

A High-throughput Phenotyping Pipeline for Image Processing and Functional Growth Curve Analysis

Ronghao Wang¹, Yumou Qiu^{2*}, Yuzhen Zhou¹, and James C. Schnable³

¹Department of Statistics, University of Nebraska-Lincoln, Lincoln, 68503, USA

²Department of Statistics, Iowa State University, Ames, 50011, USA

³Center for Plant Science Innovation, Department of Agronomy and Horticulture, University of Nebraska-Lincoln, Lincoln, 68503, USA

*yumouqiu@iastate.edu

ABSTRACT

High-throughput phenotyping system has become more and more popular in plant science research. The data analysis for such a system typically involves two steps: plant feature extraction through image processing and statistical analysis for the extracted features. The current approach is to perform those two steps on different platforms. We develop the package “*implant*” in R for both robust feature extraction and functional data analysis, which is able to provide statistical inference for the plant traits directly from the input images. For image processing, the “*implant*” package provides methods including thresholding, hidden Markov random field model, morphology operations and etc. For the growth curve analysis, this package can produce nonparametric curve fitting with its confidence region for plant growth. A functional ANOVA model to test for the treatment and genotype effects of the growth curve dynamics is also provided.

¹ Introduction

² High-throughput phenotyping is a newly emerging technique in the plant science research. Many automated systems have been
³ constructed both in greenhouse and field to study plant features ([Bucksch et al., 2014](#); [Fahlgren et al., 2015](#); [Hairmansis et al., 2014](#); [Miller et al., 2007](#)). One of the main innovation is to use automated cameras to take raw images for plants of interest.
⁵ Several types of high resolution images, including RGB, infrared, Fluorescence and hyperspectral, are recorded for a large
⁶ number of plants at designed observation time points. From the high-throughput system, we are able to process and extract
⁷ useful phenotypical features from the recorded images of the plants, such as plant height, width, size and etc. Traditionally,
⁸ researchers need to measure those features manually, which is not only time-consuming, but also requires a large number of
⁹ human power. For some traits like biomass, we need to destroy the plant to take the measurement, which makes the study of
¹⁰ growth curve for a single plant impossible. Compared to the traditional methods, the high-throughput system is able to provide
¹¹ the plant features of interest in a more efficient, accurate and non-destructive way.

12 In order to extract the phenotypical features from images, object segmentation for parts of a plant or the whole plant is
13 necessary. Thresholding is the simplest and the most commonly used method for image segmentation (Davies, 2012), which
14 classifies the image into the foreground class and the background class by a cut-off value on the pixel intensities. The choice
15 of an appropriate threshold level is critical in the thresholding methods, which may change from image to image due to the
16 variation in background and different lighting conditions. The global thresholding method (Davies, 2012) is one method that
17 uses a common threshold for all the pixels. Otsu thresholding (Otsu, 1979; Vala and Baxi, 2013) is another popular thresholding
18 method, which provides a threshold to maximize the variance between the background class and the foreground class. Moreover,
19 it respectively finds thresholds for different channels of RGB images.

20 K-means clustering algorithm (Johnson et al., 2002) is also well-known for image segmentation. This clustering method
21 automatically assigns pixels into similar groups in terms of their intensity values so that the within group variation is minimized.
22 K-means method is free of tuning parameter selection. However, it works well only for simple environment, and it usually
23 classifies dark pixels as plant in the greenhouse images. The segmentation result of K-means can be refined by the Hidden
24 Markov Random Field model (HMRF) (Celeux et al., 2003). HMRF is a hierarchical model with a hidden layer of Markov
25 random field to model the class label of each pixel. The Markov model captures the spatial dependence of each pixel to its
26 neighborhood. In a plant image, the plant pixel is more likely to be surrounded by other plant pixels than background pixels.
27 Given the class label of a pixel, its intensities follow a normal distribution with mean and variance parameters that are the
28 same across all the pixels within the same class. As the thresholding and K-means methods ignore the spatial structure of an
29 image, the HMRF model is able to provide a more accurate classification of pixels by incorporating their neighborhood class
30 information. Also see Zhang et al. (2001) for an application of HMRF in the segmentation of brain MRI images.

31 Based on the thresholding methods, several platforms have been developed for the analysis of high-throughput plant
32 phenotyping, including PlantCV (Gehan et al., 2017), ImageHarvest (Knecht et al., 2016) and HTPheno (Hartmann et al., 2011).
33 Those soft-wares have admitted procedures in processing high throughput plant images to extract phenotypical features, such
34 as plant height and width, etc. However, these platforms solely focus on processing images of plants. There is a lack of
35 functionality on statistical modeling and inference for plant traits.

36 Given a good segmentation, the measurements of phenotypical traits (such as heights, widths and size) can be efficiently
37 extracted from plant images. These numerical measurements can be used for analyzing genotype and treatment effects on the
38 dynamics of the plant growth over time. In traditional growth curve analysis, the point-wise Analysis of Variance (ANOVA) (St
39 et al., 1989) approach is widely applied at each measurement time point. However, this approach analyzes each observation
40 time separately, and thus cannot reflect the real dynamics of plant growth. Parameter modeling for the growth curve is another
41 popular tool. However, fitting the parametric models requires measurements of the plant traits over the whole growth stage
42 which may not be available for some experiments, and the temporal dependence feature of the data are usually ignored in this
43 approach. Functional ANOVA (Ramsay et al., 2005) is a recent nonparametric method for analyzing plant trait data collected
44 over time (Xu et al., 2018). Instead of parametric regression, smoothing splines (Gu, 2013; Ramsay et al., 2005) or local

