Learning Computationally Efficient Approximations of Complex Image Segmentation Metrics

Massimo Minervini, Cristian Rusu and Sotirios A. Tsaftaris
IMT Institute for Advanced Studies, Lucca, Italy
{massimo.minervini, cristian.rusu, s.tsaftaris}@imtlucca.it

***Abstract*—Image segmentation metrics have been extensively used in the literature to compare segmentation algorithms among each other, or relative to a ground-truth segmentation. Some metrics are easy to compute (e.g., Dice, Jaccard), others are more accurate (e.g., the Hausdorff distance) and may reflect local topology, but they are computationally demanding. While certain attempts have been made to create computationally efficient implementations of such complex metrics, in this paper we approach this problem from a radically different viewpoint. We construct approximations of a complex metric (e.g., the Hausdorff distance), combining a small number of computationally lightweight metrics in a linear regression model. We also consider feature selection, using sparsity inducing strategies, to restrict the number of metrics employed significantly, without penalizing the predictive power of the model. We demonstrate our methodology with image data from plant phenotyping experiments. We find that a linear model can effectively approximate the Hausdorff distance using even a few features. Our approach can find many applications, but is largely expected to benefit distributed sensing scenarios where the sensor has low computational capacity, whereas centralized processing units have higher computational capabilities.**

I. Introduction

Image segmentation is the process of partitioning an image, according to a criterion that depends on the application. A vast amount of approaches have been proposed for image segmentation, operating on a variety of image features, guided by different criteria, and founded on established mathematical and statistical frameworks (see [1] for a comprehensive survey and a taxonomy of approaches).

The quest for segmentation algorithms has been also met with a search for appropriate evaluation metrics. Such evaluation allows for the comparison of different approaches, the validation of a result against a ground truth, or the optimal choice of parameters of an algorithm for a specific application.

While human evaluation is subjective and not automated (thus, inherently low-throughput), objective yet simple metrics (such as Dice or precision/recall) frequently prove inadequate in distinguishing "good" segmentations from "bad" ones, according to human intuition of the application.

To overcome this limitation, more sophisticated metrics have been proposed that emphasize particular types of error (e.g., over- and under-segmentation [2], topological disagreements such as region splitting and merging [3]), can be robust to noise (e.g., small boundary displacement [4], white noise [5]), or correlate well with human intuition [3], [5], [6].

The Hausdorff distance [5] has been used in a variety of applications, including the validation of image segmentation results [7], [8]. However, it is characterized by high computational complexity. Thus, considerable research effort has been put in designing efficient algorithms for its calculation (either precise or approximate) in relation to specific contexts (e.g., polyhedral objects represented as polygonal meshes), while its exact calculation in the most general case of point sets (e.g., when comparing segmentation results) still lacks an efficient solution [9].

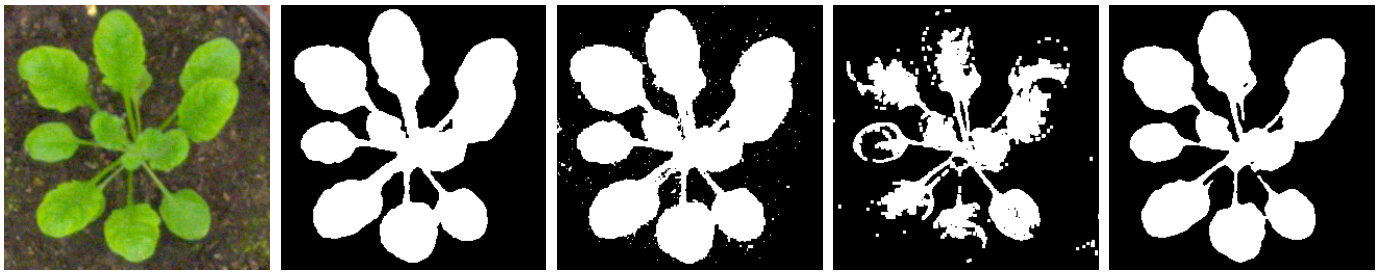
In this paper we approach this problem from a different viewpoint. Rather than investigating an efficient implementation of such metric, or defining a new metric that behaves closely to it in some scenarios, we consider a data driven approach. We propose to learn from the data a (linear) model of an accurate (although computationally complex) metric, using as features simpler to compute metrics. To demonstrate our approach we use the Modified Hausdorff Distance [5], for which we learn efficient linear approximations. We test our methodology on a dataset of plant image segmentations obtained from image based plant phenotyping experiments [10], examples of which are shown in Fig. 1. We find that a linear model is effective at approximating the Modified Hausdorff Distance, and feature selection strategies allow to reduce the number of metrics employed, without penalizing prediction accuracy.

