



HAL
open science

Segmentation using multi-thresholded Sobel images : Application to the separation of stuck pollen grains

Endrick Barnacin, Jimmy Nagau, Jean-Luc Henry, Jack Molinié

► To cite this version:

Endrick Barnacin, Jimmy Nagau, Jean-Luc Henry, Jack Molinié. Segmentation using multi-thresholded Sobel images : Application to the separation of stuck pollen grains. ICCV 2022: International Conference on Computational Vision, Jun 2022, Venice, Italy. hal-04114433

HAL Id: hal-04114433

<https://hal.science/hal-04114433>

Submitted on 1 Jun 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Segmentation using multi-thresholded Sobel images : Application to the separation of stuck pollen grains

Endrick Barnacin, Jean-Luc Henry, Jimmy Nagau, and Jack Molinié

Abstract—Being able to identify biological particles such as spores, viruses, or pollens is important for health care professionals, as it allows for appropriate therapeutic management of patients. Optical microscopy is a technology widely used for the analysis of these types of microorganisms, because compared to other types of microscopy, it is not expensive. The analysis of an optical microscope slide is a tedious and time consuming task when done manually. However, using machine learning and computer vision, this process can be automated. The first step of an automated microscope slide image analysis process is the segmentation. During this step the biological particles are localized and extracted. Very often, the use of an automatic thresholding method is sufficient to locate and extract the particles. However, in some cases, the particles are not extracted individually because they are stuck to other biological elements. In this paper, we propose a stuck particles separation method based on the use of the Sobel operator and thresholding. We illustrate it by applying it to the separation of 813 images of adjacent pollen grains. The method correctly separated 95.4% of these images.

Keywords—Image segmentation, Stuck particles separation, Sobel operator, Thresholding.

I. INTRODUCTION

In order to appropriately manage patients with infections or allergies, healthcare professionals need a means of identifying the biological particles (bacteria, microbes, viruses and others) that cause the disease. For example, the identification of a bacterium or virus allows the prescription of an appropriate antifungal or antiviral medication. The study of these microorganisms is usually performed using light microscopy because the cost of the equipment is not high compared to other types of microscopy. During the last decade, this technology has developed significantly and allows for capturing high resolution images of microscopic slides in high throughput. The processing time when these slide image analyses are conducted manually is extremely long and does not allow the analysis of slides in large quantities. However, thanks to computer processing, studies conducted in biomedical sciences are increasingly quantitative and precise. More specifically, machine learning and computer vision have facilitated and accentuated this last fact [10]. The first step of an automatic microscope slide image analysis process is the segmentation of the biological particles. Typically, when analyzing microscopic slides, the biologist stains the particles so that they stand out from the background. This is the reason why, very often, the use of an automatic thresholding method such as the Otsu

method [11] is sufficient to locate and extract the particles. However, at the end of the segmentation, the particles are not always extracted individually, because they are stuck to other biological elements. At this point, the use of a thresholding method is not sufficient, so a specialized method for detection and extraction of the stuck particles is needed. In this paper, we propose a particle separation method based on the use of the Sobel operator [12]. We illustrate it by applying it to the separation of adjacent pollens.

II. DATASET

In order to build the dataset, palynologists of the CIRAD based at Reunion Island collected honey samples from several hives. To extract the pollen grains, physicochemical treatments (dilution, centrifugation and acetolysis) [2] were performed on the samples. A small amount of the preparation (50 μ l) was pipetted and mounted between the slide and the coverslip. Then, a light microscope was used to capture 7216 images of pollen slides (Fig 1).

III. SEGMENTATION

Each pollen slide image was converted to HSV images [8] and the resulting saturation channel was thresholded using the Otsu algorithm to localize and segment pollen.

