MULTI-LEAF TRACKING FROM FLUORESCENCE PLANT VIDEOS*

Xi Yin, Xiaoming Liu†*, Jin Chen, David M. Kramer*

Michigan State University, East Lansing, MI, USA

ABSTRACT

Driven by the plant phenotyping application, this paper proposes a new leaf tracking framework to jointly segment, align and track multiple leaves from fluorescence plant videos. Our framework consists of two steps. First, leaf alignment is applied to one video frame to generate a collection of leaf candidates. Second, we define a set of transformation parameters operated on the leaf candidates in order to optimize the alignment in the subsequent video frame according to an objective function. Gradient descent is employed to solve this optimization problem. Experimental results show that the proposed multi-leaf tracking algorithm is superior to the image-based leaf alignment method in terms of three quantitative metrics.

Index Terms— Leaf tracking, alignment, multi-leaf

1. INTRODUCTION

Plant phenotyping concerns the growth of plants – a fundamental biological process studied in a range of scientific fields [1]. Recently computational approaches to plant growth modeling have emerged, owing to not only the prospects of quantitative analysis [2], but also the advance of sensing technology. One key problem in plant growth modeling is to understand the *photosynthetic efficiency* (PE) of different plants under external stimuli or genre mutations.

To study the PE, as shown in Fig. 1, a fluorescence camera captures a fluorescence video of the growing plants [3, 4] in a growth chamber, where the pixel intensity of a video frame represents the PE at a particular leaf spot and time. This video data holds the answer to many interesting biological questions, such as which leaf and which part of a leaf has higher PE, and how does the PE of each leaf vary over time. To discover this leaf-level knowledge, it is imperative to *track* all leaves over time and accurately estimate the *structure* of each leaf, i.e., leaf tips and skeleton. This computer vision problem, termed *multi-leaf tracking*, is at the heart of our paper.

Our multi-leaf tracking problem is uniquely defined to perform *leaf segmentation*, *alignment* and *tracking* for each fluorescence video. Specifically, leaf segmentation identifies the contour of each leaf and thus the total number of leaves [5]. Leaf alignment estimates the leaf structure by aligning with a leaf template [4]. Leaf tracking tracks the locations and/or contour of leaves over time [6]. In contrast to prior work focusing on one or two of these tasks, this paper addresses all three tasks simultaneously. Clearly, our unique problem is challenging due to various factors, such as the low-resolution fluorescence images, leaf overlapping, and the varying leaf shape and appearance over time.

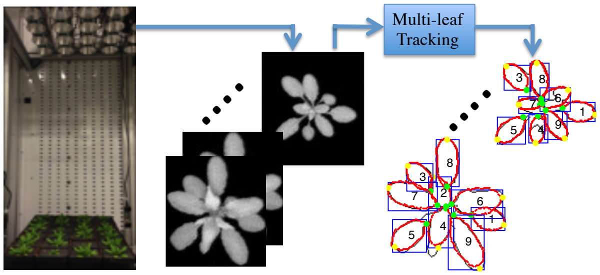

Fig. 1. Computational plant growth modeling. We capture a fluorescence video of the plants during their growth period, and employ a novel optimization method to multi-leaf tracking, i.e., estimating unique and consistent-over-time labels for all leaves and their individual leaf structure such as the leaf tips.

To tackle this unique problem, we propose to leverage our leaf alignment work [4], which jointly estimates the number of leaves and their respective structures from a fluorescence image. While performing well, it has two main drawbacks. First, being an *image*-based approach, it performs leaf alignment for each video frame independently. Hence, leaf correspondence between consecutive frames can be an issue due to potentially different leaf segmentation at different frames, especially for overlapping leaves. Second, independent alignment for each frame does not take advantage of the temporal coherence in videos and implies extra computational burden.

