This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TGRS.2022.3141907, IEEE Transactions on Geoscience and Remote Sensing

MegaStitch: Robust Large Scale Image Stitching

Ariyan Zarei, Emmanuel Gonzalez, Nirav Merchant, Duke Pauli, Eric Lyons, and Kobus Barnard

***Abstract*—We address fast image stitching for large image collections while being robust to drift due to chaining transformations and minimal overlap between images. We focus on scientific applications where ground truth accuracy is far more important than visual appearance or projection error, which can be misleading. For common large-scale image stitching use cases, transformations between images are often restricted to similarity or translation. When homography is used in these cases, the odds of being trapped in a poor local minimum and producing unnatural results increases. Thus, for transformations up to affine, we cast stitching as minimizing reprojection error globally using linear least squares with a few, simple constraints. For homography, we observe that the global affine solution provides better initialization for bundle adjustment compared to an alternative that initializes with a homography-based scaffolding, and at lower computational cost. We evaluate our methods on a very large translation dataset with limited overlap, as well as four drone datasets. We show that our approach is better compared to alternative methods such as MGRAPH in terms of computational cost, scaling to large numbers of images, and robustness to drift. We also contribute ground truth datasets for this endeavor.**

I. Introduction

AUTOMATED crop monitoring and high-throughput phenotyping have become important research topics both in plant sciences and in computer science [1] [2] [3] [4] [5]. Unoccupied aerial vehicles (UAVs) and large-scale, ground-based systems are now providing high resolution alternatives to aerial and satellite image capture. Thus, having accurately georeferenced image mosaics with large fields of view encompassing all parts of the monitored region is important for remote sensing, automatic phenotype extraction, and crop monitoring systems.

The fundamentals of image stitching have been well studied and documented in computer vision. However, there has been less effort on the challenges of image alignment and georeferencing in large scale datasets, and where minimal overlap of neighboring images makes global stitching brittle. Our motivating context is assessing how different water-stress treatments of $40,000$ individual plants of 240 different genotypes affect measured phenotypic features. This requires individual plant tracking throughout an entire growing season, across different types of image data captured by different cameras and sensors including RGB, Thermal and Photosystem II (PS2) camera (a camera that measures plant tissue fluorescence at night to determine photosynthetic capabilities). Here, having accurately georeferenced mosaics is essential for tracking individual plants, fusing data across sensors and time, extracting phenotypic features, and inferring plant performance.

In this project, sensors and cameras are mounted on a specialized, ground-based gantry system that scans two acres throughout the entire season day and night (see Figure 1). Each scanned image is associated with gantry coordinates, which have non-negligible error. This error increase when the gantry coordinates get converted to GPS. The RGB images need to be very high resolution and are thus taken close to the ground ($2-3.5m$). As a result, the images include large regions of soil. In addition, the plants are aligned in uniform rows, and due to the design of the gantry system, the images have as little as 10% overlap. Both these attributes increase the ambiguity of visual feature matches between image pairs. Additionally, the orthomosaic must have a high degree of accuracy as it will be used to track approximately $40,000$ individual plants from $10,000$ images per day throughout the growing season. Due to these factors, minor local errors in pairwise image stitching can easily contribute to major errors in estimated quantitative phenotypes. Further, images need to be accurately aligned to absolute coordinates, and thus to other sensor modalities and field measurements, which is a different task than having results that are visually appealing. Arranging this manually entails far too much human intervention for the scale of these continuously collected data.

Current approaches to image stitching [6] [7] [8] [9] [10] often rely on robust pairwise image matching typically from combining a geometry model with invariant features (e.g., SIFT [11]) using RANSAC [12]. A recently studied alternative [13] [14] uses on-board camera parameter measurements to estimate the transformations, together with a carefully tuned robust estimator to deal with outliers, thereby obviating the need for RANSAC and its associated costs. When working with pairwise transformations, one can consider chaining them together to create a large orthomosaic. However, doing so inevitably suffers from drift (error accumulation) leading to global inconsistencies in position. In particular, there is no reason to expect that chains along two paths between distant images will give the exact same transformation. This

A. Zarei was with the Department of Computer Science, University of Arizona, Tucson, AZ, 85719 USA e-mail: ariyanzarei@email.arizona.edu

E. Gonzalez was with the School of Plant Science, University of Arizona, Tucson, AZ, 85719 USA e-mail: emmanuelgonzalez@email.arizona.edu

N. Merchant was with the Data Science Institute, University of Arizona, Tucson, AZ, 85719 USA e-mail: nirav@email.arizona.edu

D. Pauli was with the School of Plant Science, University of Arizona, Tucson, AZ, 85719 USA e-mail: dukepauli@email.arizona.edu

E. Lyons was with the School of Plant Science, University of Arizona, Tucson, AZ, 85719 USA e-mail: ericlyons@email.arizona.edu

K. Barnard was with the Department of Computer Science, University of Arizona, Tucson, AZ, 85719 USA e-mail: kobus@email.arizona.edu

can be ignored to some extent if the transformations are very accurate, as is possible if there are plenty of pairwise feature matches due to significant image overlap. Generally, and definitely for challenging imaging data, there is a need for a more global approach that can make paths consistent and make image placement more accurate.

The usual solution for improving mosaics built from pairwise homographies is bundle adjustment (e.g. [15]), which entails a non-linear optimization that potentially has many local minima. As such, bundle adjustment requires a good starting point, which might be hard to find or need human intervention [16]. Additionally, bundle adjustment operates on large matrices with each row representing an equation for each of the keypoint correspondences. When the scale of the problem is very large, this becomes a computational bottleneck. However, there has been some efforts to alleviate these problems by taking advantage of inaccurate available metadata (e.g., [13] [14]). Regardless, because of these computational challenges, one certainly does not want to use methods designed for homography if homography is not warranted.

In this paper we show that non-projective transformations permit fast global solutions for minimizing feature matching error using linear least squares (§III). While linear least squares is not robust in general, we are able to use it to an advantage because we only apply it to inliers found by robust matching (e.g., RANSAC). We demonstrate that the proposed method works very well on five datasets, three of which are images of an agricultural field. The one dataset, collected by the gantry system (Figure 1), captures the field with $\sim 6,000 - 10,000$ very high resolution but minimally overlapping images which are related by a translation. The second and the third datasets image the same field with about 450 drone images that are related by a similarity transformation (the drone is relatively level during image capture). The other two datasets are drone images of a golf course and a reservoir in Colorado available online [17].

When the images are related by a homography, instead of initializing bundle adjustment parameters with iterative or graph-based approximation methods (e.g., [18]), which are prone to drift and often lead to unacceptable results, we use the results of our proposed method (§III) to estimate a good initialization for the bundle adjustment. More specifically, we assume affine transformations between the images and solve the proposed linear least squares optimization and use the result as an initialization for the bundle adjustment. This effectively decreases the computation time while maintaining an acceptable level of alignment accuracy.