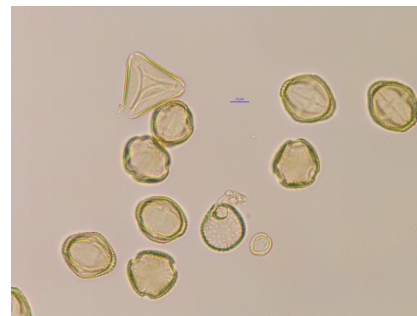


Fig. 1. Example of a pollen slide image.

Of 7216 slide images, 97.5% were correctly segmented. Approximately 2.5% of the images are not correctly segmented due to the transparency of some pollen. A database containing 1625 stuck particles, 12085 isolated grains, and 5468 residues was constructed.

IV. DETECTION OF STUCK PARTICLES

Before proceeding to the separation of the particles, it is necessary to detect them. This task can be performed using

Endrick Barnacin, Jean-Luc Henry and Jimmy Nagau is with the Department of Mathematic and Computer Science, Laboratoire de Mathématiques Informatique et Applications (LAMIA), Guadeloupe. Corresponding author e-mail: endrick.barnacin@univ-antilles.fr

All authors are with University of Antilles.

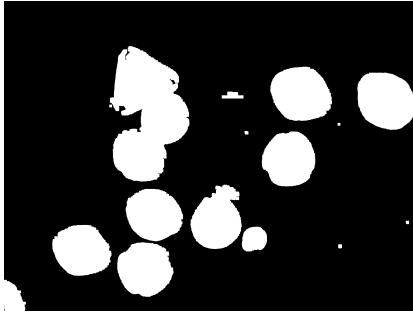


Fig. 2. Example of a pollen slide image thresholded with the Otsu method.

either classical machine learning or a deep learning approach. We chose to use the classical machine learning approach, because it requires less hardware resources than the deep learning approach. The classical machine learning approach consists of extracting features from the images and then using them as input to a classifier that will predict the types of the processed particles. We first used region features (Table I), such as area, circularity, or moments, because stuck particles, residues, and isolated pollen grains have different shapes and sizes. A pollen grain has a pseudo-triangular, elliptical or round shape compared to stuck particles whose shape cannot be described because it results from an agglomeration of pollen grains. In some cases, pollen grains and residues have a similar size and shape. In order to distinguish these particles, texture features have been added to the previously mentioned region features. We have elected to use local binary patterns [9] as they are fast to compute, are used in the literature on automatic pollen recognition [13], and have allowed us to obtain better performance than the well known co-occurrence matrix of Haralick [6]. Gabor filters [3] are an interesting alternative to the local binary pattern; however, due to a higher number of hyperparameters (7 against 2 for LBP), they require more computing power to find an optimal configuration.

The local binary patterns were introduced by Ojala et al. in 1994 [9]. They are constructed by applying the formula 1 to the pixels of the image.

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} 2^p \sigma(g_p - g_c) \quad (1)$$

x_c , y_c and g_c are respectively the coordinates and the intensity of the processed pixel. R and P are the radius and the number of neighboring pixels of a selected circle. g_p is the neighboring pixel being processed. σ is the Heaviside function. After applying formula 1 to all the pixels of the image, a new image is obtained. The histogram of the resulting image is used as features vector.

LBPs were calculated for radii between 2 and 40 with a step of 2 and neighbor numbers between 8 and 32 with a step of 2. After calculating the set of features, the most discriminating ones were selected using the Hall feature selection algorithm [5]. A multilayer perceptron [4] was then trained using the set

TABLE I
SUMMARY OF REGION FEATURES.

Shape Features	Descriptions
Area	Number of pixels representing pollen size
Perimeter	Number of pixels of pollen boundary
Compactness	$\frac{\sqrt{\frac{4}{\pi}}}{MaxDiameter}$
Circularity	$\frac{4*Area}{\pi*MaxDiameter^2}$
Solidity	$\frac{Area}{ConvexArea}$
Extent	$\frac{Area}{BoundingBoxArea}$
Central moments	$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I(x, y)$
Scale invariant moments	$V_{ij} = \frac{\mu_{ij}}{\mu_{00}^{(1+\frac{i+j}{2})}}$
Hu Moments	Seven binary moments [7]

TABLE II
RECOGNITION RATE OF THE PARTICLE TYPES.