A prime example of where such a framework can be utilized is in sensing devices with limited computational and storage capacity, which may lack the necessary resources to compute complex metrics, when operating on high-resolution images (e.g., low-cost sensors for plant phenotyping experiments [11], [12]) or under real-time considerations (e.g., autonomous vehicle navigation via object matching [5]).

The rest of the paper is organized as follows. In Sec. II, after describing the Modified Hausdorff Distance and the surrogate metrics, we learn approximations of the complex metric. In Sec. III we use selection algorithms to reduce the number of metrics employed. Experimental results are discussed in Sec. IV, while Sec. V offers concluding remarks.

II. Approximate Metrics Via Learning

Let I be an image, S its ground-truth segmentation, and $\hat{S}$ the segmentation obtained by an algorithm, where both S and $\hat{S}$ are binary maps of the same size of I, with pixel values set to 1 to denote foreground (e.g., plant), and pixel values set to 0 to denote background (e.g., tray, pot, earth, moss). The thesis of this paper is that certain complex segmentation evaluation metrics (such as the Modified Hausdorff Distance) can be approximated using linear combinations of simple to compute metrics.



(a) (b) (c) (d) (e)

(c) $\mathrm{MHD} = 0.139$, $\widehat{\mathrm{MHD}} = 0.119$

(d) $\mathrm{MHD} = 1.526$, $\widehat{\mathrm{MHD}} = 1.514$

(e) $\mathrm{MHD} = 0.118$, $\widehat{\mathrm{MHD}} = 0.073$

Fig. 1: Example segmentations of the original image in (a). Shown row wise: (b) ground truth, (c) K-means clustering, (d) Rosette Tracker [13], (e) active contour model [10]. Below (c)-(e), the corresponding values of the Modified Hausdorff Distance [5], and its approximation ($\widehat{\mathrm{MHD}}$), with respect to the ground truth in (b).

While the framework is generic and can accommodate any complex metric, in the following we use the Modified Hausdorff Distance (MHD) as a motivating example of a complex metric. This metric has proved useful when developing plant image segmentation algorithms. We observed that MHD relates well with the human intuition of the problem at hand, and is effective at penalizing segmentations containing errors that would jeopardize the accurate extraction of visual phenotypes (e.g., missing leaves, cut stems, holes inside leaves). Notably, MHD can be expressed in units of length (e.g., millimeter), easing the interpretation.

However, MHD is computationally expensive, and not feasible on sensing devices with limited storage and computational capacity. Hence, we propose to learn a (linear) model of the MHD from the data, based on (computationally) easier metrics as features.

In the following sections we begin with defining the MHD, the easier metrics, and proceed by presenting the linear model and feature selection algorithms.

A. Modified Hausdorff Distance

The Hausdorff distance measures the distance between two point sets as the greatest of all the distances from a point in one set to the closest point in the other set. It is defined as

$$f_{\mathrm{H}}(S, \hat{S}) = \max\{ \sup_{a \in S} \inf_{b \in \hat{S}} d(a, b), \sup_{b \in \hat{S}} \inf_{a \in S} d(a, b) \}, \quad (1)$$

where S and $\hat{S}$ are two non-empty subsets of a metric space and sup and inf denote supremum and infimum, respectively. Dubuisson et al. [5] proposed a Modified Hausdorff Distance and demonstrated its efficacy for real-world applications. Let $S = \{a_1, \ldots, a_N\}$ and $\hat{S} = \{b_1, \ldots, b_{\hat{N}}\}$ be sets of points. A directed distance between S and $\hat{S}$ can be defined as

$$d(S, \hat{S}) = \frac{1}{N} \sum_{a \in S} d(a, \hat{S}), \quad (2)$$

where $d(a, \hat{S}) = \min_{b \in \hat{S}} \|a - b\|_2$ is the distance between a point $a \in S$ and the set of points $\hat{S}$, and $\|a - b\|_2$ denotes the Euclidean distance between a and b. The two directed distances $d(S, \hat{S})$ and $d(\hat{S}, S)$ are then combined to define an undirected distance measure:

$$f_{\mathrm{MHD}}(d(S, \hat{S}), d(\hat{S}, S)) = \max\{d(S, \hat{S}), d(\hat{S}, S)\}. \quad (3)$$

The MHD is characterized by a large discriminatory power, and robustness to noise. However, the computational complexity of its exact calculation can be challenging when the segmentation masks (the point sets) contain millions of pixels.