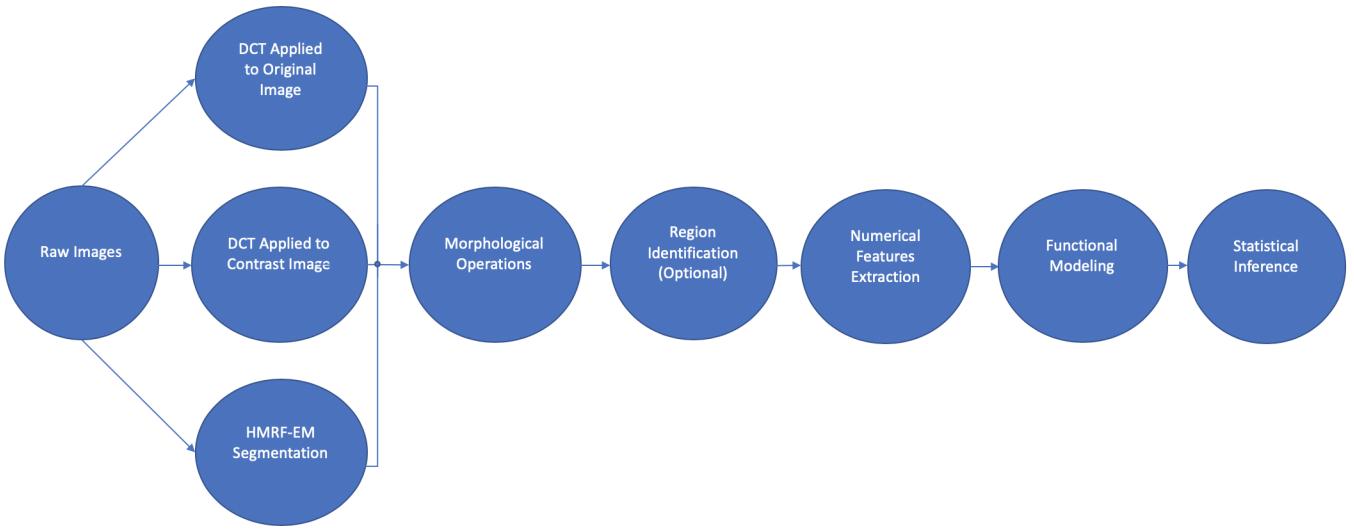


Figure 1. Flow chart of the proposed “*implant*” Pipeline. In the first step of segmentation, the DCT procedure can be applied to the contrast image (panel (c) of Figure 2) between the plant in panel (a) and the empty pot in panel (b) if the image of empty pot is available in the experiment. Multiple methods can be jointly applied to take the intersection of the segmented plants from each method. See panel (f) in Figure 2 as an example.

45 polynomial regression ([da Silva and Opsomer, 2009](#)) are used to estimate the plant growth. This approach is non-parametric,
 46 fully data-driven and adaptive to temporal dependence of the data. Despite those advantages, the implementation of Functional
 47 ANOVA is non-trivial. The current R package “*fda*” ([Ramsay et al., 2010](#)) for functional data analysis is complicated, and it is
 48 difficult to use for non-statisticians. There is no guideline of implementation for plant phenotyping data.

49 To handle the need for image processing, trait extraction and functional ANOVA for the high-throughput plant phenotyping
 50 data, we develop an R package “*implant*” that involves both the pre-processing of the plant images and the functional data
 51 analysis for the extracted features. The “*implant*” package conducts nonparametric functional estimation for the plant growth
 52 curve and the covariate effects on the growth curve over time. This package provides the statistical inference and confidence
 53 bands for those estimated curves as well. It constitutes a pipeline for automatically analyzing the high-throughput phenotyping
 54 data as shown in Figure 1.

55 The flow chart in Figure 1 illustrates the main steps of this pipeline: from raw RGB images to segmentation, feature
 56 extraction, functional data modeling and finally statistical estimation and inference. In the image processing part, we consider
 57 two segmentation methods: Double-Driteria Thresholding (DCT) and Hidden Markov Random Field model with EM Algorithm
 58 (HMRF-EM). The first DCT method applies thresholding on two criteria: green contrast intensity and average RGB intensity.
 59 The intersection of both results is considered as the segmented plant. Notice that if the image of an empty pot is available,
 60 DCT can be applied on the contrast image between the plant and the empty pot as demonstrated in panels (a)–(c) in Figure
 61 2. The other segmentation method is based on the Hidden Markov Random Field, which refines the results from K-means
 62 clustering by incorporating the class information of neighborhood pixels in the HMRF model, and predicts the updated class

63 label based on the maximum posterior probability. Morphological erosion and dilation operations (Comer and Delp, 1999;
64 Goyal, 2011) are included in our package, which can be applied following the DCT and HMRF to screen out the noises in
65 the background. Functions for automatic region identification is also available. Based on the segmented images, our package
66 provides measurements for the height, width and size of the plants.

67 For the functional modeling and statistical inference, the pipeline provides a general model for functional data, which is
68 able to estimate both the main effects and interaction effects. Our package can deal with irregular observation time points,
69 which is common in large experiments where imaging for all the plants cannot be finished in one day. In the functional model,
70 we also estimate the covariance matrix of the regression error over time. Based on such estimated covariance, this pipeline
71 constructs confidence regions for the estimated curves. Those confidence regions can demonstrate the statistical significance for
72 the treatment and genotype effects on the plant growth curves. A real data example is provided in the following sections to
73 illustrate the approach of functional data analysis by our package. In a summary, we develop a comprehensive pipeline for
74 the analysis of high-throughput plant phenotyping data that includes RGB image pre-processing, plant feature extraction and
75 functional estimation and inference for the treatment effects on growth curve dynamics.

76 **Implementation and Results**

77 In this section, we illustrate the implementation of our “implant” package by a maize experiment conducted at the University
78 of Nebraska-Lincoln (UNL) Greenhouse Innovation Center. The package and the detail documentation is available online at
79 <https://github.com/rwang14/implant>. Sample maize images and the data of extracted plant size from the whole
80 experiment are also attached in the package. Detailed descriptions for the methods are presented in the Method section.