To remedy these problems, we develop a method to track multiple leaves simultaneously over time. Assuming leaf alignment is conducted on one video frame via [4], we define a three-term objective function whose optimization leads to the aligned leaves in the subsequent frame. The first term is the average Chamfer matching distances of all transformed leaf candidates with the test image edge. The second is the difference between the summation of all transformed leaf

*THE FLUORESCENCE MEASUREMENTS AND SOFTWARE DEVELOPMENT WERE SUPPORTED IN PART BY THE CHEMICAL SCIENCES, GEOSCIENCES, AND BIOSCIENCES, OFFICE OF BASIC ENERGY SCIENCES, OFFICE OF SCIENCE, US DEPARTMENT OF ENERGY (AWARD NUMBER DE-FG02-91ER20021). THE INSTRUMENTATION WAS SUPPORTED BY THE MSU CENTER FOR ADVANCED ALGAL AND PLANT PHENOTYPING.

† CORRESPONDING AUTHOR: XIAOMING LIU, LIUXM@MSU.EDU.

978-1-4799-5751-4/14/$31.00 ©2014 IEEE

candidates and the test image mask. The third encourages the long axis of each leaf pointing toward the center of the plant. The optimization process estimates the transformation parameters of all leaves simultaneously by minimizing the objective function. Due to slow plant growth between consecutive frames, the optimization initialized with the results of previous frame has a fast convergence. Using a dataset of fluorescence videos with manual labels on the leaf structure, the quantitative evaluation shows the effectiveness of our tracking approach. In summary, the main *contributions* of this work lie in our optimization method to simultaneous multi-leaf segmentation, alignment and tracking, as well as the quantitative evaluation scheme for this complex problem.

Related Work Leaves and plants are common objects of interest in both computer graphics and vision fields, with existing efforts on tasks such as image-based modeling [7, 8], leaf segmentation [5, 9], alignment [4, 10], tracking [6], and retrieval and identification [11–13]. Prior works focus on one or two of these tasks. In contrast, this paper addresses the tasks of multi-leaf *segmentation*, *alignment* and *tracking* simultaneously from a plant video.

Most prior image alignment work, such as ASM [14], AAM [15], Chamfer matching [16] and its extensions [17–21], concentrate on aligning a single object instance. Our work extends Chamfer matching in the direction of aligning and tracking multiple potentially overlapping objects in one image. Also, unlike [15], we uniquely employ both forward and backward warping in one objective function.

2. MULTI-LEAF TRACKING APPROACH

Given a fluorescence plant video, the proposed approach is composed of two steps. First, multi-leaf alignment [4] is applied to one frame to estimate the location and shape of each leaf candidate. Second, we define a three-term objective function on the transformation parameters of those leaf candidates so as to optimize their alignment in a subsequent frame. For the wholeness of this paper, we first briefly introduce our leaf alignment work [4], and then present our tracking algorithm.

2.1. Multi-leaf Alignment

Our multi-leaf alignment [4] extends the popular Chamfer matching (CM) method that computes the distance between the edge points of an object template, $\boldsymbol{U} = \{\boldsymbol{u}^j\}_{j=1}^{k_u}$, and those of a test image $\boldsymbol{I}$, $\boldsymbol{V} = \{\boldsymbol{v}^j\}_{j=1}^{k_v}$ [16]. It can be computed efficiently via a distance transform image $DT(\boldsymbol{g}) = \min_{j\in[1,k_v]} \|\boldsymbol{g}-\boldsymbol{v}^j\|_2$, which stores the distance of each coordinate $\boldsymbol{g}$ in $\boldsymbol{I}$ to its closest edge point. Overlaying a template $\boldsymbol{U}$ on DT with a shift $\boldsymbol{t} = [t_x, t_y]^\intercal$ results in the CM distance,

$$d(\boldsymbol{U}+\boldsymbol{t}, \boldsymbol{V}) = \frac{1}{k_u}\sum_{j=1}^{k_u} DT(\boldsymbol{u}^j + \boldsymbol{t}). \tag{1}$$

We first generate a set of leaf templates with different scales and orientations from several basic templates with distinct shapes. For each template, it overlays on the test image with all possible shifts and the minimum CM distance leads to the optimal shift. This operation on all templates forms an over-completed pool of transformed leaf templates, named *leaf candidates*. Second, we select the best subset of leaf candidates by optimizing an objective function. The rationality behind our objective function is to select a minimal number of leaf templates with smaller CM distances to jointly best cover the test image mask. The function is parameterized by an indicator vector $\boldsymbol{x}$ whose length equals to the number of leaf candidates, and whose value indicates whether the corresponding candidate is selected. Gradient descent-based minimization is used to estimate $\boldsymbol{x}$, i.e., selecting leaf candidates.