B. Low-Complexity Segmentation Accuracy Metrics

Calculating the MHD involves a geometric search problem in a vast space, in order to reach a solution and output the distance between two segmentations. To lower the computational requirements, we considered the following set of metrics to model the desirable behavior of the MHD, and quantify segmentation accuracy.

1) *Statistical Error* (StatErr), is the sum of false positive and false negative errors, that is the number of misclassified pixel locations;
2) *Precision*, is the fraction of pixels in the segmentation $\hat{S}$ that matches the ground truth S;
3) *Recall*, is the fraction of ground-truth pixels contained in the segmentation $\hat{S}$;
4) *Rand Index* [4], measures the similarity between two segmentations as the frequency with which S and $\hat{S}$ agree on the classification of pairs of pixels;
5) *Variation of Information* (VoI) [14], measures the distance between two segmentations as a linear expression involving entropy of S and $\hat{S}$, and their mutual information;
6) *Jaccard Similarity Coefficient* (JSC) and *Dice Similarity Coefficient* (DSC) are used to quantify the spatial overlap between S and $\hat{S}$;
7) *Object-level Consistency Error* (OCE) [15], is based on the Jaccard Similarity Coefficient (i.e., it measures the spatial overlap between binary objects), but is more sensitive to over- and under-segmentation;
8) *Global Consistency Error* (GCE) [2], measures subset relationship between S and $\hat{S}$, based on local overlap;

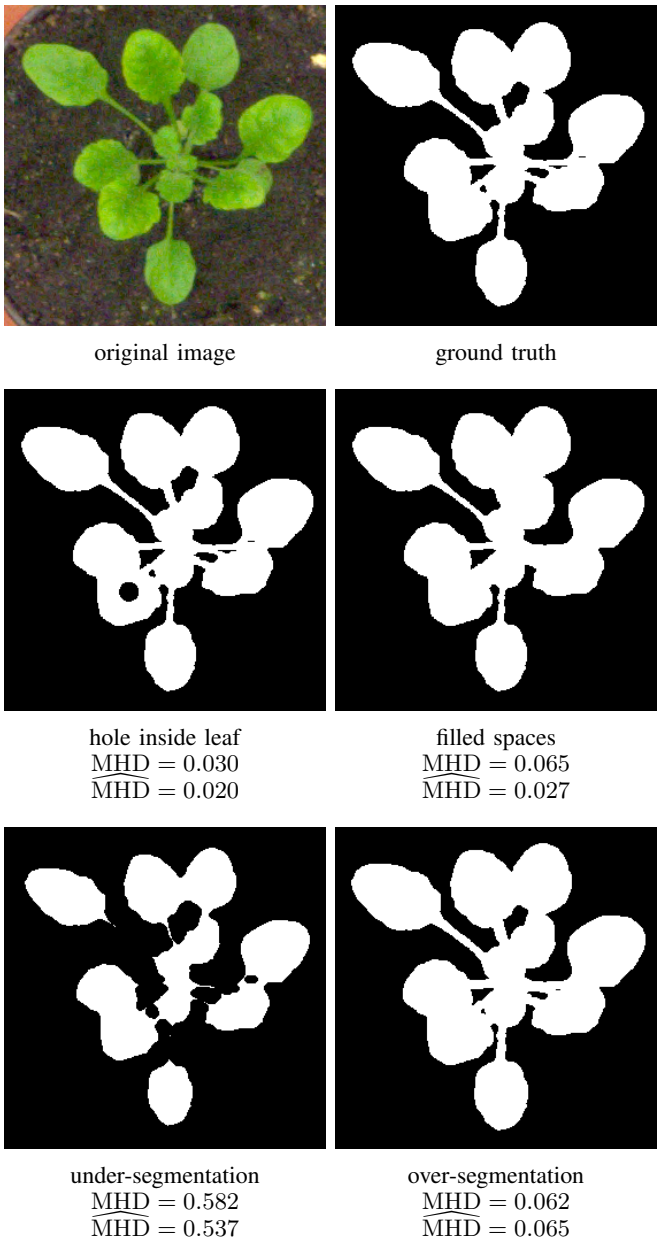


Fig. 2: Examples of synthetic errors in the segmentation of the original image (top left). Below each image, the corresponding values of exact Modified Hausdorff Distance [5], and its approximation ($\widehat{\overline{\text{MHD}}}$), with respect to the ground truth (top right).

