruction error of 0.31 +/- 1.03 (s.d.) mm averaged over sphere location for 400 frames, showing the new method is both accurate and precise. In (A) the probability density function (PDF) of the error in the measured radius is shown. (C) To

reconstruct a surface, horizontal and vertical stripes are separated using image processing techniques (Fig. S1). Next, all intersections of these stripes are calculated (one example is shown as a green dot). (D) Each stripe seen in the camera represents a 3D projected plane of light and each 2D intersection point represents a vector in 3D space on which the 3D intersection on the bird's surface lies. A sampling of projection planes is shown in black, while single horizontal and vertical planes are extended along with their intersection point on the bird's surfaces. Color coding corresponds to Figure 2C. (E) 3D surface reconstruction of the flapping wing (and body) of a bird in flight at 2.5 ms intervals (33%, 44%, and 55% of its first downstroke after takeoff).
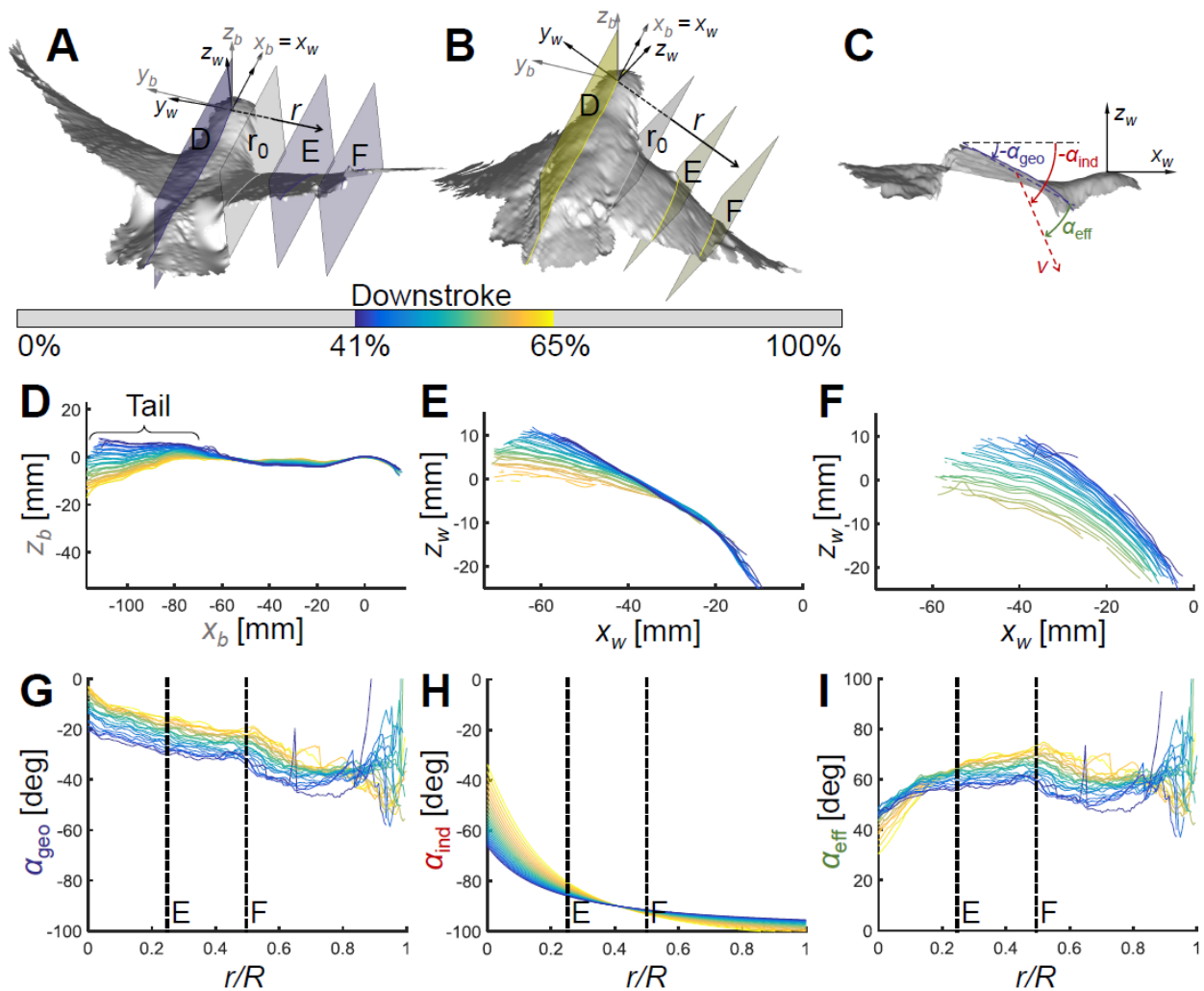