II. Related work

Early progress in image stitching in computer vision is summarized by Szeliski [19]. Image mosaicking methods typically have four components: 1) feature detection; 2) feature correspondence estimation; 3) transformation estimation and global alignment; and 4) seamless stitching and blending [20]. However, scientific applications need accurate alignment of geo-referenced images, but do not necessarily need absolutely seamless stitching. For homography transformations, a significant step forward was using robust matching of invariant features (e.g., SIFT [11]) under a geometry model using RANSAC [12], as proposed by Brown and Lowe [21], [22], and followed on by many others. Multiple researchers (e.g., [23], [24], [25]) opted for using the Harris-Laplacian detector [26] to detect feature points, and some (e.g., [27], [28]) found advantages to using speeded up robust features (SURF) [29]. To evaluate matches within the RANSAC framework, in addition to the nearest-neighbour based methods proposed originally, Zhao et al. [23] considered normalized correlation, and De Cesare et al. [24] proposed entropy and mutual information based measures.

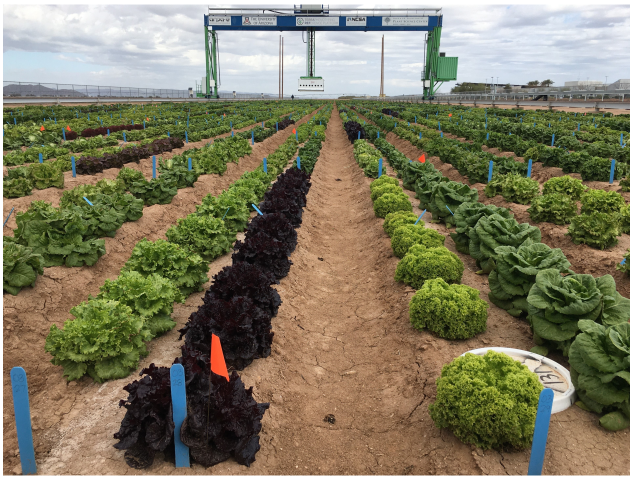


Fig. 1: The gantry system scanning lettuce plants. Different crops are grown under the gantry and scanned daily using various high-resolution sensors and scanners seen hanging below the cross beam and able to move left-to-right as the rig moves forward and back. In the bottom right corner, one can see a GPS marker that we use to develop ground truth for data from this device, as well from the drone.

Different from the above methods, Xie et al. [30] used the fast Fourier transform to estimate the displacement between two images, and subsequently estimate the transformation needed for stitching. And Preibishch et al. [31] used Fourier matching together with global optimization for translated confocal microscopy images.

Related work on orthomosaic generation from large-scale, geo-referenced images includes Mizotin et al. [32], who proposed a voting scheme for shift and rotation estimation in the mosaicing of aerial images with low overlap and significant angle rotation, Xiang et al. [33] who emphasized the importance of using GPS coordinates to find neighboring images to speed up the mosaicing by avoiding matching all possible image pairs, and Moussa et al. [34] who proposed an iterative, region growing approach which combines using GPS coordinates with constrained Delaunay triangulation [35] to avoid exhaustive matching. Although some image mosaicing challenges were addressed in the later method, it accumulates small transformation errors in locations distal from the center (seed) of the mosaic. Similar iterative methods have been proposed by others [36], [37], [38], [39], [40].

In the case of mosaicking drone images, others have used the positional information system (POS) to correct drone's attitude before performing bundle adjustment [41]. Additionally, Liu et al. [42] proposed a new approach in which they constructed each of the projection matrices using the POS information and separately estimated geometric and camera parameter errors.

Finally, Ruiz et al. [18] proposed MGRAPH, which attempts to reduce drift by using a non-linear optimization similar to bundle adjustment. They represent the image dataset by a minimum spanning tree (MST) computed using pairwise matching errors, and then estimate absolute homographies for each image with respect to a reference by chaining the pairwise transformations along the MST paths. These absolute homographies are refined as to minimize the error between matched points computed by transforming one of them to the reference image coordinates, followed by the inverse mapping to the other image. They cast their method in terms of homography, although, as near as we can tell, they use similarity in practice for their drone data. Because they explicitly seek a globally consistent solution, we implemented their method against which to compare our method.

In summary, there has been good progress on managing computation, finding initial matches, iterative stitching, and reducing computational costs assuming homography is needed. What remains is dealing with drift in large scale data, which we address using global optimization for non-projective transformations, either as an appropriate assumption for many scientific data sets, or as an effective initialization for bundle adjustment in cases when homography is needed.

III. Algorithms

For non-projective transformations such as translation, similarity, and affine, we can directly minimize the total reprojection error with constrained linear least squares. Without loss of generality, given a reference image indexed by 0, we denote the 2×3 transformation that rewrites the coordinates with respect to image i into the reference image coordinates by $T^{(i)}$, and use $T_r^{(i)\top}$ for row r of this transformation as a column vectors. We constrain $T^{(0)}$ to be the identity transform, so $T_1^{(0)} = [1, 0, 0]$ and $T_2^{(0)} = [0, 1, 0]$. Our variables are then the stacked rows of $T^{(i)}$, i.e., $[T_1^{(0)\top},\ T_2^{(0)\top},\ T_1^{(1)\top},\ T_2^{(1)\top},\ \ldots T_1^{(N)\top},\ T_2^{(N)\top}]$, where N is the number of images.

We denote the homogeneous coordinates of an arbitrary inlier feature point in image i by $p^{(i)}$, and the set of pairs of corresponding inliers for image pair i and j by $I_{i,j}$. For an inlier pair $(p^{(i)}, p^{(j)}) \in I_{i,j}$, the mapping from $p^{(i)}$ to absolute coordinates should be the close to the mapping from $p^{(j)}$ to absolute coordinates, i.e., $T^{(i)}p^{(i)} \approx T^{(j)}p^{(j)}$. This gives an equation for each of the two transformation rows, and we get the following dot products for the system of equations that we will solve in the least squares sense:

$$\begin{array}{ll} p^{(i)} \bullet T_1^{(i)} - p^{(j)} \bullet T_1^{(j)} \approx 0 & \forall (p^{(i)}, p^{(j)}) \in I_{i,j} \\ p^{(i)} \bullet T_2^{(i)} - p^{(j)} \bullet T_2^{(j)} \approx 0 & \forall (p^{(i)}, p^{(j)}) \in I_{i,j} \quad . \end{array} \tag{1}$$

Solving for affine T gives us the absolute transformations which we can use to align and warp the images into the reference frame.