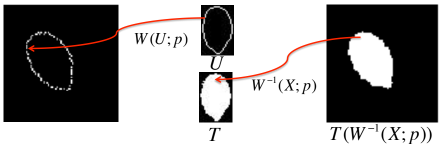

Fig. 2. Forward and backward warping.

For each of A selected leaf candidates, we denote its orientation as θ, scale as r, optimal shift as (t_x, t_y). We further denote $\boldsymbol{p} = [\theta, r, t_x, t_y]^\intercal$ as the transformation parameter that uniquely describes how a basic leaf template can be transformed into a leaf candidate. The edge points of the original basic template are denoted as $\boldsymbol{U}$ and its mask as $\boldsymbol{T}$. The collection of all A selected leaf candidates is denoted as $\mathbb{M} = \{\boldsymbol{U}_i, \boldsymbol{T}_i, \boldsymbol{p}_i\}_{i=1}^{A}$, which will be used for leaf tracking in subsequent video frames.

Our leaf tracking assumes the temporal coherence, i.e., the leaves change gradually between consecutive frames. Hence, it is possible to *transform* the leaf candidates in $\mathbb{M}$ at the current frame so that it better *aligns* with the next frame. We present three elements in our tracking algorithm: template transformation, objective function, and optimization.

2.2. Template Transformation

Let $\boldsymbol{W} : \boldsymbol{U} \mapsto \boldsymbol{U}'$ be a warping function that transfers a set of 2D coordinates in the template space to those in the test image space. This function, termed *forward warping*, is parameterized by the transformation parameter $\boldsymbol{p}$,

$$\boldsymbol{u}' = r\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}(\boldsymbol{u}-\bar{\boldsymbol{U}}) + \boldsymbol{t} + \bar{\boldsymbol{U}}, \tag{2}$$

where $\bar{\boldsymbol{U}}$ is the average of all coordinates in $\boldsymbol{U}$. Including $\bar{\boldsymbol{U}}$ allows us to model the leaf scaling and rotation w.r.t. the individual leaf center – a typical phenomenon in plant videos. For a leaf candidate from $\mathbb{M}$, $d(\boldsymbol{W}(\boldsymbol{U}_i; \boldsymbol{p}_i), \boldsymbol{V})$ represents the CM distance of the transformed template $\boldsymbol{W}(\boldsymbol{U}_i; \boldsymbol{p}_i)$ to the edge map $\boldsymbol{V}$ of the test image $\boldsymbol{I}$.

We also consider the *backward warping* $\boldsymbol{W}^{-1} : \boldsymbol{U}' \mapsto \boldsymbol{U}$ from the test image space to the template space, as shown in Fig. 2. Assuming $\boldsymbol{X}$ is a $K \times 2$ matrix including all K coordinates of $\boldsymbol{I}$, $\boldsymbol{W}^{-1}(\boldsymbol{X}; \boldsymbol{p})$ are the corresponding coordinates of $\boldsymbol{X}$ in the template space. The purpose of this backward warping is to generate a K-dim vector $\boldsymbol{T}_i(\boldsymbol{W}^{-1}(\boldsymbol{X}; \boldsymbol{p}_i))$ representing the warped image of the template mask $\boldsymbol{T}_i$.

2.3. Objective Function

We denote the set of transformation parameters as $\boldsymbol{P} = \{\boldsymbol{p}_i\}_{i=1}^{A}$. Since $\boldsymbol{P}$ specify leaf candidates matching the current frame, the leaf tracking problem, i.e., finding the candidates for the subsequent frames, is equivalent to updating $\boldsymbol{P}$ according to certain objectives. Hence, we formulate a three-term objective function that is parameterized by $\boldsymbol{P}$. We now describe each objective and its mathematical formulation.