81 **Experiment**

82 The experiment involved 420 maize plants with 140 different genotypes. There were three replicates for each genotype. The
83 pots in the greenhouse were divided into three blocks based on the layout of the belt conveyor system. We conducted the
84 randomized complete block design (RCBD) such that each of the 140 genotypes was randomly located within a block and the
85 three replicates of the same genotype were assigned to different blocks. Each plant was imaged about every two or three days
86 from May to July, and the imaging time points were irregular due to the large number of plants in the experiment.

87 **Image Processing**

88 Our methods focus on processing the RGB images of plants. To extract plant features, we need to first segment the RGB image
89 into plant part and non-plant part. We introduce two processing procedures for this purpose: the Double Criteria Thresholding
90 (DCT) procedure and Hidden Markov Random Field model with EM algorithm (HMRF-EM).

91 **1. Image Segmentation Using DCT**

92 Figure 2 shows the general process of the DCT procedure for one of the plant images from the experiment. Here, panels (a) and
93 (b) are the RGB maize image and the empty pot image without a plant, respectively. panel (d) is obtained by applying function

```

94 imageB = imageBinary(original_image, weight = c(-1, 2, -1),
95 threshold1 = 30 / 255, threshold2 = 0.02)

96 where "threshold1" is applied to the sum of the RGB intensities, and "threshold2" is applied on the contrast intensity by
97 the specified weight in the function. The two thresholds are to delete the black pixels and to segment the plant green pixels,
98 respectively. We choose a small level 0.02 for the second threshold to retain most part of the plant. The background noises in
99 panel (d) can be much reduced by applying the DCT procedure on the contrast image in panel (c) resulting from the difference
100 between panels (a) and (b). The image in panel (e) is obtained by setting "threshold1 = 0.7", "threshold2 = 0" and "weight =
101 c(1, -2, 1)" in the "imageBinary" function for panel (c). Then we take intersection between (d) and (e) to obtain (f). As we
102 observe from panel (f), most of the background noises are eliminated and the plant body is segmented well. Those thresholding
103 parameters work consistently well over the whole experiments for the UNL greenhouse. The double criteria procedure makes
104 the results less sensitive to the threshold levels than the procedure using only one criterion. Though for a different system, those
105 parameters need to be properly tuned for good segmentation results. In the case of no empty-pot images, we should set a more
restrict threshold for the green contrast intensities.

```

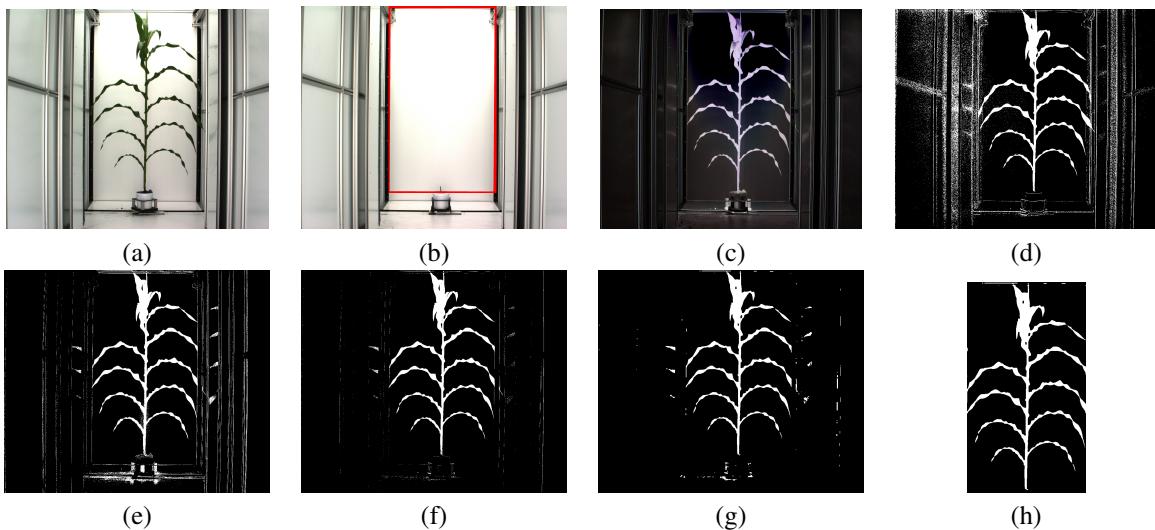


Figure 2. (a) Original plant image. (b) Original empty pot image; the red square is the identified region of interest by the functions “*ColorB*” and “*ColorG*”. (c) Contrast of (a) and (b). (d) Segmented image of (a) using DCT (e) Segmented image of (c) using DCT. (f) Intersection of (d) and (e). (g) Dilated-Eroded-Eroded-Dilated image of (f). (i) Final segmented image by identifying the region of interest.

```

106
107 Morphological operations (Comer and Delp, 1999) can be applied on the thresholding results to further reduce the
108 segmentation errors. Those operations can be performed by the “dilation” and “erosion” functions in our packages as follows

```

```

109 imageBD = dilation(imageB, mask = matrix(1, 5, 5))
110 imageBDE = erosion(imageBD, mask = matrix(1, 5, 5))
111 imageBDEE = erosion(imageBDE, mask = matrix(1, 3, 3))
112 imageBDEED = dilation(imageBDEE, mask = matrix(1, 3, 3))

```

113 which results to panel (g), where “mask” is a structuring matrix specifying the neighborhood structure of a pixel (Goyal, 2011).
114 We call a Dilation followed by an Erosion as a morphological closing operation, and an Erosion followed by a Dilation as a
115 morphological opening operation.