However, as discussed above, often similarity or translation is called for, and for these cases, we need additional constraints, and we can make additional simplifications. In what follows, we will further index element j in row i of T_i as $T_{i,j}$. For similarity, we augment (1) with the constraints:

$$T_{1,2}^{(\circ)} = -T_{2,1}^{(\circ)} \text{ and } T_{1,1}^{(\circ)} = T_{2,2}^{(\circ)} \quad , \tag{2}$$

using $\circ$ for i or j. Note that if we did not enforce these constraints, unintentionally we would have solved for affine transformation. To reduce the size of the least squares problem, we use shared variables for $T_{1,2}^{(\circ)}$, $T_{2,1}^{(\circ)}$ and $T_{1,1}^{(\circ)}$, $T_{2,2}^{(\circ)}$, negating two of the coefficients in (1) to account for the negation in $T_{1,2}^{(\circ)} = -T_{2,1}^{(\circ)}$.

For translation, we can further simplify the equations by only considering the translation parameters for the x and y directions. This gives:

$$\begin{array}{l} T_{1,3}^{(i)} - T_{1,3}^{(j)} \approx p_x^{(i)} - p_x^{(j)} \\ T_{2,3}^{(i)} - T_{2,3}^{(j)} \approx p_y^{(i)} - p_y^{(j)} \quad . \end{array} \tag{3}$$

This has far fewer parameters and therefore can be solved faster. Note that in order to be robust to drift, we need to keep using the absolute transformations T and avoid chaining pairwise transformations. This permits the least squares optimization to find the best parameters considering all pairwise transformations. While it is known that translation can be addressed with least squares [20], this formulation has not been exploited for very large-scale image stitching problems.

Corner point oriented translation. If we have noisy corner point location estimates (e.g., GPS), as are available from the gantry data collection system, we can incorporate that information in our formulation. Incorporating these priors is easier if we recast the above equation in terms of corners, $c_k^{(i)}$, indexed by k where $k = 1$ is top-left, $k = 2$ is top-right, $k = 3$ is bottom-left, and $k = 4$ is bottom-right corner of image i. We use $\hat{c}_k^{(i)}$ for noisy corner measurements, and assume that we have measured σ_{GPS} which is the ratio of the standard deviation of the point location estimates to the standard deviation of transformation estimation. We use this ratio to inversely weight the matrix rows for corner estimation.

Similarly, if we have a ground truth (GT) anchor location, $a^{(i)}$ for image i, we want to constrain the result so that the pixel corresponding to that anchor has that location. We denote the constrained pixel coordinates for $a^{(i)}$ by (h, w), where h counts down from the top, and w counts rightwards from the left, in images with height H and width W. The corner coordinates in the reference coordinate system are then constrained by the anchor via:

$$\sum_{k=1}^{4} \phi(k, h, w) c_k^{(i)} = a^{(i)} \quad , \tag{4}$$

where

$$\begin{aligned}\phi(1,h,w) &= (1-\frac{h}{H})(1-\frac{w}{W})\\ \phi(2,h,w) &= (1-\frac{h}{H})(\frac{w}{W})\\ \phi(3,h,w) &= (\frac{h}{H})(1-\frac{w}{W})\\ \phi(4,h,w) &= (\frac{h}{H})(\frac{w}{W}) \quad .\end{aligned} \tag{5}$$

rewrites anchor and keypoints locations in terms of the four corners of the corresponding image, which are the variables for which we are solving. This construction works because affine transformations preserve the convex combination (please refer to supplementary materials (§VIII)). We use the same construct to write the reprojection error as the difference of two convex combinations coming from the two mappings for inlier pairs. Finally, for translation, we ensure that the corner points are consistent with a fixed rectangle being a translated image in the anchoring coordinate system. Here we constrain the corners to have the same first/second coordinate as its horizontal/vertical neighbor (third and forth rows of 6). This prevents rectangles from being deformed during the optimization process. Note that for other cases rather than translation, we do not need this constraint.

Allowing for all sources of information, the restructured formulation for translation is:

$$\begin{aligned}& c_k^{(i)} \approx \frac{1}{\sigma_{GPS}} \hat{c}_k^{(i)} \quad \forall i,k && \text{(noisy corners)}\\ & \sum_{k=1}^{4} \phi(k,h,w) c_k^{(i)} = a^{(i)} && \text{(fixed anchors)}\\ & c_{1,y}^{i} - c_{2,y}^{i} = 0, c_{3,y}^{i} - c_{4,y}^{i} = 0, && \text{(translation)}\\ & c_{1,x}^{i} - c_{3,x}^{i} = 0, c_{2,x}^{i} - c_{4,x}^{i} = 0 && \text{(translation)}\\ & \sum_{k=1}^{4} \phi(k,p_x^{(i)},p_y^{(i)})\ c_k^{(i)} - \sum_{k=1}^{4} \phi(k,p_x^{(j)},p_y^{(j)})\ c_k^{(j)} \approx 0 &&\\ & \forall (p^{(i)},p^{(j)}) \in I_{i,j} \quad . && \text{(reprojection)}\end{aligned} \tag{6}$$

We can further improve the efficiency for translation by directly using the initial transformation estimates instead of simply using them to get inliers. Here, each overlapping image pair contributes a pair of equations, rather than twice the number of inliers we choose to use. Hence the least squares problem is significantly reduced. While this results in a larger reprojection error, we find (Table II) the difference is not significant, and this more efficient method provides better ground truth accuracy.

For each overlapping image pair i and j, we form the following equations:

$$\begin{aligned}& c_x^i - c_x^j \approx \hat{T}_x^{(i,j)} && \text{(translation in x)}\\ & c_y^i - c_y^j \approx \hat{T}_y^{(i,j)} && \text{(translation in y)}\\ & c_k^{(i)} \approx \frac{1}{\sigma_{GPS}} \hat{c}_k^{(i)} \quad \forall i,k && \text{(noisy corners)}\\ & c_x^{(i)} = a_x^{(i)} - \alpha_x w && \text{(fixed anchors in x)}\\ & c_y^{(i)} = a_y^{(i)} - \alpha_y h \quad , && \text{(fixed anchors in y)}\end{aligned} \tag{7}$$

where c_x^i and c_y^i are the variables corresponding respectively to the x and y coordinates of the upper-left corner of the image i, $\hat{T}_x^{(i,j)}$ and $\hat{T}_y^{(i,j)}$ are the estimated x and y pairwise transformation between image i and j, and α_x, α_y are the ratio of the GPS field of view over the width and height of the images respectively. Using these linear least squares equations, we can solve for the upper-left corner of each image and calculate the other corners afterwards. This approach uses much less memory and CPU but is not linearly generalizable to other more complicated transformations such as similarity, affine and homography.