9) Assuming a connected component labeling of the objects in the error mask (i.e., false positive and negative pixels), their number (CC-count), the error with maximum size (CC-max), their average size (CC-mean), and the standard deviation in size (CC-std). These metrics were inspired by application specific metrics, e.g., [6].

Calculating the above metrics is significantly less complex than the MHD. They do not require any optimization procedures and most of them can be calculated in a single pass of the segmentation masks. Only the last class requires connected component labeling that is slightly more complex, although efficient implementations exist. Additionally, several groups of such metrics share common intermediate steps for their computation, hence favoring optimized implementations. Our goal is to construct linear combinations of the presented surrogate metrics, to approximate the behavior of the MHD with simple pixel processes.

C. Modeling Complex Metrics

We used linear regression to learn from the data the relationship between the MHD and the set of surrogate metrics presented in Sec. II-B. We use a linear regression since it is computationally efficient; other approaches such as support vector regression [16] and random forest regression [17], while they have shown to have better performance in some applications, they can be significantly more demanding computationally and create models that are less interpretable. As we will show in the results section, linear regression adequately fits the problem at hand.

A linear regression model can be formulated in matrix notation as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}, \tag{4}$$

where $\boldsymbol{y} \in \mathbb{R}^n$ is called the response (i.e., the MHD values), $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ is the design matrix (i.e., the collection of surrogate metrics), and $\boldsymbol{\beta} \in \mathbb{R}^p$ contains the regression coefficients, which are obtained from the pseudo-inverse solution $\boldsymbol{\beta} = \boldsymbol{X} \backslash \boldsymbol{y}$. In order to preserve the non-negativity of the response, we calculate the predicted value $y_{\text{new}} \in \mathbb{R}$ for a new sample $\boldsymbol{x}_{\text{new}} \in \mathbb{R}^p$, as $y_{\text{new}} = zH(z)$, where $z = \boldsymbol{x}_{\text{new}}^T \boldsymbol{\beta}$, and H is the Heaviside step function.

III. Selection Algorithms

We also investigated the possibility to identify appropriate subsets of surrogate metrics (Sec. II-B) without loss of performance, using feature selection strategies based on sparse approximation algorithms. The sparse solution not only eliminates features (in our context the surrogate metrics) that are unnecessary, but also creates a more interpretable model. In this paper we analyze two popular approaches that originate from the sparse approximation field: convex optimization and greedy iterations. While sparse penalty terms have been previously and successfully proposed for regression (e.g., Lasso [18]), in our approach we explicitly impose the maximum allowed representation error, with respect to the model obtained with the full support of predictors.

A. ℓ_1 Optimization-Based Selection Algorithm

The first algorithmic approach involves the utilization of the ℓ_1 norm optimization problem, that has been used successfully in sparse representation and model reduction applications. We propose a variation of the Iterative Reweighted ℓ_1 (IRL1) minimization [19], that was shown to produce some of the best results in terms of model reduction capabilities, in the family of convex optimization methods.

IRL1 algorithm. Given response $\boldsymbol{y} \in \mathbb{R}^n$, design matrix $\boldsymbol{X} \in \mathbb{R}^{n \times p}$, maximum number of iterations M, and target

TABLE I: Example regression coefficients estimated on the entire dataset and corresponding R^2 statistic.

Model	β coefficients													R^2
	StatErr	Precision	Recall	Rand	VoI	JSC	DSC	OCE	GCE	CC-count	CC-max	CC-mean	CC-std	
No selection	-1.18	-2.15	-3.30	-2.63	-0.12	3.38	0.04	-0.05	-1.22	0.02	0.03	-0.02	0.01	0.902
IRL1	0	-1.85	-2.81	-0.78	0	2.81	0	-0.05	-0.77	0.01	0	0.01	0	0.895
Greedy	-0.90	-2.14	-3.27	-2.17	0	3.36	0	-0.05	-1.18	0	0	0	0.01	0.895

representation error $\epsilon \in \mathbb{R}$, return the sparse solution $\boldsymbol{\beta} \in \mathbb{R}^p$ such that $\|\boldsymbol{W}\boldsymbol{\beta}\|_1$ is minimized under the error constraint.