116 Region of interest can be automatically identified by some specific characteristics on the background of an imaging system.
117 For the UNL greenhouse, we can identify the inner black bars and the border-top of the pot to obtain the region of interest for
118 the plants; see the red rectangle in panel (b). Notice that this identification strategy is for the dataset from the UNL greenhouse
119 system only. Different systems need different strategies for locating the region of interest. It is worth mentioning that although
120 identifying the region of interest can help us easily remove most of the background noises, parts of the plants might be lost as
121 well. See panel (h) as the chopped image by the identified region in panel (b).

122 **2. Image Segmentation Using HMRF-EM**

123 The segmentation method by HMRF model is also provided in the package. Compared to the former thresholding procedure,
124 HMRF model is fully data driven and free of tuning parameter selection. Panel (c) in Figure 3 shows the segmentation result
125 by the HMRF-EM algorithm with the K-Means clustering result in panel (b) as an initial class assignment for each pixel.
126 From panel (c), we see the HMRF-EM algorithm provides a quite good segmentation for the plant with little classification
127 errors. Comparing to the K-Means result, the HMRF is able to fill in the missing plant pixels by using their neighborhood class
128 information. Comparing to the thresholding result in panel (f) of Figure 2, the result from HMRF approach eliminates most of
129 the background noises. This method is implemented by the function “*HMRF_EM*” in the package as

130 `HMRF_EM(X, Y, ...)` \$image_matrix

131 where X is a matrix of initial class label (for example, results from K-means clustering), and Y is a matrix of relative green
132 intensity. Detailed information on other arguments of this function can be found in the help documentation of the “implant”
133 package.

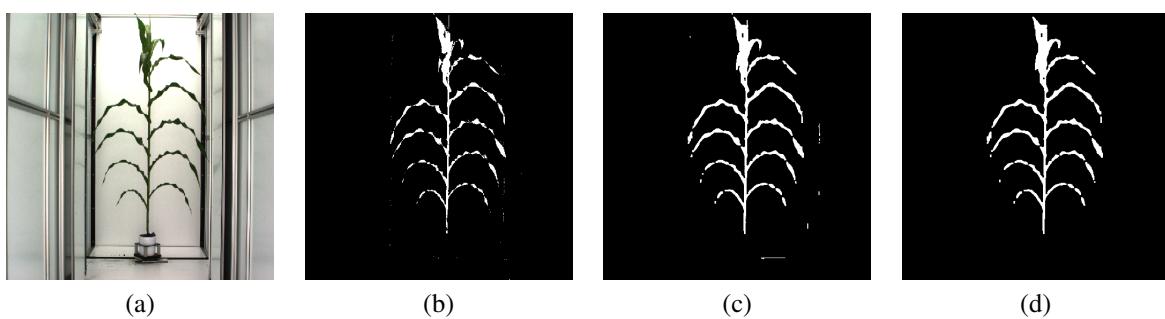


Figure 3. (a) Original image. (b) Initial classification using k-means. (c) Segmentation result using HMRF-EM. (d) Applying morphological closing and opening to (c).

134 In order to remove the background noises in the HMRF result in panel (c) of Figure 3, we could apply the morphological
135 closing operation followed by a morphological opening operation with a 3-by-3 square structural element. Panel (d) in Figure
136 3 shows the final result of this procedure, which is better than the result from morphological operation on the thresholding

137 method in panel (g) of Figure 2. The HMRF method can generally get good segmentation result without the need of identifying
 138 the region of interest, which broadens the scope of its application. Moreover, it can be used to refine the segmentation results
 139 obtained by other methods.

140 Plant Feature Extraction

141 Based on the segmented images, we can extract the phenotypical features of a plant. Given the information of the pixel size in
 142 millimeters, we can obtain plant height, width, and size using the functions

```
143 extract(processed_image, Xsize = 1, Ysize = 1,...)$height
144 extract(processed_image, Xsize = 1, Ysize = 1,...)$width
145 extract(processed_image, Xsize = 1, Ysize = 1,...)$size
```

146 where “processed_image” is a segmented image of a plant by either the thresholding or HMRF methods as those in panel (h) of
 147 Figure 2 and panel (d) of Figure 3, and “Xsize” and “Ysize” are the actual horizontal and vertical lengths of one pixel.

148 Functional ANOVA

149 Based on the numeric phenotypical parameters we obtained by processing images, we can study the relationship between
 150 the plant sizes and treatments using functional data models. Let $y_i(t)$ be the size of the i^{th} plant measured at time t , where
 151 $i = 1, \dots, 420$. We treat genotype and block as fixed effects. There are 140 different genotypes and 3 blocks in this study. We
 152 can use $\mathbf{G}_i = (G_{ik})_{k=2}^{140}$ as the categorical indicator of the i^{th} plant genotype. Specifically, G_{ik} is set to one if the plant has the k^{th}
 153 genotype; otherwise, $G_{ik} = 0$. For example, if the i^{th} plant has the 2^{nd} genotype, then $\mathbf{G}_i = (0, 1, 0, \dots, 0)$. And G_{i1} being zeros
 154 represents the 1^{st} genotype, which is treated as the baseline. Similarly, $\mathbf{P}_i = (P_{ik})_{k=2}^3$ is defined to indicate the block that the i^{th}
 155 plant belongs to, and the first block is set as the baseline.

Our functional ANOVA model for analyzing the relationship between plant size and genotype and block is:

$$y_i(t) = \mu(t) + \sum_{k=2}^{140} G_{ik} g_k(t) + \sum_{k=2}^3 P_{ik} p_k(t) + \varepsilon_i(t), \quad (1)$$

156 where $\mu(t)$ is the growth function of the plant with the 1^{st} genotype (Genotype 129) from the 1^{st} block, $g_k(t)$ s are the genotype
 157 effect functions, $p_k(t)$ s are the fixed block effect functions, and the residuals $\varepsilon_i(t)$ s are modeled by independent random
 158 processes with mean zeros.