Lastly, for the case of projective transformations, instead of initializing the parameters of the bundle adjustment using naive approaches that are susceptible to drift, one can use the proposed method to solve for an affine approximation of the transformations and use it as the starting point of the non-linear bundle adjustment to speed up the convergence.

IV. Implementation

We implemented our proposed methods in Python (version 3.6) and we used the Scipy optimization library (version 1.4.1) to solve the least squares problems ("lsq_linear" for linear systems, and "least_squares" for non-linear systems, specifically for bundle adjustment and for our implementation of MGRAPH [18]). Both functions implement the Trust Region Reflective method [43]. For non-linear least squares, we derived the Jacobian matrix analytically for computational efficiency. We also used OpenCV (version 3.4.2) for extracting SIFT keypoints, finding matches, and estimating transformations. However, we implemented a RANSAC-based method for estimating translation parameters to enable additional optimizations.

We used the GPS coordinates associated with the drone images to find the nearest neighbors in order to reduce the number of pairwise transformations to be estimated. However, for the experiments on the gantry images, we also used them as priors on the coordinates of the corners as described in equations 6 and 7. Using these associated GPS coordinates, for each image we selected its k nearest neighbors using the following approach. For each pair of neighboring images, we extracted the SIFT keypoint locations and descriptors and computed putative matches using the two nearest neighbours in the descriptor space, dropping second neighbours whose score was not less than 80% of the first one as used by many others (e.g., [44], [45]). We used $k = 4$ and $k = 8$ for the drone and gantry experiments respectively. Using these pairs of keypoints, we estimated transformations of the appropriate type using RANSAC and saved them alongside with the inliers for subsequent use. Following Ruiz et al. [18], to reduce computation time and memory usage we used only the top 20 inliers to form the equations in all variations of our methods. Note that in equations 6 and 7 σ_{GPS} is used to properly incorporate the noisy estimates of the corners in the equations. The accuracy of these estimates is given by $\frac{1}{\sigma_{GPS}}$. We calculated σ_{GPS} by manually validating and revising (as needed) one of the gantry scans after being geo-corrected by our proposed method. More specifically, after the manual revision, the pairwise transformations are calculated using the accurate relative locations of the images. Comparing these results against the estimated transformations by RANSAC enabled measuring σ_{GPS}.

All of our experiments were performed on a system with Intel(R) Xeon(R) CPU X7560 and 882 GB of RAM.

V. Evaluation methodology

Datasets. We evaluated our proposed method on datasets of agricultural images captured by UAVs and the field gantry machine described above as well as two drone datasets from the DroneMapper website [17] (also used by Ruiz et al. [18]). The subject of the agricultural image datasets was a two acre research field crop where different plants were grown and monitored during different growing seasons throughout the year. Accurate orthomosaics are needed to closely monitor different phenotypes of individual crops growing over time. One set of agricultural images was captured by a DJI Phantom 4 v2 drone (quad-copter) which flies over the field and takes about 450 images on average from the field in each scan and it remains almost level during the flight. Another set of agricultural images was taken by cameras on board the gantry machine scanning the field with cameras near the ground to capture detailed features of the crops. The gantry system takes $6000-9000$ images on each scan of the field at a resolution of 0.3 mm per pixel. The resulting images have very low overlap ($\sim$ 9% vertically and $\sim$ 30% horizontally) and few distinct visual features which pose a challenge to the image stitching problem. The other two datasets are images of a reservoir (Gregg) and a golf course (Back 9 Golf Course) in Colorado which were captured by a drone. The Gregg and Golf Course datasets consist of 187 and 664 images respectively, and are available in the DroneMapper website.

Ground truth and evaluation measures. Given the scientific requirements of our setting, we needed to evaluate stitching with respect to ground truth locations. For this purpose, in the agricultural field we had ground control points (GCPs) installed on specific locations that were detected and identified in the images. For the Golf Course and Gregg datasets we found the imaged regions on Google Maps and manually selected a number of distinct locations as GCPs, noting the GPS locations provided by Google Maps. We evaluated all methods using four different measures: GCP root mean square error (RMSE), projection RMSE, normalized projection RMSE, and optimization time.

Computing GCP RMSE for the experiments with the gantry data is straight-forward as we include the rough estimates of the image corner GPS coordinates, as well as a single GCP anchor point in our equations (6,7), leading to orthomosaics in GPS units. A single anchor point suffices to correct for any global error assuming translation, but the evaluation does not depend on this.

For drone data we estimate the transformation between the reference image coordinate system and the GPS using all the GCPs to compute either a similarity or a homography as appropriate. We then transform the GCPs into the GPS space and calculate the root mean squared error between these transformed coordinates and the known locations of the GCPs. We use the following equation to calculate the distance in meters between two given points:

$$\begin{aligned} a &= sin^2(\frac{P_Y^1 - P_Y^2}{2}) + cos(P_Y^1) \times cos(P_Y^2) \times \\ &\qquad sin^2(\frac{P_X^1 - P_X^2}{2}) \qquad (8) \\ c &= 2 \times atan2(\sqrt(a), \sqrt(1-a)) \\ d &= 6371 \times 10^3 \times c \end{aligned}$$

where P^1 and P^2 are the two GPS coordinates with degree values converted to radians (X and Y are int the direction of longitude and latitude respectively) and 6371×10^3 is the earth radius. This is called the Haversine formula [46].

Projection RMSE, which measures how well points in overlapping areas align, is commonly used to evaluate stitching and alignment methods, and hence we report it. However, projection RMSE does not take into account the arbitrary deformations of the final mosaic, and so we also report normalized projection RMSE which is similarly computed after transforming to GPS coordinates as described above for computing the GCP RMSE measure.

VI. Results

We evaluated our method on the drone datasets using similarity as a solitary alignment method, as well as assuming that an affine transformation is a good initialization for the bundle adjustment. We compared these two variants to our implementation of MGRAPH [18]). Quantitative results are provided in Table I and qualitative results are shown in Figures 2, 4, and 5.

As we discussed briefly, the minor misalignment errors in pairwise matching of images may accumulate and cause drift. As a result of that, two different sequences of chaining pairwise transformations may not result in the same absolute transformation as we would expect. Moreover, the non-linear optimization used in methods like MGRAPH are prone to local minimum trappings that may cause further global misalignment, and unnatural deformations and warps. As illustrated in the mosaics in Figure 2, the drift caused by a bad initialization in MGRAPH unnaturally warps the mosaic. In this case, although the keypoints might be nicely aligned as indicated by the projection RMSE in Table I, the ground truth GPS location is inaccurate as both the qualitative and quantitative results suggest. This example supports our focus on GCP RMSE, and normalized projection RMS as a better proxy.