Case	TR
Isolated pollen	98,7%
Stuck particles	94,9%
Residue	98,6%
All particles	98,4%

of selected features to predict the type of particles (isolated pollen, stuck pollens, or residues). Table II shows the recognition rates obtained for each particle type. As shown, 98.4% of the particles were correctly classified and 94.9% of the stuck particles were correctly detected. The less satisfactory score of the latter is explained by the fact that stuck particles occur 10 times less frequently in the database than the other types.

V. SEPARATION OF STUCK PARTICLES

The proposed procedure (Figure 4) starts by converting the image of stuck particles to grayscale and HSV. Then the hue, saturation and grayscale channels are extracted. A Sobel filter is then applied to each of the three images to highlight the contours of the pollens. The three filtered images are then thresholded with threshold values between 1 and 50. This interval maximizes the number of correct separations obtained. We proceeded in this way because it is not analytically possible to know which thresholds are optimal for pollen grains separation. Figure 3 illustrates this situation.

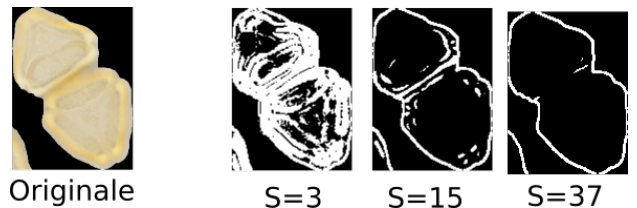


Fig. 3. Examples of thresholded Sobel transforms.

In the case of the first thresholding ($S=3$), the contours of the two pollens are poorly defined, because the thresholding result is noisy. In the case of $S=15$, the contours of each pollen appear distinctly and it is possible to isolate them. Finally in the last case, $S=37$, the threshold is too high; we obtain only the contour of the stuck particles. At the end of the thresholding operation, 150 binary images are obtained. The contours that are present in each binary image are extracted and used to create binary masks. These masks are then applied to the original image in order to isolate the different objects present in it. Among the obtained object images, there are pollens and sections of stuck particles. Also, a selection is necessary to retain only the pollen grains. In order to perform this selection, region features and local binary patterns were computed for each object. These features were used to train a RandomForest [1] which was used to sort the objects. This classifier was chosen because it is relatively fast to train and provides correct results in this application. Region attributes are used because pollens are generally round in shape, while the stuck particles sections have random shapes and sizes. Some sections of stuck particles have the shape and size of pollen grains. The local binary patterns were combined with the region features so that the classifier could distinguish them.

Removal of duplicate pollens

In the algorithm 1, the images of the stuck particles are thresholded several times. Thus, it is common that the same pollens are extracted several times. In order to obtain the correct number of pollen grains at the end of the treatment, the duplicate grains must be removed. We solved this problem by using classification. The bounding box of each pollen was extracted, and a bounding box comparison image database was built. Two by two, the bounding box of each pollen was extracted and placed in a single image. Figure 5 illustrates this point. In the comparison images, a color between blue and green was assigned to each of the two pollens bounding boxes. Thus, when the bounding rectangles intersect they become cyan. Comparison images with a lot of cyan were labeled as containing the same pollen (positive). Inversely, comparison images with few cyan were labeled as containing different pollens (negative). The areas of each zone (blue, green, cyan, black) of the comparison images obtained from the training database were used to train a RandomForest classifier. This classifier was used to determine the classes of the comparison images of the test base. As a result, redundant particles have been removed.

VI. EXPERIMENTATION AND RESULTS

The proposed procedure was evaluated using two criteria : the rate of correctly extracted contours and the rate of successful separations. The first is the percentage of stuck particles for which the method succeeded in extracting all the pollen grains that comprises them. The second is the percentage of stuck particles for which the method succeeded in extracting and selecting the total amount of pollens that compose them. The stuck particles base was divided into two equally sized

Algorithm 1: Procedure for separating stuck pollens.