The observation time points in this study are irregular. Let m_i be the number of days that the i^{th} plant was imaged, which
 could vary from plant to plant. Let t_{ij} be the j^{th} observation time of the i^{th} plant, where $j = 1, \dots, m_i$. Then, our model can be
 described as:

$$y_i(t_{ij}) = \mu(t_{ij}) + \sum_{k=2}^{140} G_{ik} g_k(t_{ij}) + \sum_{k=2}^3 P_{ik} p_k(t_{ij}) + \varepsilon_i(t_{ij}). \quad (2)$$

159 The parameter estimation is implemented by the function “fanova_mean” in our package, i.e.,

```
160 fit = fanova_mean(Y.na.mat, X, formula, ...)
```

161 where X is a matrix of explanatory variables, $Y.na.mat$ is the matrix of the extracted features, and $formula$ specifies the model
162 we use, namely, Model (2) in this example. For both the covariate matrix X and the response matrix $Y.na.mat$, each row gives
163 the values for different plants. The columns of X and $Y.na.mat$ correspond to explanatory variables and observation dates,
164 respectively. If a measurement of a plant is missing on a particular date, the value is filled by “NA” in the matrix. More details
165 can be found in the help documentation.

166 Afterwards, we can test the significance of the treatment effects of interest by constructing the corresponding confidence
167 regions with the function “*CI.trt.diff*” in the package, i.e.,

```
168 CI.trt.diff(fit, j1, j2, ...)
```

169 where fit is the output from *fanova_mean*, and $j1$ and $j2$ specify the columns of the design matrix corresponds to the treatment
170 of interest. For example, the confidence region of the block effect function $p_3(\cdot)$ infers whether there is significant difference in
171 plant size between block 1 and block 3, given the same plant genotype. Figure 4 (a) shows the 95% confidence regions of p_3
172 from day 1 to day 44. We find that, compared to block 1, block 3 does not have significant effect on the size of plants in the
173 early stage of the growth until about the 15th day. As another example, by constructing the confidence regions of the coefficient
174 difference $g_2(\cdot) - g_3(\cdot)$, we can test whether the plant size has any significant difference over time between the 2nd genotype
175 (Genotype 78) and the 3rd genotype (Genotyp 69), given the plants belonging to the same block. Figure 4 (b) shows the 95%
176 confidence region of the two coefficients difference from day 1 to day 44, from where we see no significant difference between
177 them.

178 In general, we can estimate a linear combination of treatment effects and its confidence regions by the following function:

```
179 CI.trt(fit, L, ...)
```

180 where L is a contrast vector under the model (2) specifying the linear combination of the parameters of interest. This includes
181 estimating the average growth curve of a particular genotype over all the blocks. Figure 5 (a) shows the estimated average
182 growth curves of plants Genotype 129 and Genotype 69 with their 95% confidence regions. Since Genotype 129 under block 1
183 is the baseline in the fitted model, the corresponding vector L for Genotype 129 in the function *CI.trt* places 1 on the intercept,
184 1/3 on blocks 2 and 3, and 0 on other coefficients. We can see that Genotype 69 grows faster than Genotype 129 for most of time.
185 Indeed, such figures not only provide the average growth curves, but also show their variations along the time. Similarly, Figure
186 5 (b) shows the growth curves of plants with Genotype 78 and Genotype 69, with their 95% confidence regions, respectively.
187 By comparing Figure 5 (b) to Figure 4 (b), the two figures draw the same conclusion that Genotype 78 and Genotype 69 do not
188 have significant difference.

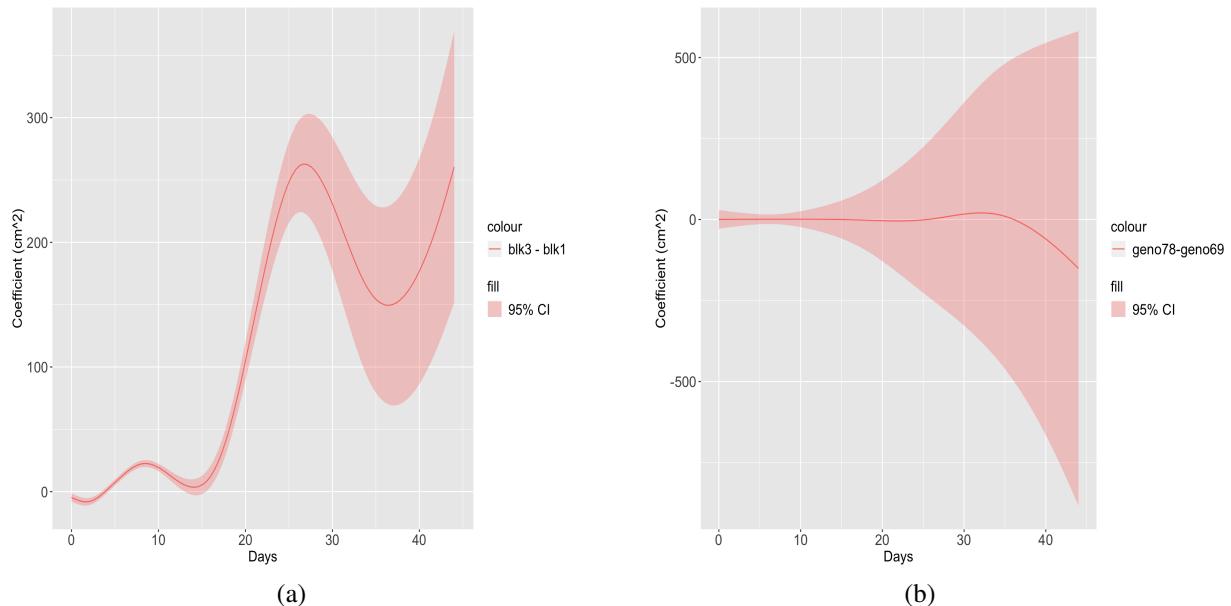


Figure 4. (a) Confidence region of the coefficient of the difference between block 3 and block 1. (b) Confidence region of the coefficient of the difference between the Genotype 78 and Genotype 69.