Fig. 3. The dorsal wing profile, twist and aerodynamic angle of attack of a bird is automatically tracked throughout a section of the downstroke for which the wing is fully in view. (A-B) 3D upper surface of a parrotlet at 41% and 65% of its first downstroke after takeoff respectively. Two reference frames are defined as a function of time: a translating *body* reference frame (subscript *b*) with z_b pointing

up and x_b pointing forward, and a translating and rotating *wing* reference frame (subscript w). The origin for both reference frames is the top of the head. The wing reference frame is rotated about x_b so it is parallel with the right wing. The r axis is parallel to y_w and begins at the plane labeled D, which intersects the bird's right shoulder. (C) Side view of bird shown in (A), illustrating the definition of geometric, induced, and effective angle of attack. (D) The shape of the bird at the centerline of its dorsal surface $(y_b = 0)$, where the purple line corresponds to (A) and the yellow line corresponds to (B). (E-F) The dorsal profile shape of the bird wing as a function of wing radius, r, equal to $0.25R$ and $0.50R$ where R is the length of the right wing from the plane r_0 to the wingtip and is held constant across all frames. The value r is equal to zero at the plane labeled r_0. (G) The geometric angle of attack (twist) of the wing with respect to the body of the bird is approximately linear. (H) Induced angle of attack of the wing due to bird forward and wing flapping velocity based on the average spanwise velocity of center of the wing. (I) Effective angle of attack calculated as the geometric angle of attack minus the induced angle of attack.
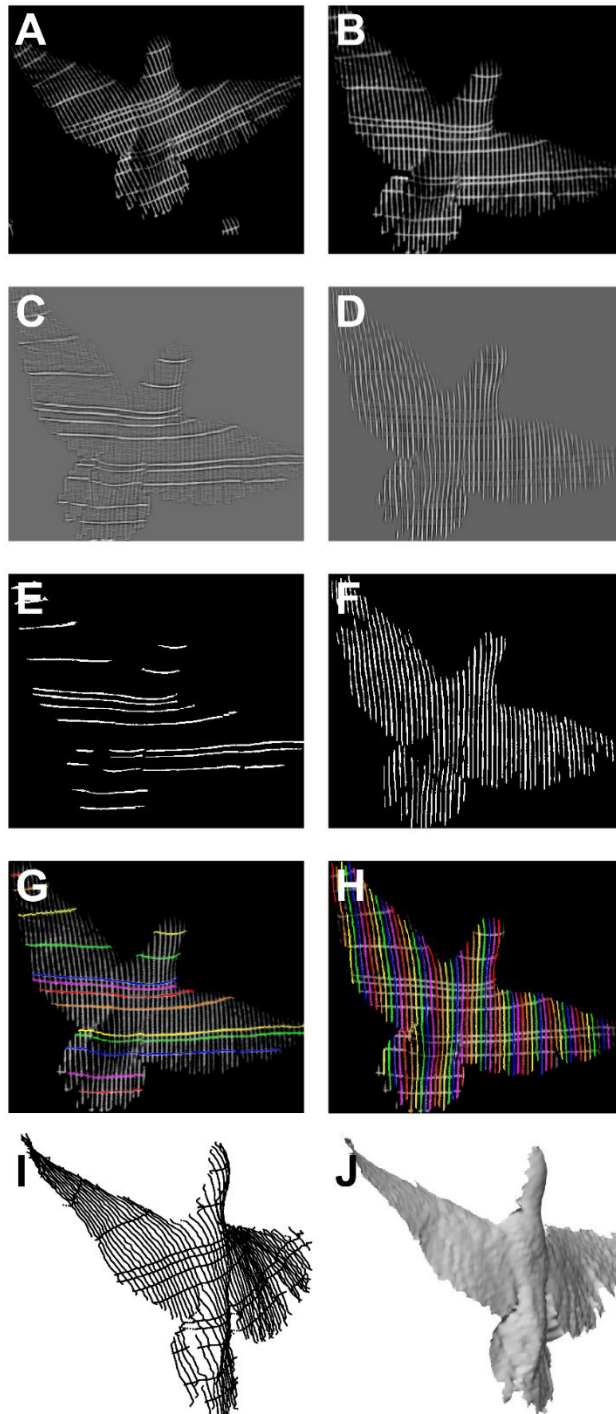
Fig. S1. Image processing is used to separate and order the horizontal and vertical stripes necessary to achieve 3D reconstruction the flying bird. (A) Cropped version of the original image of projected stripes on the bird captured by the high-speed camera. (B) The original image is first rotated so equally