Input: stuckParticlesImg, classifier
Output: pollenImagesList

```

1 hsv ← RGB2HSV(stuckParticlesImg);
2 channels ← split(hsv);
3 gray ← RGB2GRAY(stuckParticlesImg);
4 hue ← channels[0];
5 saturation ← channels[1];
6 sobelGray ← Sobel(gray);
7 sobelHue ← Sobel(hue);
8 sobelSaturation ← Sobel(saturation);
9 for (th=0; th < 50; th=th+1) do
10 | hueMask ← Thresholding( sobelHue, th);
11 | saturationMask ← Thresholding( sobelSaturation,
12 | th);
13 | grayMask ← Thresholding( sobelGray,th);
14 | contoursHue ← GetContours(hueMask);
15 | contoursSaturation ←
16 | GetContours(saturationMask);
17 | contoursGray ← GetContours(grayMask);
18 | contoursList←Concatenate(hueMask,
19 | saturationMask, grayMask)
20 | for (j=0; j < len(contoursList); j=j+1) do
21 | | mask ← EmptyImage ();
22 | | mask ← DrawContours(mask, contoursList[j]);
23 | | pollenImage ← ApplyMask(stuckParticlesImg,
24 | | mask);
25 | | Append(pollenImagesList, [pollenImage,
26 | | contoursList[j]]);
27 | end
28 end
29 end
30 for (i=0; i < len(pollenImagesList); i=i+1) do
31 | ImageFeatures ← ExtractFeatures
32 | (pollenImagesList[i] );
33 | classe ← classifier.predict (ImageFeatures)
34 | if classe = "IsNotPollenGrain" then
35 | | DeleteFromList(pollenImagesList,
36 | | pollenImagesList[i])
37 | end
38 end

```

Rate of extracted contours	Rate of successful separations
98,84%	95,40%

TABLE III
RESULTS OF THE STUCK PARTICLES SEPARATION PROCEDURE.

bases, a training and a test base. The stuck particles present in the training base were used to train the classifier responsible for the selection between the stuck particles sections and the pollen grains. By applying the separation procedure to the test base, the method succeeded in extracting 98.84% of the pollen grain contours that compose the stuck particles. Then in 95,4% of the cases it succeeded in selecting these contours. These results are presented in Table III.