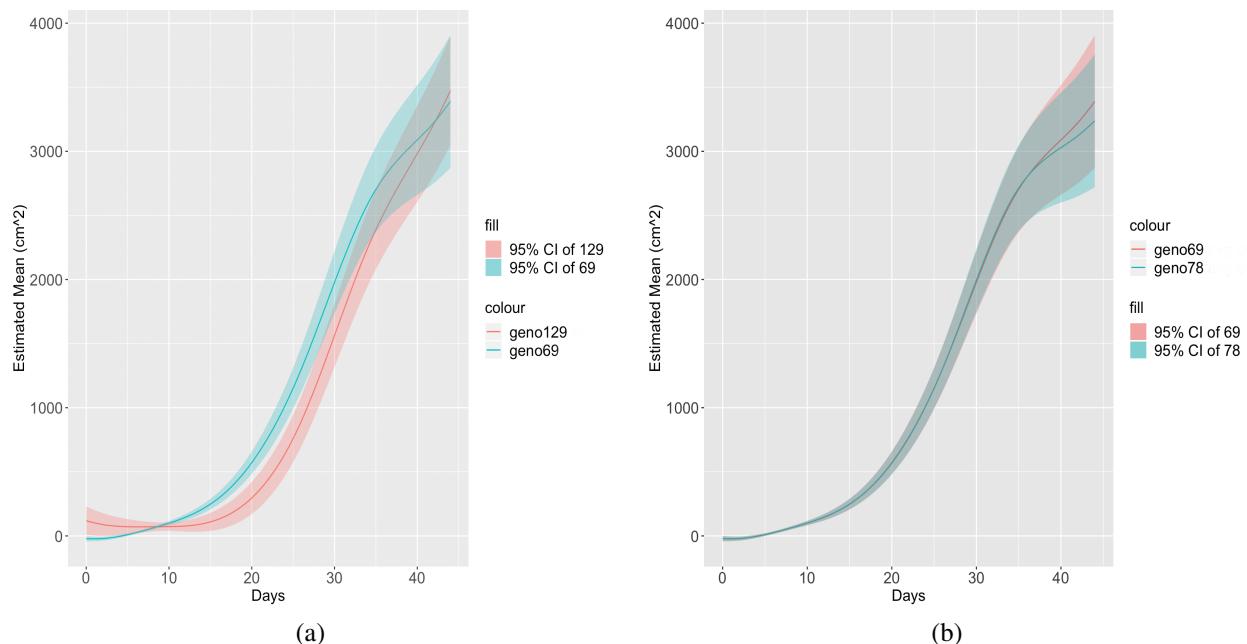


Figure 5. (a) Confidence regions of the estimated mean of the Genotype 129 and Genotype 69, averaging over three blocks, respectively. (b) Confidence regions of the estimated mean of Genotype 78 and Genotype 69, averaging over three blocks, respectively.

189 **Methods**

190 In this section, we introduce the double criteria thresholding method and the hidden Markov random field model for image
191 segmentation, as well as the functional ANOVA for growth curve analysis in detail.

192 **Double Criteria Thresholding Method**

193 Thresholding is a simple but effective method in image segmentation. This method segments the object of interest from its
194 background by comparing pixel intensity to a threshold level. We consider to threshold the relative green intensity ([Ge et al.,
195 2016; Meyer and Neto, 2008](#)) to segment plants. However, the relative green intensity is not decisive to separate the green
196 pixels from the black pixels in the background. To reduce those background noises, we consider an additional criterion to
197 screen out the dark pixels by the average intensity of the red, green, and blue channels of the image. The final result is the
198 intersection of the two thresholding criteria.

199 **Hidden Markov Random Field Model**

200 The hidden Markov random field (HMRF) model is a hierarchical model with an unobserved layer for the pixel class and an
201 observed layer for the pixel intensity given its class. The hidden layer of pixel class is modeled by Markov chain, where the
202 probability of a pixel from the plant category depends on the classes of its neighborhood pixels. As the plant pixel is more
203 likely to be surrounded by plants, this transition probability matrix models the spatial dependence of the pixel classes. We
204 assume that the pixel intensity follows a normal distribution where its mean and variance are determined by the class of this
205 pixel. And, the joint distribution of the unobservable classes for all the pixels follows the Gibbs distribution, according to the
206 Hammersley-Clifford Theorem ([Besag, 1974](#)). The aim of this method is that for each pixel, given the observed pixel intensity,
207 we predict its class label by maximizing the probability that the pixel is classified into this class.

208 We use the relative green intensity as our response variable. In order to fit the HMRF model, we apply the expectation
209 maximization (EM) algorithm ([Wang, 2012; Zhang et al., 2001](#)). By using the segmentation results obtained by K-means
210 clustering as the initial class label for the EM algorithm, we iteratively find the maximal likelihood estimators for the mean
211 and variance of the relative green intensities for the plant class and the background class. Then, the class label of each pixel is
212 predicted by the level that maximizes the posterior probability of the pixel class given the observed intensities. We write a
213 function called “*HMRF_EM*” to implement the HMRF method for plant segmentation in R.