For the two drone datasets of the agricultural field, MegaStitch with similarity yields the best results. For the Gregg II and Golf Course datasets homography is called for. For Gregg II, MegaStitch with affine as initialization for the bundle adjustment produces the best results. By contrast, the MGRAPH initialization is not good, and the optimization process rapidly finds a poor local minimum, explaining the very fast optimization

Measure	Methods	Datasets (Drone)			
		Ag. Field Lettuce	Ag. Field Sorghum	Golf Course	Gregg II
GCP RMSE (meters)	Pix4DMapper	0.25	1.39	**2.32**	**0.25**
	MegaStitch Similarity	**0.15**	**0.49**	29.96	10.77
	MegaStitch Affine + Bndl. Adj.	1.63	10.31	9.33	0.75
	MGRAPH	8.75	6.83	6.93	24.02
Projection RMSE (pixels)	MegaStitch Similarity	0.69	**0.68**	**1.15**	1.96
	MegaStitch Affine + Bndl. Adj.	**0.57**	4.35	1.40	**1.37**
	MGRAPH	0.72	0.85	1.69	8145.37
Normalized Projection RMSE (meters)	MegaStitch Similarity	**0.02**	**0.02**	**0.63**	0.60
	MegaStitch Affine + Bndl. Adj.	0.02	0.36	0.77	**0.46**
	MGRAPH	0.09	0.04	1.00	6831.92
Optimization Time	MegaStitch Similarity	**2m 24s**	**2m 59s**	**5m 23s**	**9s**
	MegaStitch Affine + Bndl. Adj.	$13m$ $37s$	$4m$ $1s$	$14m$ $50s$	$1m$ $16s$
	MGRAPH	$10m$ $36s$	$4m$ $47s$	$37m$ $39s$	$16s$

TABLE I: Results of our proposed method and MGRAPH on four drone datasets. Qualitative results are shown in figures 2,4, and 5. GCP RMSE and normalized projection RMSE are distances in the GPS coordinate system (meters) and projection RMSE are distances in the reference image coordinate system (pixels). On the agricultural field drone datasets MegaStitch with similarity produces the best GCP RMSE, normalized projection error and optimization time. For the Golf Course dataset, MegaStitch with affine and bundle adjustment produces comparable results to MGRAPH (GCP RMSE is slightly worse and projection error is slightly better), but three times faster. For the Gregg II dataset, MGRAPH quickly goes to a poor local minimum, with accuracy being orders of magnitude worse than MegaStitch requiring pruning outliers to show part of the mosaic in Figure 5. Also included in the table is a result using the closed-source commerical software, Pix4DMapper, which cannot handle the gantry data, but does well on the drone datasets. Because Pix4DMapper only runs on Windows platforms, we are not able to provide meaningfull run time results.

Methods	Performance on the Gantry datasets		
	GCP RMSE	Projection RMSE	Optimization Time
Keypoint-based	0.20	0.01	$5h$ $23m$ $48s$
RANSAC Transformations	0.18	0.01	$13m$ $42s$

TABLE II: Results for the two methods for gantry data assuming translation. MegaStitch expressed in terms of translation parameters estimated by RANSAC (7) yields a better GCP RMSE result than raw keypoints (6) in a considerably shorter period of time. Note that the GCP and Projection RMSE scores are distances in meters.

time. Further inspection reveals that the bulk of the mosaic, which includes all GCPs, is visually reasonable (Figure 5), but to make this figure we had to remove outliers. Specifically, we removed the top and bottom 5% of mapped image sizes. These outliers partly explain the quantitative results being several orders of magnitude worse than those for MegaStitch.

For the Golf Course images, MGRAPH does not run into these issues, and does slightly better then MegaStitch on GCP RMSE, and slightly worse on projection RMSE, which we attribute to projection RMSE being closer to what MegaStitch actually optimizes. However, MGRAPH runs three times slower compared to MegaStitch which undermines its gain in GCP RMSE.

We also evaluated our method on the gantry images. Since the gantry has only two perpendicular axes of motion, images are connected by translations. We incorporated the initial noisy locations of each image in the equations. One of the GCPs was also included in the equations as an anchor point. We evaluated methods corresponding to equations 6 and 7 on the gantry images. The quantitative results are presented in Table II and the qualitative results are illustrated in Figure 3. Note that the illumination differences and shadows did not have any adverse effect on the ability of our proposed method to align and geo-correct the images as SIFT is somewhat invariant to illumination differences. However, shadow artifacts are present in our final images. However, since the geo-corrected images are used for downstream scientific analysis (such as measuring greenness indices), we did not apply pixel blending or seamless stitching. These processing steps could be added to our method as needed by other applications.

We find that optimizing with the translation parameters (7) is substantially faster than the raw keypoint based projection minimization (6). This second approach also consumes 10 times more memory since it takes into account a large subset of keypoint matches rather than the robustly estimated translation parameters. While this

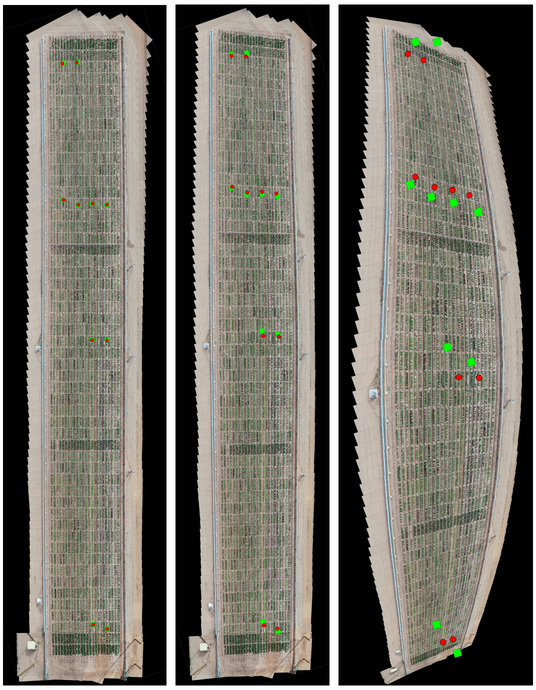


Fig. 2: Mosaics generated by MegaStitch and MGRAPH for the lettuce drone dataset. The sorghum dataset yields similar results. Red circles are the GCP locations, and green squares are their estimated locations. MegaStitch with similarity (left) is the best at estimating GCP location. On the other hand, the MGRAPH mosaic (right) exhibits drift which is not repaired by the optimization, and the final result has unwanted warp and global inconsistency. By contrast, bundle adjustment assuming homography from an affine initialization (center), does not have these issues on this data.