For iterations: $k = 1, \ldots, M$

1) If $k = 1$ then $\boldsymbol{W} = \boldsymbol{I}_p$, otherwise compute the new weights in the diagonal matrix $\boldsymbol{W}$:
$$W_{ii} = 1/(|\beta_i| + c),\ 0 < c \ll 1,\ i = 1, \ldots, p. \quad (5)$$
2) Solve the new optimization problem:
$$\boldsymbol{\beta} = \underset{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2 \leq \epsilon}{\operatorname{argmin}} ||\boldsymbol{W}\boldsymbol{\beta}||_1. \quad (6)$$

Notice that in the first iteration IRL1 solves the actual ℓ_1 optimization problem. The next iterations refine this result by adding weights to the problem, such that if a coefficient of the solution is large in absolute value, the weight is small. Conversely, if a coefficient is small in absolute value, the weight is large (we try to drive the coefficient exactly to zero). Due to this improvement, IRL1 consistently outperforms regular ℓ_1 optimization procedure [19].

B. Greedy Iterations-Based Selection Algorithm

The second optimization strategy in this paper uses optimized orthogonal greedy iterations to gradually eliminate design patterns that cause the smallest model error increase, while the representation error constraint is still satisfied.

Greedy algorithm. Given response $\boldsymbol{y} \in \mathbb{R}^n$, design matrix $\boldsymbol{X} \in \mathbb{R}^{n \times p}$, and target representation error $\epsilon \in \mathbb{R}$, return the sparse solution $\boldsymbol{\beta} \in \mathbb{R}^p$ and its support set S optimized under the error constraint.

- Given initial support $S^1 = \{1, \ldots, p\}$,
- For iterations: $k = 1, \ldots, p$
  1) Iterations: $j = 1, \ldots, p - k + 1$
     a) Eliminate from the support S^k the j^{th} column to get the active support $A = S^k - S^k_j$.
     b) Solve $\boldsymbol{X}_A\boldsymbol{\beta}_A = \boldsymbol{y}$.
     c) $e_j = \|\boldsymbol{y} - \boldsymbol{X}_A\boldsymbol{\beta}_A\|_2$.
  2) Find the minimum error increase $j_0 = \operatorname{argmin} \boldsymbol{e}$.
  3) If $\epsilon > e_{j_0}$ stop iterations and return support $S = S^k$, otherwise continue with $S^{k+1} = S^k - S^k_{j_0}$.
- Get final solution $\boldsymbol{\beta} = \boldsymbol{X} \backslash \boldsymbol{y}$ on S.

The greedy method reduces the support of the solution by removing, at each iteration, the design pattern that offers the lowest increase in the representation error, while still under the target imposed representation error. Also, the Greedy approach does not require any parameters, while IRL1 requires two parameters (M, c) to be selected a priori by the user.

IV. Results and Discussion

The aim of our evaluation is to demonstrate the appropriateness of a linear model for predicting the MHD, and assess the benefit of reducing the number of features employed, using selection algorithms. We first describe the plant segmentation dataset used and the experimental validation employed (cross validation), and conclude with results and their discussion.

A. Plant Segmentation Dataset

We conducted our experiments on a dataset of 222 segmentations of Arabidopsis plant images ($\sim$ 0.14 megapixel). The dataset was composed of 62 algorithmic results and 160 synthetic segmentations. Each sample of the dataset was accompanied by the corresponding ground-truth segmentation obtained manually, which was used to compute the accuracy metrics described in Sec. II.

The algorithmic results (see Fig. 1 for an example) were obtained using four different image segmentation methods previously proposed to segment plants in images:

- K-means clustering on pixel intensity values (commonly used in plant image segmentation [20]);
- Rosette Tracker software [13];
- vector-valued active contour model operating on pixel color intensities [10];
- vector-valued active contour model operating on color intensity and texture features [10].

The synthetic data (see Fig. 2) were obtained by introducing a variety of errors commonly encountered in this context (e.g., missing object parts) in the ground-truth segmentations. The synthetic data were generated in a computational manner, easing the process of obtaining training data to learn a model of a complex metric. While the algorithmic data contain mixed types of errors, the synthetic data represent specific classes (e.g., resulting from a biased segmentation), overall accounting for greater variability in both type and amount of errors.