The first objective is to update $\boldsymbol{P}$ so that the edge points of the transformed candidates are well aligned with those of the test frame. This is achieved by minimizing the average CM distances between these two sets of edge points,

$$J_1 = \frac{1}{A}\sum_{i=1}^{A} d(\boldsymbol{W}(\boldsymbol{U}_i;\boldsymbol{p}_i), \boldsymbol{V}). \tag{3}$$

The second objective is to encourage the synthesized mask from all transformed candidates to be similar to the test frame mask $\boldsymbol{m}$, a K-dim binary vector, i.e.,

$$J_2 = \frac{1}{K}\|\sum_{i=1}^{A}\boldsymbol{T}_i(\boldsymbol{W}^{-1}(\boldsymbol{X};\boldsymbol{p}_i)) - \boldsymbol{m}\|_2. \tag{4}$$

Due to the round shape of certain leaves, the in-plane rotation θ could be any angle without an additional constraint, which may cause a large error in leaf tip estimation. Hence, the third objective is to enforce that the rotation angle of each leaf candidate be similar to that of the vector from the plant center to the leaf candidate center, i.e., all leaf should point toward the plant center. We formulate the third objective as,

$$J_3 = \frac{1}{A}\sum_{i=1}^{A}\|\frac{c_i^y - c_y}{\sqrt{(c_i^x - c_x)^2 + (c_i^y - c_y)^2}} - \sin(\theta_i)\|_2, \tag{5}$$

where (c_i^x, c_i^y) is the center of a leaf candidate, i.e., the average of $\boldsymbol{W}(\boldsymbol{U}_i;\boldsymbol{p}_i)$, and (c_x, c_y) is the geometric center of a plant, i.e., the average coordinates of all foreground pixels in $\boldsymbol{m}$. Finally, we define the objective function as,

$$J(\boldsymbol{P}) = J_1 + \lambda_1 J_2 + \lambda_2 J_3, \tag{6}$$

where λ_1 and λ_2 are the weighting parameters.

2.4. Gradient-based Optimization

Given the objective function in Eqn. 6, our goal is to minimize it by estimating $\boldsymbol{P}$, i.e., $\boldsymbol{P} = \arg\min_{\boldsymbol{P}} J(\boldsymbol{P})$. Since $J(\boldsymbol{P})$ involves texture warping, it is an nonlinear optimization problem without close-form solutions. We use gradient descent to iteratively solve this problem.

By assuming the temporal coherence, $\boldsymbol{P}$ is initialized as the transformation parameters of the previous frame $\boldsymbol{P}^0 = \{\boldsymbol{p}_i^0\}_{i=1}^{A}$. During the tracking, $\boldsymbol{P}$ is iteratively updated by $\boldsymbol{p}_i^t = \boldsymbol{p}_i^{t-1} - \alpha\frac{dJ}{d\boldsymbol{p}_i}$ for the i^{th} leaf candidate at iteration t, where α is the step size and $\frac{dJ}{d\boldsymbol{p}_i}$ is the gradient of J w.r.t. $\boldsymbol{p}_i$. The iteration stops when there is little change in $\boldsymbol{P}$ or it reaches the maximum iteration number (80 in our system). Although the updating is applied to each leaf candidate *independently*, the computation of $\frac{dJ_2}{d\boldsymbol{p}_i}$ involves all A candidates and therefore this is a multi-leaf *joint* optimization problem.