There are 5 primary reasons for failed separations: Stuck

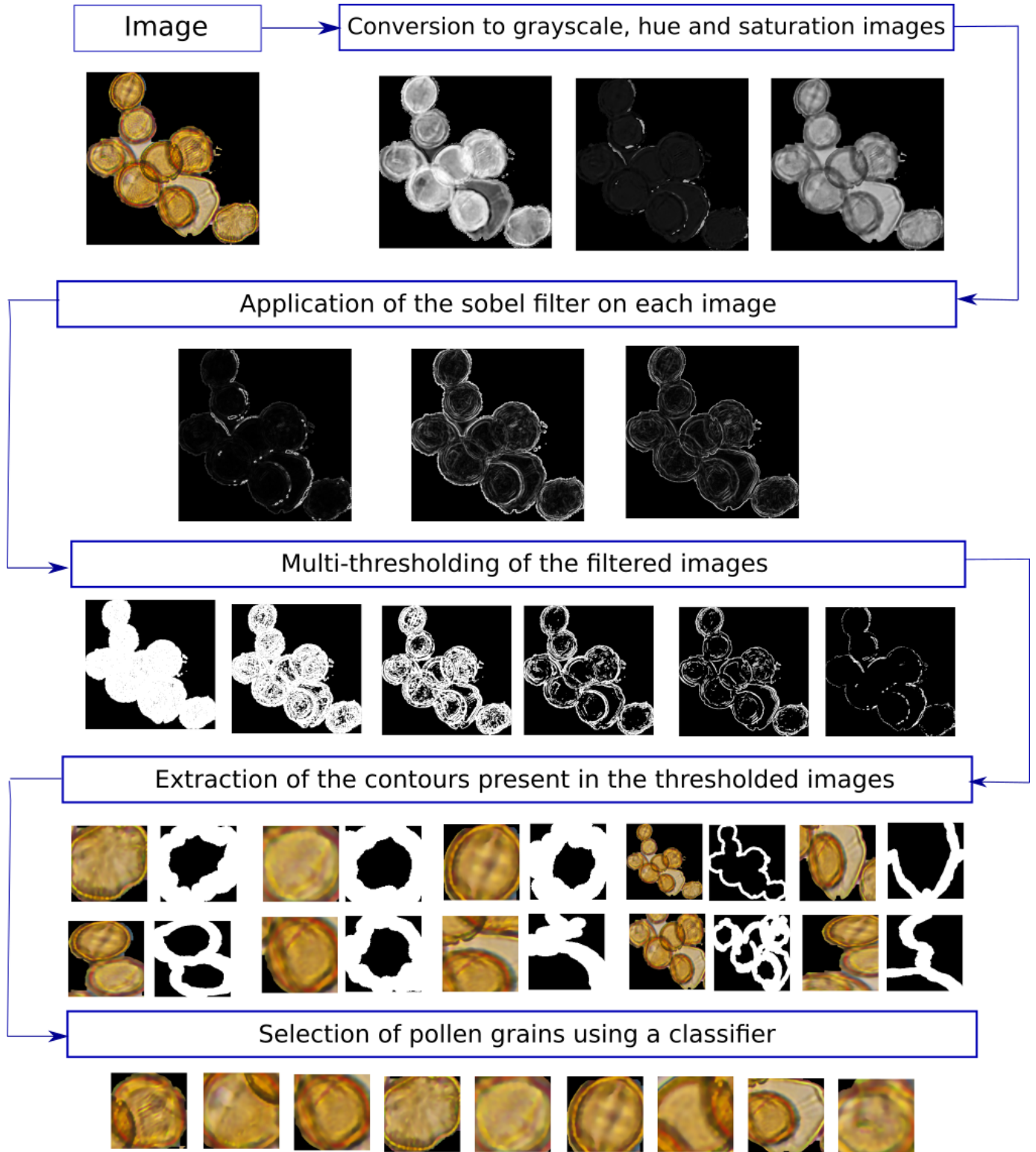


Fig. 4. Procedure for separating stuck pollens.

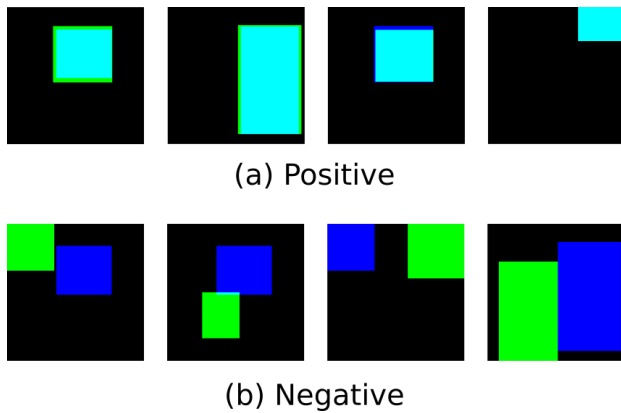


Fig. 5. Examples of comparison images.

particles for which the procedure failed to retrieve the exact contours (Fig 6a); Pollen species with a weak presence in the image database, i.e., species with less than 0.1% of the total number of images (Fig 6b); Deformed pollens (Fig 6c); Layered pollens (Fig 6d); And finally, images misclassification.