Figures 1 and 2 show examples of plant segmentations, and the corresponding value of the MHD. The MHD penalizes more boundary displacements (e.g., cut leaf), and holes inside the object, while remaining robust to noise (e.g., scattered noise in the background, or noisy border lines). Also critical in the context of this paper, it penalizes under-segmentation that causes splits in the object (Fig. 2). In addition, the MHD exhibits a linear behavior with the amount of error introduced, rendering it suitable for linear regression.

The metrics presented here were implemented (using Matlab release 2012b) and their values were recorded. Prior to fitting any model, the metrics in the design matrix $\boldsymbol{X}$ were

TABLE II: Average selection rates for inclusion of the metrics in the model, obtained from K-fold cross validation.

Model	Selection Rate (%)												
	StatErr	Precision	Recall	Rand	VoI	JSC	DSC	OCE	GCE	CC-count	CC-max	CC-mean	CC-std
IRL1	20	100	100	80	20	100	20	100	70	80	20	40	20
Greedy	60	100	100	90	0	100	10	100	90	40	10	10	20

normalized to mean 0 and standard deviation 1, while the responses $\boldsymbol{y}$ were centered on the mean. Although this step is not necessary, it renders the regression coefficients comparable and eases discussion.

B. Experimental Settings

To demonstrate the generalization capability of the linear model, a K-fold cross validation strategy where $K = 10$ was employed, with random splits of the original data into training/testing sets. As a goodness-of-fit measure, the coefficient of determination $R^2 = 1-(SS_{\text{err}}/SS_{\text{tot}})$ was used, where SS_{err} is the sum of squared residuals, and SS_{tot} is the total sum of squares. At each round of the cross validation, the R^2 was calculated on both the training set, to evaluate the goodness of fit, and the testing set, to assess the capability of the model of generalizing to previously unknown data samples.

For the IRL1 algorithm $M = 5$, and $c = 10^{-7}$ in Eq. (5) were chosen empirically. For both Greedy and IRL1 the target error ϵ was chosen such that an increase in the representation error of up to a 4% of error of the full model was allowed. All experiments were executed on a machine equipped with Intel Core 2 Duo CPU E8200 2.10 GHz, 3 GB memory, and running 64-bit GNU/Linux.

C. Results

Table I shows an example of standardized regression coefficients of a linear model using all features learned on the entire dataset, to study the contribution of the surrogate metrics. The $R^2 = 0.9$ supports the validity of a linear model in representing the MHD, as 90% of the response variation in the data can be explained using a linear relationship. Precision, Recall, Rand, and JSC exhibit high coefficient magnitude, providing the largest contribution to the linear approximation of the MHD. Conversely, VoI, DSC, and OCE have small coefficients, as they convey similar information to JSC. In addition, the last group of metrics based on the connected component labeling (which can be less computationally efficient) has small coefficient magnitudes. Figures 1 and 2 report predicted values of the MHD for some example images (excluded from the training set), using the full linear model.

Similar conclusions can be obtained by optimizing which metrics are chosen using the selection algorithms presented previously (Table I). Allowing a 4% representation error, both IRL1 and Greedy selection strategies halved the number of predictors (by setting to 0 the coefficients of the discarded ones) with minimal reduction in R^2. These predictors (except CC-count, which is occasionally included) are underused also when considering the K-fold cross validation (Table II shows the percentage of times a feature was selected).

The outcome of the K-fold cross validation with respect to R^2 is shown in Table III. The full linear model obtained a cross-validated $R^2 \approx 0.9$ on the training set, showing a good fit of the regression. The capability of the linear model of generalizing to new unseen data is supported by an $R^2 = 0.835$ on the testing set. This observation holds also for the sparse approaches. The sparser solutions with a support of only 7 features (on average) were able to obtain comparable R^2 values in the testing and training set with respect to the full linear model. IRL1 incurs slightly lower prediction accuracy than the full model, while Greedy exhibits the best performance among the three. This result demonstrates that the shrinkage operation actually improved the ability of the model to generalize, and that the Greedy strategy selected better features than IRL1.

TABLE III: Cross-validated results of the approximate linear models, represented as mean (standard deviation).

Model	R^2		Support
	Training	Testing	
No selection	0.904 (0.012)	0.835 (0.128)	15.0 (0.0)
IRL1	0.898 (0.014)	0.814 (0.133)	7.7 (1.3)
Greedy	0.897 (0.013)	0.839 (0.123)	7.3 (1.2)

The proposed approximations reduced execution time considerably, by several orders of magnitude. On average, the exact computation of the MHD required 35 minutes, whereas all surrogate metrics were computed collectively in $\sim$ 0.05 seconds.