214 **Functional ANOVA**

215 Phenotypical parameters extracted from the processed images can be used as numerical data to build functional data models
216 and draw statistical inference. As we discussed in the introduction part, comparing to point-wise ANOVA in dealing with the
217 analysis of time series phenotypical dataset, functional ANOVA has many advantages. Point-wise ANOVA yields discontinuous
218 growth curves, which cannot reflect the dynamics of growth patterns, while functional ANOVA models continuous growth
219 curves by nonparametric smooth functions ([Xu et al., 2018](#)). Besides, functional ANOVA considers the temporal dependence

220 while the point-wise ANOVA does not. The existing R packages for functional ANOVA are complicated and not easy to use,
 221 especially for non-statisticians. Therefore, we developed a series of functions for functional ANOVA for researchers in plant
 222 science.

Assume that the response variable of our model is one of the phenotypical parameters, e.g., the size of a plant. Let $y_i(t)$ be the size of the i^{th} plant measured at time t , where $i = 1, \dots, n$ with n the number of plants studied. Suppose the mean of the response variable is influenced by q factors. For $i \in \{1, \dots, n\}$, $j \in \{1, \dots, q\}$, we define $\mathbf{A}_{ij} = (A_{ijk})_{k=2}^{\ell_j}$ as the categorical indicator of the j^{th} factor of the i^{th} plant, where ℓ_j is the number of levels that the j^{th} factor has. Specifically, A_{ijk} is set to one if the j^{th} factor of the i^{th} plant has level “ k ”; otherwise, $A_{ijk} = 0$. When all A_{ijk} are zeros, it indicates the j^{th} factor level of the i^{th} plant is “1”. A functional multi-way ANOVA model with interactions can be written in the following form:

$$y_i(t) = \mu(t) + \mathbf{A}_{i1}^\top \mathbf{a}_1(t) + \mathbf{A}_{i2}^\top \mathbf{a}_2(t) + \dots + \mathbf{A}_{iq}^\top \mathbf{a}_q(t) + (\mathbf{A}_{i1}^\top \otimes \mathbf{A}_{i2}^\top) \mathbf{a}_{1,2}(t) + \dots + (\mathbf{A}_{i1}^\top \otimes \dots \otimes \mathbf{A}_{iq}^\top) \mathbf{a}_{1,2,\dots,q}(t) + \varepsilon_i(t), \quad (3)$$

223 where $\mathbf{a}_j(t)$ are the treatment effect functions of the j^{th} factor with dimension $\ell_j - 1$, $\mathbf{a}_{j_1,j_2}(t)$ are the pairwise interaction effect
 224 functions between factor j_1 and j_2 with dimension $(\ell_{j_1} - 1)(\ell_{j_2} - 1)$, and $\mathbf{a}_{1,2,\dots,q}(t)$ are the interaction effect functions of all
 225 factors with dimensions $\prod_{j=1}^q (\ell_j - 1)$, and $\varepsilon_i(t)$ s are independent random processes with zero means. The above genearl model
 226 provides a full specification of interactions among factors. In real applications, researchers can build up a simpler model with
 227 less interaction terms or even without interactions according to the purpose of the study. A specific real data example has been
 228 illustrated in the previous section.

229 In the following, we briefly describe the estimation procedure. The objective is to estimate the temporal varying coefficient
 230 functions, $\mathbf{a}_1(\cdot)$, ..., $\mathbf{a}_q(\cdot)$, $\mathbf{a}_{1,2}(\cdot)$, ..., and $\mathbf{a}_{1,2,\dots,q}(\cdot)$. First, we approximate all of the coefficient functions with rank K splines
 231 expansions. For example, $a_{1k}(t) = \sum_{\ell=1}^K \beta_{a_{1i},\ell} B_{d_1,\ell}(t)$, where $B_{d_1,\ell}(t)$ is an order d_1 B-spline basis function, and $\beta_{a_{1i},\ell}$ is the
 232 corresponding coefficient. Then, the problem is reduced to estimate all those coefficients $\boldsymbol{\beta}$ s in front of basis functions. We
 233 apply the least squares estimation with penalizations on the d^{th} derivatives of the rank K spline expansion functions. More
 234 details can be found in (Ramsay et al., 2005). For simplicity, we use a common smoothing parameter λ to regularize the
 235 roughness of all coefficient functions, which is often estimated by generalized cross validation (GCV). By minimizing the
 236 penalized error sum of squares, we can obtain the estimators of parameter $\boldsymbol{\beta}$ s. By plugging them back to the rank K splines
 237 expansions, we got the estimated varying coefficient functions in the model. We write the function “*fanova_mean*” in our
 238 package to implement the estimation procedure.

239 The solution of the optimization problem in the estimation procedure is explicit and is linear in terms of the response y_i s.
 240 Hence, it is straightforward to derive the covariance of $\boldsymbol{\beta}$ s given the sample covariance of y s. Afterwards, the covariance of
 241 varying coefficient functions $\mathbf{a}(\cdot)$ s can be obtained as well and confidence intervals of the treatment effects of interest can be
 242 constructed accordingly. We write functions “*CI.trt.diff*” and “*CI.trt*” to construct confidence regions over time for treatment
 243 effects and for average mean curve of plants under some treatments.

244 **Discussion**

245 In this paper, we developed a comprehensive package in R for both feature extraction and functional data analysis for high
246 throughput phenotyping data. For image pre-processing, we proposed Double-Criteria Thresholding method and Hidden
247 Markov Random Field model. The thresholding method is computationally efficient and works well under simple backgrounds.
248 However, its segmentation result gets worse if the background of the imaging system is complicated. The HMRF model is able
249 to provide robust segmentation of plants without identifying the region of interest, but it has a higher computational burden. For
250 the functional modeling and data analysis, we constructed R functions that are easy to implement for estimating the growth
251 curve of plants. Our algorithm generalizes the model considered in ([Xu et al., 2018](#)), and provides confidence regions for the
252 curves of treatment effects.