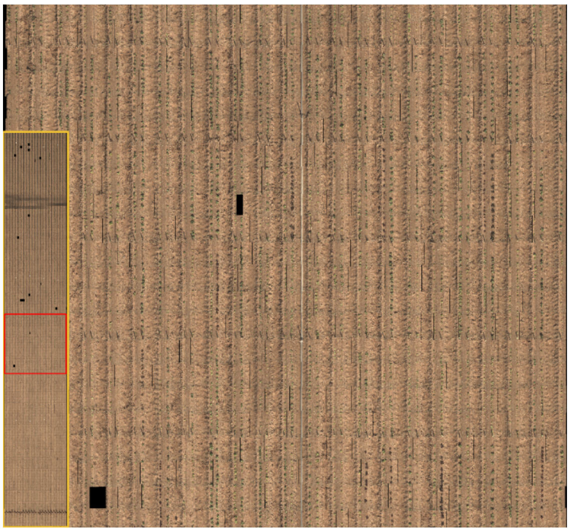


Fig. 3: Mosaic generated by MegaStitch on the gantry dataset based on the intermediate translations found using RANSAC (7). The keypoint-based method yields visually indistinguishable results, but uses about 10 times the resources. The scale of this data set ($\approx 10,000$ images) defeated multiple alternative methods, motivating this work.

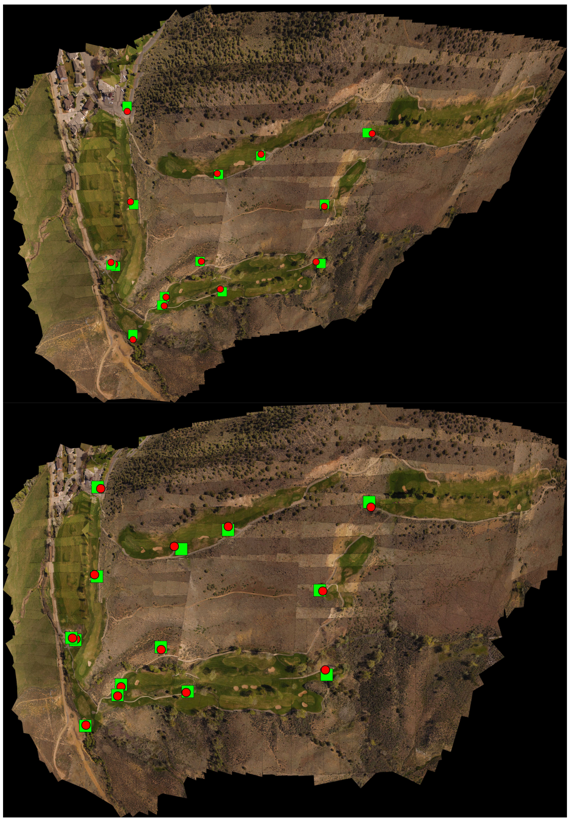


Fig. 4: Mosaics generated by MegaStitch using affine followed by bundle adjustment (top) and the MGRAPH (bottom) on the drone images of the Golf Course dataset. Red circles are the GCP locations and green squares are their estimated locations. The two results are similar. As reported in Table I, MGRAPH does a bit better on GCP RSME, and MegaStitch does a bit better on projection RMSE.

is the case for all the general bundle adjustment methods, translation affords the alternative approach. The accuracies of the two methods are similar, with the second (faster) approach being slightly better on GCP RMSE, and slightly worse on projection RMSE, likely because projection RMSE is closer to what the raw keypoint method is optimizing.

VII. Conclusion

We contribute methods for large scale image alignment for scientific monitoring applications where accuracy with respect to ground truth is critical. Our approach is more robust and significantly faster than alternatives. We use inliers from pairwise transformations directly for global least squares solutions, as for many applications non-projective transformations suffice. Moreover, we found that non-projective alignment using non-linear optimization is sensitive to initialization, and that the globally valid approximate solution from MegaStitch can efficiently provide a good initialization. We also developed a large-scale ground truth dataset for this task which is available along with our code at https://github.com/ariyanzri/MegaStitch.

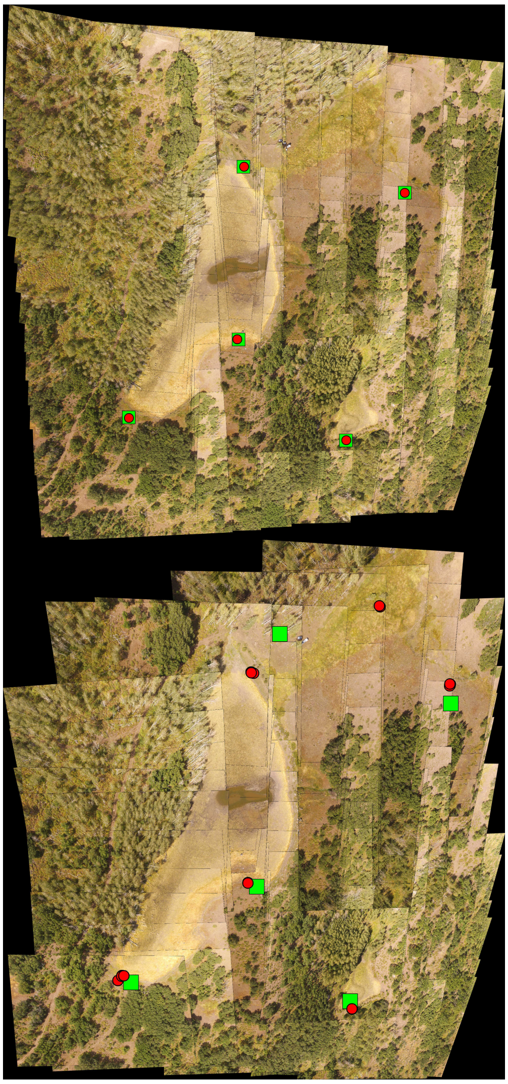


Fig. 5: Mosaics generated by MegaStitch using affine followed by bundle adjustment (top) and MGRAPH (bottom) on the Gregg II drone dataset. Red circles are the GCP locations and green squares are their estimated locations. For the MGRAPH mosaic we removed a few images since the optimization falls into a such a bad local minimum that the inlier results are hard to inspect otherwise. MegaStitch provides a good initialization for the bundle adjustment, and does much better overall.

Acknowledgments

Funding for this research was provided by Department of Energy Advanced Research Projects Agency – Energy (award numbers: DE-AR0000594 and DE-AR0001101), Department of Energy Biological and Environmental Research (award number: DE-SC0020401), The Bill and Melinda Gates Foundation (award number: OPP1129603), and Cotton Incorporated (award number: 20-720).