Finally, Fig. 3 demonstrates the relationship between support size and R^2 and the behavior of the selection algorithms, as obtained with cross validation and by varying the error ϵ. (Overall, the larger the ϵ allowed the more sparse the solution and the less features are used.) As previously, the Greedy strategy outperforms the ℓ_1 optimization in the choice of the predictors to include in the model, providing comparable or better prediction accuracy across the range of support (or equivalently the allowed error ϵ). The Greedy algorithm exhibits the global maximum with 7 predictors; i.e., discarding a subset of predictors improved the capacity of the model to generalize to new unseen data samples.

As was discussed in the introduction, remote sensing applications would benefit from the principle introduced in this paper. Recently, affordable remote sensors have been proposed for the collection and transmission of digital images of plants for phenotyping experiments [11], [12], to increase the availability of automated solutions for phenotype analysis. The sensors collect and compress the images in an application-aware fashion being aware of processing that will occur in a centralized (remote) location. Thus, the sensor must decide the compression parameters with regards to segmentation accuracy, where the uncompressed image is considered the ground truth. (The sensor has the capability to obtain a roughly good

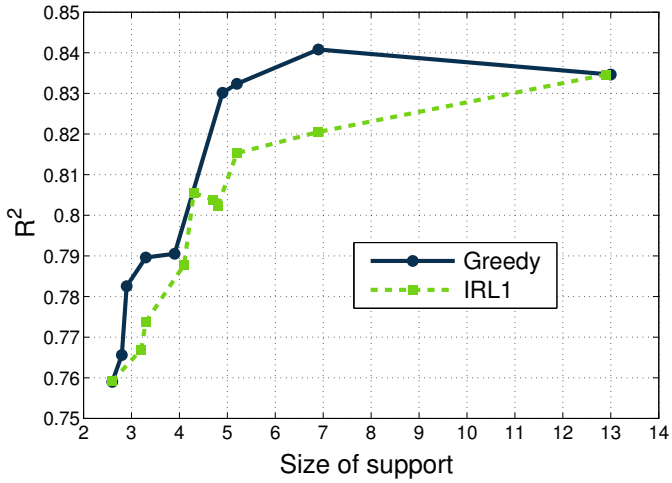


Fig. 3: Cross-validated prediction accuracy of the linear models, when varying the degree of sparsity enforced by the selection strategies.

segmentation using a low-complexity algorithm and feedback from a centralized unit [12].) Identifying a segmentation metric that could be included efficiently within a rate-distortion optimization framework is critical for optimal performance.

The linear approximation of the MHD proposed here would be an ideal candidate for such a scenario, since a linear combination of 7 simple to compute metrics can adequately approximate the MHD ($R^2 \approx 0.9$ in most cases). The optimization of the linear coefficients and which metrics to be used can be done at the centralized unit and communicated via feedback to the sensor. Thus, the sensor can adapt to the scene at hand and always use an efficient approximation of the MHD.

V. Conclusions

In this paper we model the behavior of a complex accuracy metric for image segmentation (e.g., the Modified Hausdorff Distance), using easier to compute metrics. We consider several of such metrics and use sparse representation approaches for feature selection to build linear regression models that could approximate the complex one. This paper presents for the first time the concept of using a collection of low-complexity metrics to approximate the behavior of a complex one. We apply and test the models on a dataset of plant image segmentations from phenotyping experiments. Our experimental results show that it is possible to closely estimate the MHD ($R^2 \approx 0.9$ in most cases) without necessarily calculating its exact value (a process involving a costly geometric search), using a linear combination of 7 simple to compute metrics. While we used here the MHD, the data-driven framework and concept presented in this paper can be deployed with other complex metrics as well, or even not in segmentation scenarios (e.g., in object matching) with appropriately identified features. This result is expected to pave the way towards the deployment of affordable sensors in distributed sensing applications, where the use of easier to compute segmentation metrics would enable the sensor to make informed decisions on the encoding and transmission of the acquired images.

Acknowledgment

This work was supported in part by a Marie Curie Action: "Reintegration Grant" (ref# 256534) of the EU's Seventh Framework Programme (FP7).