253 Beside the RGB images, the UNL greenhouse also takes the hyperspectral images for every plant. Comparing to RGB
254 images which only have three channels, the hyperspectral images record the pixel intensities at every 5nm over the whole
255 spectrum, which contain more information than RGB images. The hyperspectral images can be used to separate plant organs
256 and predict chemical concentration within a plant ([Ge et al., 2016](#)). In future works, we will extend the HMRF model and
257 functional ANOVA to hyperspectral images for studying traits from different plant organs.

258 **References**

- 259 Besag, Julian (1974), “Spatial interaction and the statistical analysis of lattice systems.” Journal of the Royal Statistical Society:
260 Series B (Methodological), 36, 192–225.
- 261 Bucksch, Alexander, James Burridge, Larry M York, Abhiram Das, Eric Nord, Joshua S Weitz, and Jonathan P Lynch (2014),
262 “Image-based high-throughput field phenotyping of crop roots.” Plant Physiology, 166, 470–486.
- 263 Celeux, Gilles, Florence Forbes, and Nathalie Peyrard (2003), “Em procedures using mean field-like approximations for markov
264 model-based image segmentation.” Pattern recognition, 36, 131–144.
- 265 Comer, Mary L and Edward J Delp (1999), “Morphological operations for color image processing.” J. Electronic Imaging, 8,
266 279–289.
- 267 da Silva, Damião N and Jean D Opsomer (2009), “local polynomial regression.” Survey Methodology, 165.
- 268 Davies, E Roy (2012), Computer and machine vision: theory, algorithms, practicalities. Academic Press.
- 269 Fahlgren, Noah, Malia A Gehan, and Ivan Baxter (2015), “Lights, camera, action: high-throughput plant phenotyping is ready
270 for a close-up.” Current opinion in plant biology, 24, 93–99.
- 271 Ge, Y., G. Bai, V. Stoerger, and J. C. Schnable (2016), “Temporal dynamics of maize plant growth, water use, and leaf water
272 content using automated high throughput rgb and hyperspectral imaging.” Computers and Electronics in Agriculture, 127,
273 625–632.

- 274 Gehan, Malia A, Noah Fahlgren, Arash Abbasi, Jeffrey C Berry, Steven T Callen, Leonardo Chavez, Andrew N Doust, Max J
275 Feldman, Kerrigan B Gilbert, John G Hodge, et al. (2017), “Plantcv v2: Image analysis software for high-throughput plant
276 phenotyping.” *PeerJ*, 5, e4088.
- 277 Goyal, Megha (2011), “Morphological image processing.” *IJCST*, 2.
- 278 Gu, Chong (2013), *Smoothing spline ANOVA models*, volume 297. Springer Science & Business Media.
- 279 Hairmansis, Aris, Bettina Berger, Mark Tester, and Stuart John Roy (2014), “Image-based phenotyping for non-destructive
280 screening of different salinity tolerance traits in rice.” *Rice*, 7, 16.
- 281 Hartmann, Anja, Tobias Czauderna, Roberto Hoffmann, Nils Stein, and Falk Schreiber (2011), “Htpheno: an image analysis
282 pipeline for high-throughput plant phenotyping.” *BMC bioinformatics*, 12, 148.
- 283 Johnson, Richard Arnold, Dean W Wichern, et al. (2002), *Applied multivariate statistical analysis*, volume 5. Prentice hall
284 Upper Saddle River, NJ.
- 285 Knecht, Avi C, Malachy T Campbell, Adam Caprez, David R Swanson, and Harkamal Walia (2016), “Image harvest: an open-
286 source platform for high-throughput plant image processing and analysis.” *Journal of experimental botany*, 67, 3587–3599.
- 287 Meyer, George E and João Camargo Neto (2008), “Verification of color vegetation indices for automated crop imaging
288 applications.” *Computers and electronics in agriculture*, 63, 282–293.
- 289 Miller, Nathan D, Brian M Parks, and Edgar P Spalding (2007), “Computer-vision analysis of seedling responses to light and
290 gravity.” *The Plant Journal*, 52, 374–381.
- 291 Otsu, Nobuyuki (1979), “A threshold selection method from gray-level histograms.” *IEEE transactions on systems, man, and
292 cybernetics*, 9, 62–66.
- 293 Ramsay, J., J. Ramsay, and B.W. Silverman (2005), *Functional Data Analysis*. Springer Series in Statistics, Springer.
- 294 Ramsay, JO, Hadley Wickham, Spencer Graves, and Giles Hooker (2010), “fda: Functional data analysis. r package version 2.2.
295 6.”
- 296 St, Lars, Svante Wold, et al. (1989), “Analysis of variance (anova).” *Chemometrics and intelligent laboratory systems*, 6,
297 259–272.
- 298 Vala, Hetal J and Astha Baxi (2013), “A review on otsu image segmentation algorithm.” *International Journal of Advanced
299 Research in Computer Engineering & Technology (IJARCET)*, 2, 387–389.
- 300 Wang, Quan (2012), “Hmrfe-m-image: implementation of the hidden markov random field model and its expectation-
301 maximization algorithm.” *arXiv preprint arXiv:1207.3510*.

302 Xu, Yuhang, Yumou Qiu, and James C Schnable (2018), “Functional modeling of plant growth dynamics.” The Plant Phenome
303 Journal, 1.

304 Zhang, Yongyue, Michael Brady, and Stephen Smith (2001), “Segmentation of brain mr images through a hidden markov
305 random field model and the expectation-maximization algorithm.” IEEE transactions on medical imaging, 20, 45–57.

306 **Additional information**

307 The “*implant*” package and its documentation are available on Github at <https://github.com/rwang14/implant>.
308 The raw image data employed in this study are hosted at Holland Computer Center at UNL. A similar public available
309 experiment on maize diversity panel is available at CyVerse under doi 10.7946/P22K7V.x

310 **Competing financial interests**

311 The authors have no competing financial interests.