Algorithm 1: Performance evaluation process.

```
Input: Tracking results $\{\boldsymbol{M}_n^e, \boldsymbol{S}_n^e\}$ and label $\{\boldsymbol{M}_n^l, \boldsymbol{S}_n^l\}$.
Output: $F$, $D$, $E$, and $T$.
1. Initialize $f = d = 0$, and $\boldsymbol{e} = [\,]$.
2. for $n = 1, \ldots, \#plant$ do
    Perform index switch according to Eqn. 5 in [4].
    for $j = 1, \ldots, \#image$ do
        for $i = 1, \ldots, \#leaf$ do
            switch $[\boldsymbol{M}_n^l(j,i), \boldsymbol{M}_n^e(j,i)]$ do
                case $[0,0]$ Continue;
                case $[0,1]$ $f++$;    ▷ one false alarm
                case $[1,0]$ $d++$;    ▷ one miss detection
                case $[1,1]$ $\boldsymbol{e} = [\boldsymbol{e};\ e_{le}]$.
3. for $\tau = 0 : 0.01 : 2$ do
    $F(\tau) = \frac{f+\text{sum}(\boldsymbol{e}>\tau)}{\sum_n \text{sum}(\boldsymbol{M}_n^l)}$; $D(\tau) = \frac{d+\text{sum}(\boldsymbol{e}>\tau)}{\sum_n \text{sum}(\boldsymbol{M}_n^l)}$;
    $E(\tau) = \text{mean}(\boldsymbol{e} \leqslant \tau)$; $T(\tau) = \frac{\text{sum}(\boldsymbol{e}\leqslant\tau)}{\sum_n \text{sum}(\boldsymbol{M}_n^l)}$.
```

Since $\frac{dJ}{d\boldsymbol{p}_i} = \frac{dJ_1}{d\boldsymbol{p}_i} + \lambda_1\frac{dJ_2}{d\boldsymbol{p}_i} + \lambda_2\frac{dJ_3}{d\boldsymbol{p}_i}$, due to limited space we use $\frac{dJ_2}{d\boldsymbol{p}_i}$ as an example of how to compute $\frac{dJ}{d\boldsymbol{p}_i}$. Specifically,

$$\frac{dJ_2}{d\boldsymbol{p}_i} = \frac{2}{K}\left(\frac{\partial\boldsymbol{T}_i(\boldsymbol{W}^{-1})}{\partial\boldsymbol{p}_i}\right)^{\mathsf{T}}\left(\sum_{i=1}^{A}\boldsymbol{T}_i(\boldsymbol{W}^{-1}(\boldsymbol{X};\boldsymbol{p}_i)) - \boldsymbol{m}\right),$$

where $\frac{\partial\boldsymbol{T}_i(\boldsymbol{W}^{-1})}{\partial\boldsymbol{p}_i} = \nabla\boldsymbol{T}_i^x\frac{\partial\boldsymbol{W}_x^{-1}}{\partial\boldsymbol{p}_i} + \nabla\boldsymbol{T}_i^y\frac{\partial\boldsymbol{W}_y^{-1}}{\partial\boldsymbol{p}_i}$, $\nabla\boldsymbol{T}_i^x$ and $\nabla\boldsymbol{T}_i^y$ are the gradient images of the template mask $\boldsymbol{T}_i$ at the x and y direction respectively, and $\frac{\partial\boldsymbol{W}_x^{-1}}{\partial\boldsymbol{p}_i}$ and $\frac{\partial\boldsymbol{W}_y^{-1}}{\partial\boldsymbol{p}_i}$ can be easily computed based on the inverse function of Eqn. 2.

Given a multi-day fluorescence video, we apply leaf alignment [4] on the last frame and employ the proposed tracking toward the first frame. This backward tracking is based on two considerations: the last frame is favorable to leaf alignment due to the maximum number and size of leaves; it is easier to handle disappearing leaves in backward tracking than the emerging leaves in forward tracking. Also, if the area of any leaf candidate at one frame is less than a threshold s, the area of the smallest interested leaf, we remove it from $\mathbb{M}$ and $\boldsymbol{P}$.

3. EXPERIMENTS

Our publicly available dataset[1] has 30 Arabidopsis fluorescence videos taken across 5 days. Each video has 389 frames with the plant size ranging from 40×40 to 100×100 pixels. For each video we manually labeled two tips of all visible leaves for 5 frames, one from the middle of each day. While tracking is applied to all frames of 30 videos, the quantitative evaluation is conducted on the 1100 total labeled leaves.