References

[1] S. R. Vantaram and E. Saber, "Survey of contemporary trends in color image segmentation," *J. Electron. Imag.*, vol. 21, no. 4, pp. 1–28, 2012.

[2] M. Martin, C. Fowlkes, D. Tal, and J. Malik, "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics," in *Proc. 8th Int. Conf. on Computer Vision*, vol. 2, 2001, pp. 416–423.

[3] V. Jain, B. Bollmann, M. Richardson, D. R. Berger, M. N. Helmstaedter, K. L. Briggman, W. Denk, J. B. Bowden, J. M. Mendenhall, W. C. Abraham, K. M. Harris, N. Kasthuri, K. J. Hayworth, R. Schalek, J. C. Tapia, J. W. Lichtman, and H. S. Seung, "Boundary Learning by Optimization with Topological Constraints," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 2488–2495.

[4] W. M. Rand, "Objective Criteria for the Evaluation of Clustering Methods," *J. Am. Statist. Assoc.*, vol. 66, no. 336, pp. 846–850, 1971.

[5] M. P. Dubuisson and A. K. Jain, "A Modified Hausdorff Distance for Object Matching," in *IAPR*, no. 1, 1994, pp. 566–568.

[6] L. E. Boucheron, N. R. Harvey, and B. S. Manjunath, "A Quantitative Object-Level Metric for Segmentation Performance and Its Application to Cell Nuclei," in *Adv. in Vis. Comput.*, 2007, vol. 4841, pp. 208–219.

[7] G. Gerig, M. Jomier, and M. Chakos, "Valmet: A New Validation Tool for Assessing and Improving 3D Object Segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2001, vol. 2208, pp. 516–523.

[8] K. O. Babalola, B. Patenaude, P. Aljabar, J. Schnabel, D. Kennedy, W. Crum, S. Smith, T. F. Cootes, M. Jenkinson, and D. Rueckert, "Comparison and Evaluation of Segmentation Techniques for Subcortical Structures in Brain MRI," in *MICCAI*, vol. 5241, 2008, pp. 409–416.

[9] M. J. Hossain, Dewan, K. Ahn, and O. Chae, "A Linear Time Algorithm of Computing Hausdorff Distance for Content-based Image Analysis," *Circ. Syst. Signal Process.*, vol. 31, no. 1, pp. 389–399, 2012.

[10] M. Minervini, M. M. Abdelsamea, and S. A. Tsaftaris, "Image-based plant phenotyping with incremental learning and active contours," *Ecological Informatics*, under review.

[11] S. A. Tsaftaris and C. Noutsos, "Plant Phenotyping with Low Cost Digital Cameras and Image Analytics," in *Information Technologies in Environmental Engineering*, 2009, ch. 18, pp. 238–251.

[12] M. Minervini and S. A. Tsaftaris, "Application-Aware Image Compression for Low Cost and Distributed Plant Phenotyping," in *Proc. 18th Int. Conf. on Digital Signal Processing (DSP)*, to appear (2013).

[13] J. De Vylder, F. Vandenbussche, Y. Hu, W. Philips, and D. Van Der Straeten, "Rosette tracker: an open source image analysis tool for automatic quantification of genotype effects," *Plant physiology*, vol. 160, no. 3, pp. 1149–1159, 2012.

[14] M. Meilă, "Comparing Clusterings by the Variation of Information," in *Computational Learning Theory*, vol. 2777, 2003, pp. 173–187.

[15] M. Polak, H. Zhang, and M. Pi, "An evaluation metric for image segmentation of multiple objects," *Image and Vision Computing*, vol. 27, no. 8, pp. 1223–1227, 2009.

[16] H. Drucker, C. J. C. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support Vector Regression Machines," in *Advances In Neural Information Processing Systems 9*, 1996, pp. 155–161.

[17] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[18] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *J. R. Stat. Soc. Series B Stat. Methodol.*, vol. 58, no. 1, pp. 267–288, 1996.

[19] E. Candes, M. B. Wakin, and S. Boyd, "Enhancing sparsity by reweighted l_1 minimization," *J. Fourier Analysis and App.*, vol. 14, no. 5, pp. 877–905, 2008.

[20] A. Hartmann, T. Czauderna, R. Hoffmann, N. Stein, and F. Schreiber, "HTPheno: An image analysis pipeline for high-throughput plant phenotyping," *BMC Bioinformatics*, vol. 12, no. 1, p. 148, 2011.