spaced stripes are vertical. There is a depth edge between the bottom of the left wing and the body which is manually separated by coloring it black. (C-D) The Laplacian of a directional Gaussian filter is applied to image (B). In (C-H) horizontal stripes are identified in the left figures while vertical stripes are identified in the right figures. (E-F) Automated locally adaptive thresholding is applied in order to identify the highest image intensity values (C-D) and only the connected white regions exceeding a minimum directional length requirement are kept. (G-H) Stripes from images (E-F) are ordered in a repeating rainbow pattern using the intersection data between horizontal and vertical stripes. (I) The raw 3D surface data for a single frame. (J) The surface fit to the 3D data.
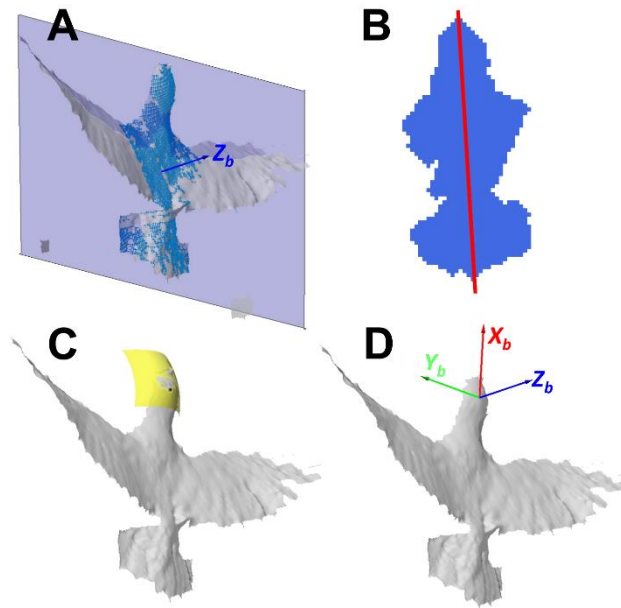