3.1. Performance Evaluation

There are prior work in evaluating multi-target tracking performance [22,23] . However they may not be directly applicable to our problem since we perform three tasks, leaf segmentation, alignment and tracking, simultaneously. Therefore we

[1] `http://www.cse.msu.edu/~liuxm/plant`

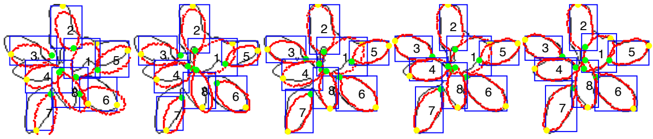

Fig. 3. A synthetic tracking result at ite. #1, 10, 30, 60, 100. Yellow/green dots are the estimated outer/inner leaf tips. Red contour is $\boldsymbol{W}(\boldsymbol{U};\boldsymbol{p})$. Blue box encloses the edge points matching $\boldsymbol{W}(\boldsymbol{U};\boldsymbol{p})$.

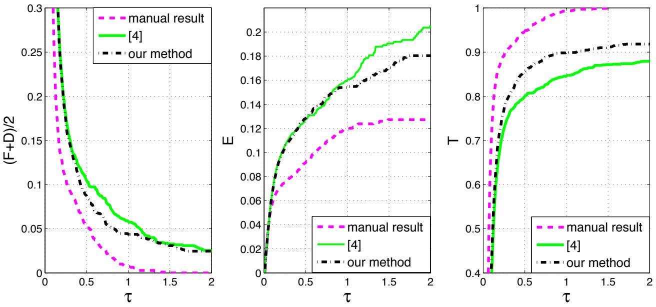

Fig. 4. Performance comparison using $\frac{1}{2}(F + D)$, E, and T vs. τ.

develop an evaluation scheme (Algorithm 1) to quantitatively measure the performance of each individual task.

We first explain the input of Algorithm 1. For the n^{th} plant video, a $5 \times L_n$ matrix $\boldsymbol{S}_n^l$ is generated, where L_n is the number of unique labeled leaves in this plant. Each element in $\boldsymbol{S}_n^l$ stores the locations of two leaf tips for an image/leaf combination. Since not every leaf is visible in all frames, a binary matrix $\boldsymbol{M}_n^l$ of the same size indicates whether a leaf is labeled (i.e, visible) or not. Similarly, the results of a tracking algorithm are stored as two $5 \times A_n$ matrices $\boldsymbol{S}_n^e$ and $\boldsymbol{M}_n^e$, where A_n is the number of unique estimated leaves.

For each video, we first build leaf correspondence between $\boldsymbol{S}_n^e$ and $\boldsymbol{S}_n^l$. A $L_n \times A_n$ error matrix is generated to store the average tip-based errors for all leaf pairs over 5 frames. We find $min(L_n, A_n)$ leaf pairs and perform sequence-wise index switch. For each video frame, we compute the number of miss detection d, number of false alarm f and tip-based error e_{le} for all leaves with correspondence. $d, f, \boldsymbol{e}$ are accumulated for all frames of all videos. Finally we compute four metrics by applying a varying threshold τ to $\boldsymbol{e}$. **Landmark error** E is the average of all tip-based errors that are less than τ. **False alarm** F and **Miss detection** D are the percentages of respective counts. Both F and D can attribute to two sources: the $|L_n - A_n|$ leaves without correspondence, and the corresponded pair whose tip-based error is larger than τ. **Tracking consistency** T is the percentage of leaf pairs whose tip-based errors are less than τ.

3.2. Experimental Results

We experimentally set parameters as $\lambda_1 = 3$, $\lambda_2 = 20$, and $s = 64$. Since θ and r are more sensitive than t_x and t_y, the step size α is 0.01 for θ and r, and 1 for t_x and t_y.

To test the feasibility of our method, we show a synthetic example in Fig. 3 where the initialization is much further than the typical between-frame difference in plant videos. Our method performs well for this challenging situation. Note that while some leaves converge earlier, other leaves still continue to improve their alignment in later iterations.