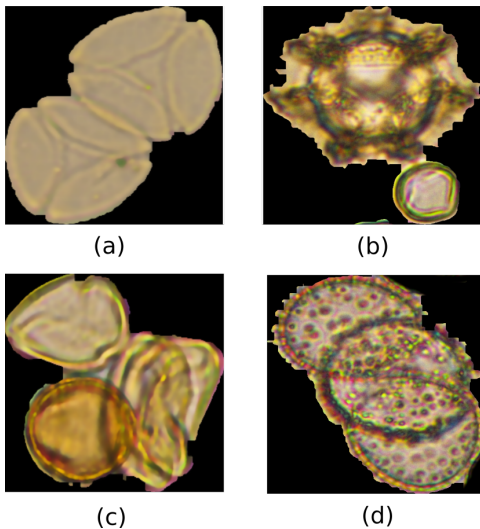


Fig. 6. Example of stuck particles for which the separation has failed.

VII. CONCLUSION

In this paper, we have introduced a method for separating adjacent microscopic particles. We first presented the sets of images on which we worked. Then, we described the procedures we used to extract pollen particles from these images and how the isolated pollen, stuck particles and residues were automatically separated. In a third step, we present the proposed method for the separation of stuck particles. This method consists of converting the image of the processed stuck particles into HSV space and applying the Sobel filter on channels of the resulting image. The filtered image is then thresholded several times, and the contours of the different thresholded images are extracted to obtain the particles. A selection is finally made using a classifier and shape and texture features to retain only isolated pollens. This method was applied to a database of 813 images of

adjacent pollens. It correctly separated 95.4% of these images.

A major drawback of this method is its execution time. Processing all of the images took approximately 5 hours using the 4 cores of a raspberry pi 4 (ARM Cortex-a72, 1.8 GHz). This corresponds to 0 m 22 s per image with an average size of 134 x 138. Our future work will focus on various ways to optimize this method such as the use of an automatic multiple or adaptive thresholding algorithm. It is also possible to define optimal thresholds by calculating them on the training base.

REFERENCES

- [1] Breiman, Leo. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32.
- [2] Erdtman, G. 1960. "The Acetolysis Method—a Revised Description."
- [3] Fogel, I., and D. Sagi. 1989. "Gabor Filters as Texture Discriminator." *Biological Cybernetics* 61 (2): 103–13.
- [4] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016. "Deep Learning". MIT Press.
- [5] Hall, Mark Andrew. 1999. "Correlation-based feature selection for machine learning."
- [6] Haralick, Robert M., K. Shanmugam, and Its 'hak Dinstein. 1973. "Textural Features for Image Classification." *IEEE Transactions on Systems, Man, and Cybernetics SMC-3* (6): 610–21.
- [7] Hu, Ming-Kuei. 1962. "Visual Pattern Recognition by Moment Invariants." *IRE Transactions on Information Theory* 8 (2): 179–87.
- [8] Joblove, George H., and Donald Greenberg. 1978. "Color Spaces for Computer Graphics." *SIGGRAPH Comput. Graph.* 12 (3): 20–25.
- [9] Ojala, T., M. Pietikainen, and D. Harwood. 1994. "Performance Evaluation of Texture Measures with Classification Based on Kullback Discrimination of Distributions." In *Proceedings of 12th International Conference on Pattern Recognition*, 1:582–85 vol.1.
- [10] Orlov, Nikita V., Josiah Lohse Johnston, Tomasz Macura, and Ilya Goldberg. 2007. "Computer Vision for Microscopy Applications." In *Vision Systems: Segmentation and Pattern Recognition*. unknown.
- [11] Otsu, Nobuyuki. 1979. "A Threshold Selection Method from Gray-Level Histograms." *IEEE Transactions on Systems, Man, and Cybernetics* 9 (1): 62–66.
- [12] Sobel, Irwin. 2014. "An Isotropic 3x3 Image Gradient Operator." February.
- [13] Vega, Gildardo Lozano. 2015. "Image-Based Detection and Classification of Allergenic Pollen." June. <http://dx.doi.org/>.