References

[1] M. Louargant, G. Jones, R. Faroux, J.-N. Paoli, T. Maillot, C. Gée, and S. Villette, "Unsupervised classification algorithm for early weed detection in row-crops by combining spatial and spectral information," *Remote Sensing*, vol. 10, no. 5, p. 761, 2018. 1

[2] A. Bauer, A. G. Bostrom, J. Ball, C. Applegate, T. Cheng, S. Laycock, S. M. Rojas, J. Kirwan, and J. Zhou, "Combining computer vision and deep learning to enable ultra-scale aerial phenotyping and precision agriculture: A case study of lettuce production," *Horticulture research*, vol. 6, no. 1, pp. 1–12, 2019. 1

[3] J. Dong, J. G. Burnham, B. Boots, G. Rains, and F. Dellaert, "4d crop monitoring: Spatio-temporal reconstruction for agriculture," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3878–3885, IEEE, 2017. 1

[4] A. Patrignani and T. E. Ochsner, "Canopeo: A powerful new tool for measuring fractional green canopy cover," *Agronomy Journal*, vol. 107, no. 6, pp. 2312–2320, 2015. 1

[5] H. AliAkbarpour, K. Gao, R. Aktar, S. Suddarth, and K. Palaniappan, "Structure from motion and mosaicking for high-throughput field-scale phenotyping," in *High-Throughput Crop Phenotyping*, pp. 55–69, Springer, 2021. 1

[6] R. Aktar, D. E. Kharismawati, K. Palaniappan, H. Aliakbarpour, F. Bunyak, A. E. Stapleton, and T. Kazic, "Robust mosaicking of maize fields from aerial imagery," *Applications in plant sciences*, vol. 8, no. 8, p. e11387, 2020. 1

[7] R. Viguier, C. C. Lin, H. AliAkbarpour, F. Bunyak, S. Pankanti, G. Seetharaman, and K. Palaniappan, "Automatic video content summarization using geospatial mosaics of aerial imagery," in *2015 IEEE International Symposium on Multimedia (ISM)*, pp. 249–253, IEEE, 2015. 1

[8] R. Aktar, V. S. Prasath, H. Aliakbarpour, U. Sampathkumar, G. Seetharaman, and K. Palaniappan, "Video haze removal and poisson blending based mini-mosaics for wide area motion imagery," in *2016 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 1–7, IEEE, 2016. 1

[9] R. Aktar, H. AliAkbarpour, F. Bunyak, T. Kazic, G. Seetharaman, and K. Palaniappan, "Geospatial content summarization of uav aerial imagery using mosaicking," in *Geospatial Informatics, Motion Imagery, and Network Analytics VIII*, vol. 10645, p. 106450I, International Society for Optics and Photonics, 2018. 1

[10] R. Aktar, V. H. Huxley, G. Guidoboni, H. AliAkbarpour, F. Bunyak, and K. Palaniappan, "Mosaicing of dynamic mesentery video with gradient blending," in *2020 IEEE International Conference on Image Processing (ICIP)*, pp. 563–567, IEEE, 2020. 1

[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004. 1, 2

[12] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the ACM*, vol. 24, pp. 381–395, 1981. 1, 2

[13] H. Aliakbarpour, K. Palaniappan, and G. Seetharaman, "Robust camera pose refinement and rapid sfm for multiview aerial imagery—without ransac," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 11, pp. 2203–2207, 2015. 1, 2

[14] H. Aliakbarpour, K. Palaniappan, and G. Seetharaman, "Stabilization of airborne video using sensor exterior orientation with analytical homography modeling," in *Machine Vision and Navigation*, pp. 579–595, Springer, 2020. 1, 2

[15] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment – a modern synthesis," in *Vision Algorithms: Theory and Practice* (B. Triggs, A. Zisserman, and R. Szeliski, eds.), pp. 298–372, Springer-Verlag, 2000. 2

[16] P. F. McLauchlan and A. Jaenicke, "Image mosaicing using sequential bundle adjustment," *Image and Vision computing*, vol. 20, no. 9-10, pp. 751–759, 2002. 2

[17] "Dronemapper." https://dronemapper.com/sample_data/. Accessed: 2021-03-10. 2, 5

[18] J. J. Ruiz, F. Caballero, and L. Merino, "Mgraph: A multigraph homography method to generate incremental mosaics in real-time from uav swarms," *Ieee Robotics and Automation Letters*, vol. 3, no. 4, pp. 2838–2845, 2018. 2, 3, 4, 5

[19] R. Szeliski, *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010. 2

[20] R. Szelisk, "Image alignment and stitching," in *Handbook of Mathematical Models in Computer Vision* (P. N, C. Y, and F. O, eds.), Boston, MA: Springer, 2006. 2, 3

[21] M. Brown and D. Lowe, "Recognising panoramas," in *IEEE International Conference on Computer Vision*, pp. 1218–1225, 2003. 2

[22] M. Brown and D. Lowe, "Automatic panoramic image stitching using invariant features," *International Journal of Computer Vision*, vol. 74, no. 1, pp. 59–73, 2007. 2

[23] F. Zhao, Q. Huang, and W. Gao, "Image matching by normalized cross-correlation," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 2, pp. II–II, IEEE, 2006. 2

[24] C. de Cesare, M.-J. Rendas, A.-G. Allais, and M. Perrier, "Low overlap image registration based on both entropy and mutual information measures," in *OCEANS 2008*, pp. 1–9, IEEE, 2008. 2

[25] E. Zagrouba, W. Barhoumi, and S. Amri, "An efficient image-mosaicing method based on multifeature matching," *Machine Vision and Applications*, vol. 20, no. 3, pp. 139–162, 2009. 2

[26] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International journal of computer vision*, vol. 60, no. 1, pp. 63–86, 2004. 2

[27] N. Geng, D. He, and Y. Song, "Camera image mosaicing based on an optimized surf algorithm," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 10, no. 8, pp. 2183–2193, 2012. 2

[28] J. Xingteng, W. Xuan, and D. Zhe, "Image matching method based on improved surf algorithm," in *2015 IEEE International Conference on Computer and Communications (ICCC)*, pp. 142–145, IEEE, 2015. 2

[29] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*, pp. 404–417, Springer, 2006. 2

[30] H. Xie, N. Hicks, G. R. Keller, H. Huang, and V. Kreinovich, "An idl/envi implementation of the fft-based algorithm for automatic image registration," *Computers & Geosciences*, vol. 29, no. 8, pp. 1045–1055, 2003. 2

[31] S. Preibisch, S. Saalfeld, and P. Tomancak, "Globally optimal stitching of tiled 3d microscopic image acquisitions," *Bioinformatics*, vol. 25, no. 11, p. 1463–1465, 2009. 2