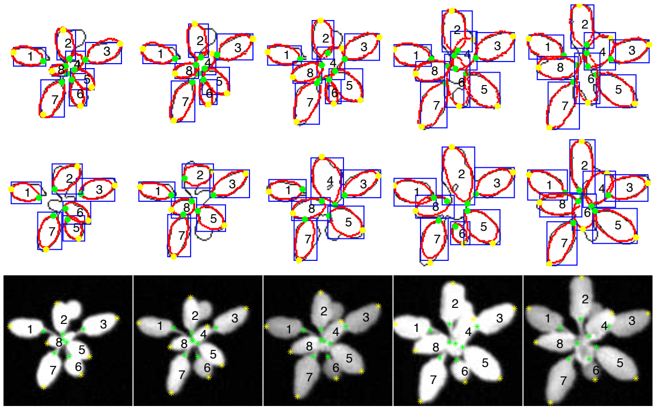

Fig. 5. Leaf tracking results. Vertically from top to bottom each row shows: results of our tracking method; results of [4]; test frames with manual labels. Each column is one labeled frame from day 1-5.

We compare the proposed method with the leaf alignment work [4]. Since [4] is an image-based method, we build the leaf correspondence of one video using the leaf center distances between two frames, which enriches a basic tracking capability to [4]. Similarly to [4], we treat the "manual result" as our upper bound, by using the labeled tips to find the corresponding leaf with the minimum tip-based error in all leaf candidates. No other baseline is chosen due to the lack of prior work performing all three tasks simultaneously.

Figure 4 shows our method performs generally better than [4], especially for tracking consistency T. Specifically, when τ is relatively small, the performance of the proposed method and [4] are very similar. This is because at this point, we have very strict requirement on the accuracy of tip estimation. Both methods work well on easy-to-align leaves. As τ increases, the tip estimation requirement is relaxed. Our method can generate more well-aligned $(\frac{F+D}{2})$ and well-tracked leaves (T) without increasing E too much.

The tracking results of one video are shown in Fig. 5. Our tracking method is clearly superior in maintaining tracking consistency especially in early days when leaves are small, which is likely to be missed by [4]. In contrast, since [4] is image-based alignment, the alignment results can be very different between frames, e.g., leaf 2 in day 4 is actually two leaves but aligned as one. Note that our tracking method can handle the leaf disappear problem, but not the emerging of new leaves, which will be the future work. In terms of *efficiency*, the tracking method and [4] take ~ 2 and ~ 16 seconds per frame respectively using a Matlab implementation.

4. CONCLUSIONS

This paper proposes a novel framework for multi-leaf tracking from fluorescence videos. Experimental results show the effectiveness of our method, which is potentially applicable to a wide variety of plants in studying plant phenotyping.

978-1-4799-5751-4/14/$31.00 ©2014 IEEE

5. REFERENCES

[1] Thomas A MacMahon and John Tyler Bonner, *On size and life*, Scientific American Books, 1983.

[2] Xu Zhang, Ronald J Hause, and Justin O Borevitz, "Natural genetic variation for growth and development revealed by high-throughput phenotyping in Arabidopsis thaliana," *G3: Genes| Genomes| Genetics*, vol. 2, no. 1, pp. 29–34, 2012.

[3] Ladislav Nedbal and John Whitmarsh, "Chlorophyll fluorescence imaging of leaves and fruits," in *Chlorophyll a Fluorescence*, pp. 389–407. Springer, 2004.

[4] Xi Yin, Xiaoming Liu, Jin Chen, and David Kramer, "Multi-leaf alignment from fluorescence plant images," in *IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2014.

[5] Chin-Hung Teng, Yi-Ting Kuo, and Yung-Sheng Chen, "Leaf segmentation, its 3D position estimation and leaf classification from a few images with very close viewpoints," in *Image Analysis and Recognition*, pp. 937–946. Springer, 2009.

[6] Jonas Vylder, Daniel Ochoa, Wilfried Philips, Laury Chaerle, and Dominique Straeten, "Leaf segmentation and tracking using probabilistic parametric active contours," in *Computer Vision/Computer Graphics Collaboration Techniques*, vol. 6930 of *Lecture Notes in Computer Science*, pp. 75–85. Springer Berlin Heidelberg, 2011.