Fig. S2. Automated definition of a bird-fixed reference frame. (A) Surface points are identified that move less than a preset threshold distance in order to filter out the wings. The unit normal of a plane fit to these points is defined as the z_b axis. (B) These points are projected onto the fit plane, and the line of symmetry is found and defined as the x_b axis. The y_b axis is defined as the cross product of the z_b and x_b axes. These axes directions are held constant as a function of time. (C) Points on the head of the bird are used to find the origin of the reference frame by fitting a paraboloid to these data points and finding the top point. (D) The final body reference frame only translates as a function of time and does so linearly.
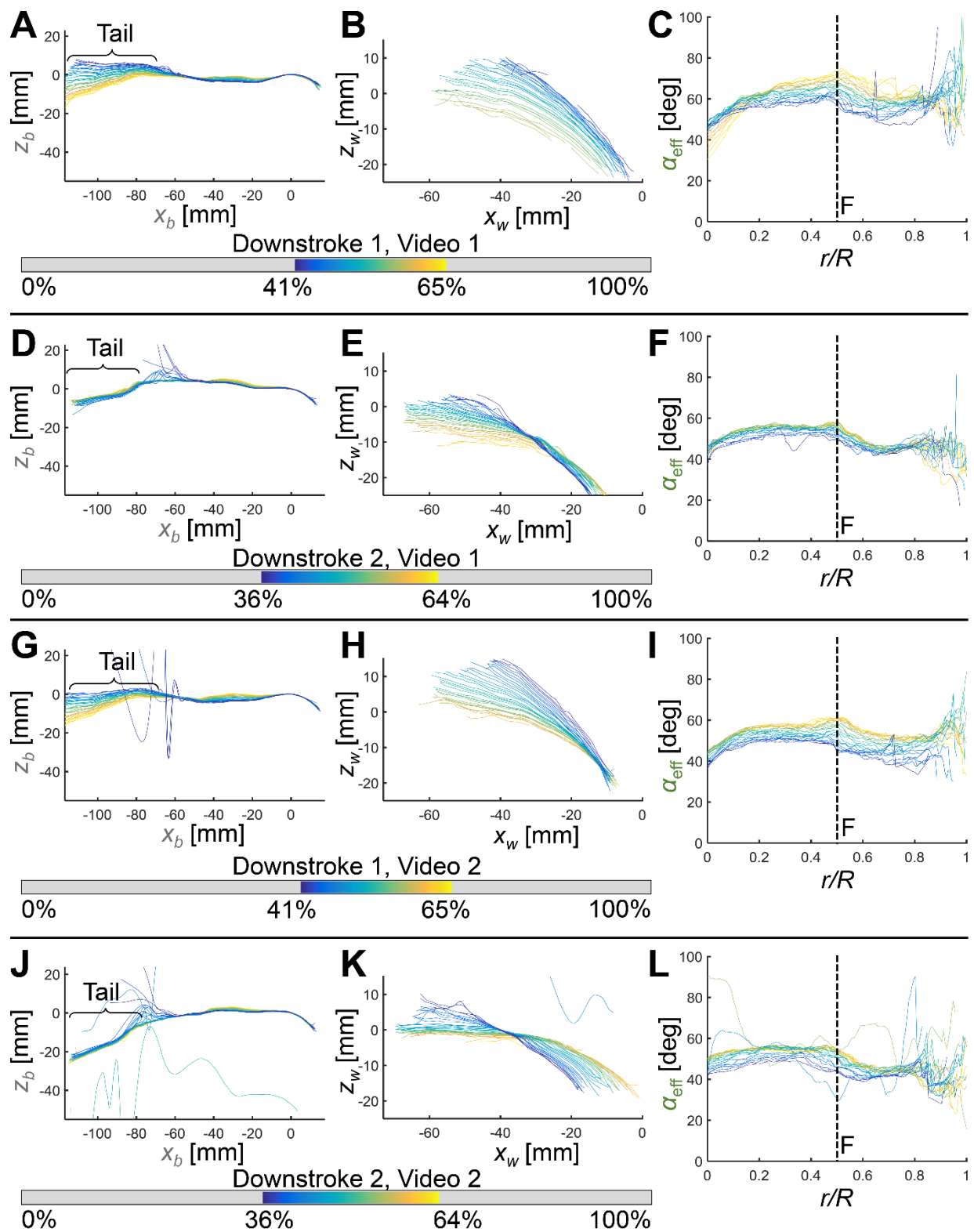
Fig. S3. Three plots from Fig. 3 are shown for each of 4 downstrokes. (A-C) The first downstroke (41% to 65% of the downstroke) after takeoff in video 1. (D-F) The second downstroke (36% to 64% of the downstroke) after takeoff in video 1. (G-I) The first downstroke (41% to 65% of the downstroke) after