[32] M. Mizotin, G. Krivovyaz, A. Velizhev, A. Chernyavskiy, and A. Sechin, "Robust matching of aerial images with low overlap," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, pp. 13–18, 2010. 2

[33] R. Xiang, M. Sun, C. Jiang, L. Liu, H. Zheng, and X. Li, "A method of fast mosaic for massive uav images," in *Land Surface Remote Sensing II*, vol. 9260, p. 92603W, International Society for Optics and Photonics, 2014. 2

[34] A. Moussa and N. El-Sheimy, "A fast approach for stitching of aerial images.," *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. 41, 2016. 2

[35] L. P. Chew, "Constrained delaunay triangulations," *Algorithmica*, vol. 4, no. 1-4, pp. 97–108, 1989. 2

[36] J. Zhao, X. Zhang, C. Gao, X. Qiu, Y. Tian, Y. Zhu, and W. Cao, "Rapid mosaicking of unmanned aerial vehicle (uav) images for crop growth monitoring using the sift algorithm," *Remote Sensing*, vol. 11, no. 10, p. 1226, 2019. 3

[37] D. Arteaga Meléndez, G. Kemper Vásquez, S. G. Huaman Bustamante, J. Telles Castillo, L. Bendayán Acosta, and J. Sanjurjo Vílchez, "A method for mosaicking aerial images based on flight trajectory and the calculation of symmetric transfer error per inlier," 2019. 3

[38] E. Petro, C. Martinez, D. Patino, M. Rebolledo, J. Colorado, *et al.*, "Aerial mapping of rice crops using mosaicing techniques for vegetative index monitoring," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 846–855, IEEE, 2018. 3

[39] J. Rojas, C. Martinez, I. Mondragon, and J. Colorado, "Towards image mosaicking with aerial images for monitoring rice crops," in *Advances in Automation and Robotics Research in Latin America*, pp. 279–296, Springer, 2017. 3

[40] T. Botterill, S. Mills, and R. Green, "Real-time aerial image mosaicing," in *International Conference of Image and Vision Computing*, 2010. 3

[41] C. Xing, J. Wang, and Y. Xu, "A method for building a mosaic with uav images," *IJIEEB*, vol. 2, pp. 9–15, 2010. 3

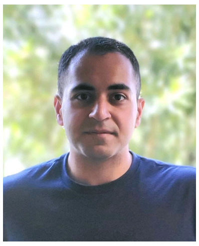
[42] J. Liu, J. Gong, B. Guo, and W. Zhang, "A novel adjustment model for mosaicking low-overlap sweeping images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 4089–4097, 2017. 3

[43] D. C. Sorensen, "Newton's method with a model trust region modification," *SIAM Journal on Numerical Analysis*, vol. 19, no. 2, pp. 409–426, 1982. 4

[44] K. Gao, H. Aliakbarpour, J. Fraser, K. Nouduri, F. Bunyak, R. Massaro, G. Seetharaman, and K. Palaniappan, "Local feature performance evaluation for structure-from-motion and multi-view stereo using simulated city-scale aerial imagery," *IEEE Sensors Journal*, vol. 21, no. 10, pp. 11615–11627, 2020. 4

[45] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004. 4

[46] H. Goodwin, "The haversine in nautical astronomy," in *US Naval Institute Proceedings*, vol. 36, pp. 735–746, 1910. 5



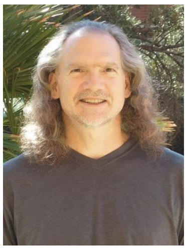
Ariyan Zarei received his B.Sc. degree in Computer Science from Shahid Beheshti University, Tehran, Iran in 2018 and his M.Sc. degree from the University of Arizona, Tucson in 2020. He is currently pursuing his PhD in Computer Science at the University of Arizona. His research focuses on the applications of Computer Vision and Machine Learning in Plant Sciences and Remote Sensing.

Emmanuel Gonzalez earned a Bachelor of Science in Biology with a minor in Chemistry from Pacific Lutheran University. He is now a graduate student in Dr. Duke Pauli's lab at the University of Arizona. His research involves leveraging sensor technology, high-performance computing, and machine learning to extract phenotypic trait data at scale. These data are used to (1) identify genetic factors contributing to stress-adaptive traits and (2) develop predictive models for key agronomic traits.

Nirav Merchant Nirav Merchant is the Co-PI for NSF CyVerse(link is external) a national scale Cyberinfrastructure for life sciences and (link is external)NSF Jetstream(link is external) the first user-friendly, scalable cloud environment for NSF XSEDE. He received his undergraduate degree in Industrial engineering from the University of Pune, India, and graduate degree in Systems and Industrial Engineering from the University of Arizona (1994). Over the last two decades his research has been directed towards developing scalable computational platforms for supporting open science and open innovation, with emphasis on improving research productivity for geographically distributed interdisciplinary teams. His interests include data science literacy, large-scale data management platforms, data delivery technologies, managed sensor and mobile platforms for health interventions, workforce development, and project based learning.

Dr. Duke Pauli is an Assistant Professor in the School of Plant Sciences at the University of Arizona where his research focuses on elucidating the genetic basis of stress-adaptive traits in crop plants.His specific area of expertise is leveraging phenomic technologies in junction with quantitative genetics to identify, characterize, and quantify the effects of these loci in mitigating the impact of abiotic stress conditions on plant performance. His skills and experience span a diverse set of crops, over ten, including growing and evaluating lettuce underneath the UA Field Scanner, the world's largest outdoor phenotyping robot.

Dr. Eric Lyons is an associate professor in the school of Plant Sciences at the University of Arizona. His research focuses on scalable computational systems and infrastructure to support and accelerate life science research. To support this, Dr. Lyons is a co-PI on CyVerse, a 115M $ project funded by the National Science Foundation to provide cyberinfrastructure for life science research. In addition, he develops and maintains the comparative genomics platform, CoGe (http://genomevolution.org), which currently stores over 50,000 genomes from all domains of life. He has authored over 100 peer reviewed articles and book chapters, and teaches students how to use large-scale computing to solve problems and answer questions in biology. Dr. Lyons serves on several boards of non-profit companies and research institutions and has worked in biotech, pharma, and software companies around the SF Bay Area, and has served as a Program Director at the National Science Foundation in the Plant Genome Research Program.



Dr. Kobus Barnard is a professor of computer science at the University of Arizona, and has appointments with electrical and computer engineering (ECE), cognitive science, statistics, and applied math. Before coming to Arizona, he was a post doctoral fellow in computer vision at the University of California at Berkeley. He did his Ph.D. in computer science at Simon Fraser University, specializing in colour constancy. His research interests include building and learning explanatory probabilistic models for computer vision and interdisciplinary applications.