[7] Long Quan, Ping Tan, Gang Zeng, Lu Yuan, Jingdong Wang, and Sing Bing Kang, "Image-based plant modeling," in *ACM SIGGRAPH*, Boston, Massachusetts, 2006, pp. 599–604, ACM.

[8] Derek Bradley, Derek Nowrouzezahrai, and Paul Beardsley, "Image-based reconstruction and synthesis of dense foliage," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 74, 2013.

[9] Yann Chéné, David Rousseau, Philippe Lucidarme, Jessica Bertheloot, Valérie Caffier, Philippe Morel, Étienne Belin, and François Chapeau-Blondeau, "On the use of depth camera for 3D phenotyping of entire plants," *Computers and Electronics in Agriculture*, vol. 82, pp. 122–127, 2012.

[10] Guillaume Cerutti, Laure Tougne, Julien Mille, Antoine Vacavant, and Didier Coquin, "Understanding leaves in natural images–a model-based approach for tree species identification," *Comput. Vision and Image Understanding*, vol. 117, no. 10, pp. 1482–1501, 2013.

[11] Sofiene Mouine, Itheri Yahiaoui, and Anne Verroust-Blondet, "Advanced shape context for plant species identification using leaf image retrieval," in *Proc. ACM Int. Conf. Multimedia Retrieval (ICMR)*, Hongkong, China, 2012, pp. 49:1–49:8, ACM.

[12] Neeraj Kumar, Peter N Belhumeur, Arijit Biswas, David W Jacobs, W John Kress, Ida C Lopez, and João VB Soares, "Leafsnap: A computer vision system for automatic plant species identification," in *Proc. European Conf. Computer Vision (ECCV)*, pp. 502–516. Springer, 2012.

[13] Guillaume Cerutti, Laure Tougne, Julien Mille, Antoine Vacavant, Didier Coquin, et al., "A model-based approach for compound leaves understanding and identification," in *Proc. Int. Conf. Image Processing (ICIP)*, 2013, pp. 1471–1475.

[14] T. Cootes, C. Taylor, and A. Lanitis, "Active shape models: Evaluation of a multi-resolution method for improving image search," in *Proc. British Machine Vision Conf. (BMVC)*, York, UK, Sept. 1994, vol. 1, pp. 327–336.

[15] Iain Matthews and Simon Baker, "Active appearance models revisited," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 135–164, November 2004.

[16] Harry G Barrow, Jay M Tenenbaum, Robert C Bolles, and Helen C Wolf, "Parametric correspondence and Chamfer matching: Two new techniques for image matching," Tech. Rep., DTIC Document, 1977.

[17] Xiaoming Liu, Peter Tu, and Frederick Wheeler, "Face model fitting on low resolution images," in *Proc. British Machine Vision Conf. (BMVC)*, Edinburgh, UK, Sept. 2006, vol. 3, pp. 1079–1088.

[18] Yan Tong, Xiaoming Liu, Frederick W. Wheeler, and Peter Tu, "Automatic facial landmark labeling with minimal supervision," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Miami, FL, June 2009.

[19] Ming-Yu Liu, Oncel Tuzel, Ashok Veeraraghavan, and Rama Chellappa, "Fast directional Chamfer matching," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 1696–1703.

[20] Tianyang Ma, Xingwei Yang, and Longin Jan Latecki, "Boosting Chamfer matching by learning Chamfer distance normalization," in *Proc. European Conf. Computer Vision (ECCV)*, pp. 450–463. Springer, 2010.

[21] Xiaoming Liu, "Video-based face model fitting using adaptive active appearance model," *Image and Vision Computing*, vol. 28, no. 7, pp. 1162–1172, July 2010.

[22] Dominic Schuhmacher, Ba-Tuong Vo, and Ba-Ngu Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Robot. Autom.*, vol. 56, no. 8, pp. 3447–3457, 2008.

[23] Luka Čehovin, Matej Kristan, and Aleš Leonardis, "Is my new tracker really better than yours?," in *IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2014.

978-1-4799-5751-4/14/$31.00 ©2014 IEEE