takeoff in video 2. (J-L) The second downstroke (36% to 64% of the downstroke) after takeoff in video 2. For both videos, the first downstroke was 28.6 μs and the second downstroke was 24.2 μs. The leftmost plots are the same as Fig. 3D for their respective downstrokes. The middle plots are the same as Fig. 3F for their respective downstrokes. The rightmost plots are the same as Fig. 3I for their respective downstrokes. Erroneous lines are due to occasional local stripe mismatching occurring in the image processing steps, but overall the automated 3D reconstruction is robust and accurate.

Table S1. Variable definitions. Brackets indicate the size of matrices [rows x columns] or the cell number of matrices [row, column], parentheses are used for coordinates, inequality signs are used for vectors <x,y,z>, and 'pix' stands for pixels.

Variable	Units	Description/Equation
Planes – known (projected)		
$<A_K, B_K, C_K>$	-	Unit normal of Kth vertical known plane
M	-	Number of vertical known planes
$<D_L, E_L, F_L>$	-	Unit normal of Lth horizontal known plane
N	-	Number of horizontal known planes
Planes – unknown (camera image)		
$<a_k, b_k, c_k>$	-	Unit normal of kth vertical unknown plane
m	-	Number of vertical unknown planes
$<d_l, e_l, f_l>$	-	Unit normal of lth horizontal unknown plane
n	-	Number of horizontal unknown planes
Points		
P_c	pix_{cam}	2D homogeneous camera coordinates [3×1]
P_p	pix_{proj}	2D homogeneous projector coordinates [3×1]
P_w	mm	3D world coordinates [3×1]
(x_{kl}, y_{kl})	pix_{cam}	2D intersection of kth vertical and l^{th} horizontal planes
Calibration		
K_c	pix_{cam}	Internal camera calibration matrix [3×3]
α_c	pix_{cam}	Camera focal length (x), $\alpha_c = K_c[1,1]$
β_c	pix_{cam}	Camera focal length (y), $\beta_c = K_c[2,2]$
u_{0c}	pix_{cam}	Camera principal point (x), $u_{0c} = K_c[1,3]$
v_{0c}	pix_{cam}	Camera principal point (y), $v_{0c} = K_c[2,3]$
K_p	pix_{proj}	Internal projector calibration matrix [3×3]
α_p	pix_{proj}	Projector focal length (x), $\alpha_p = K_p[1,1]$
β_p	pix_{proj}	Projector focal length (y), $\beta_p = K_p[2,2]$
u_{0p}	pix_{proj}	Projector principal point (x), $u_{0p} = K_p[1,3]$
v_{0p}	pix_{proj}	Projector principal point (y), $v_{0p} = K_p[2,3]$
R	-	Rotation matrix: camera to projector [3×3]

T	mm	Translation matrix: camera to projector [3×1]
Singular Value Decomposition		
X	-	Contains m vertical (a,b,c) and n horizontal planes (d,e,f) [$m+n \times 1$]
X_p	-	A particular solution of X [$m+n \times 1$]
X_h	-	The homogeneous solution of X [$m+n \times 1$]
p	-	Tuning parameter to choose a particular solution for X
B	-	Constant matrix [$2(m+n)+$(#intersections) $\times 1$]
M	-	Constant matrix [$2(m+n)+$(#intersections) $\times m+n$] = [$i \times j$]
U	-	[$i \times i$] unitary matrix of SVD(M)
σ_s	-	s^{th} singular value M of j values
V	-	[$j \times j$] unitary matrix of SVD(M)