

CLASSIFICATION OF PROTEIN CRYSTALLIZATION IMAGERY

Xiaoqing Zhu¹, Shaohua Sun¹, Samuel E. Cheng¹ and Marshall Bern²

¹Department of Electrical Engineering, Stanford University, CA, USA

²Palo Alto Research Center, CA, USA

ABSTRACT

We investigate automatic classification of protein crystallization imagery, and evaluate the performance of several modern mathematical tools when applied to the problem. For feature extraction, we try a combination of geometric and texture features; for classification algorithms, the support vector machine (SVM) is compared with an automatic decision-tree classifier.

Experimental results from 520 images are presented for the binary classification problem: separating successful trials from failed attempts. The best false positive and false negative rates are at 14.6% and 9.6% respectively, achieved by feeding both sets of features to the decision-tree classifier with boosting.

1. INTRODUCTION

The most popular technology of our age for understanding the 3-D structure of protein molecules is X-ray crystallography, which analyzes the diffraction patterns of protein crystals. A major challenge, however, is the procedure of protein crystallization, as the outcome is very sensitive to experimental conditions such as chemical solution, temperature and air pressure. In order to successfully produce protein crystals suitable for X-ray diffraction, hundreds of thousands of trials are typically needed.

Modern robotic crystallization systems can perform more than 10,000 trials per day, each with a unique set of chemical conditions and periodically recorded outcomes via digital photography. This allows a wider range of parameters to be tested efficiently. Fast automatic evaluation of the outcome images, however, remains a challenge [1]. The key issue is the classification and ranking of crystallization results. This is a machine learning problem, usually with human-annotated images as training data. For simplicity, we now only consider binary classification, which distinguishes between successful and unsuccessful trials. The ultimate goal is to automatically predict trends from previous experimental outcomes and to guide future parameter configurations.

Sample images of successful and failed trials are shown in Fig. 1 and 2. Note that the appearances of precipitates

and crystals vary significantly under different conditions for different proteins. Moreover, the boundary between crystal and precipitate images is somewhat blurred when the outcome contains both precipitates and tiny crystals.

We note that current research in this area is still in the trial-and-error stage. The choices of features are mainly based on heuristics, and the understanding of how well each feature can perform is generally lacking. Either an off-the-shelf classifier is used, or an ad hoc solution is devised with hand-tuned thresholds. In addition, current classification algorithms typically achieve false negative and false positive rates around 15-20% [1]-[6], which is not good enough to support automatic data analysis. A vast area of algorithms for feature extraction and classification remains yet to be explored.

In this work, we investigate several modern mathematical tools as applied to crystallization imagery classification. For feature extraction, we combine geometric features characterizing local pixel gradients with texture features derived from the gray-level-cooccurrence-matrix (GLCM). For classification, we experiment with the support vector machine (SVM) and an automatic decision-tree algorithm.

The remaining part of the paper is organized as follows. A brief survey on existing methods for the classification of crystallization imagery is provided in Section 2. In Section 3, we explain the overall structure of the system, followed by detailed discussions of feature extraction in Section 4 and classification algorithms in Section 5. Experimental results are presented in Section 6.

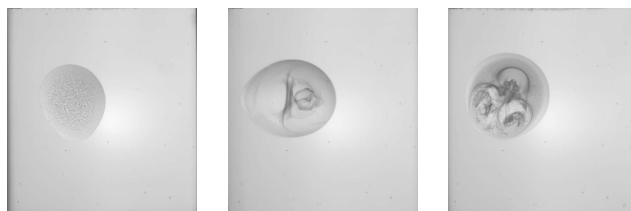


Fig. 1. Sample images of different precipitate images.

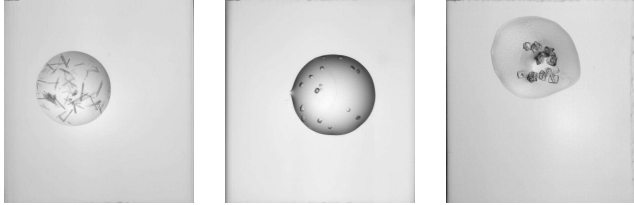


Fig. 2. Sample images of different crystal images.

2. RELATED WORK

Early work by Zuk and Ward [3] uses Hough transform to detect straight edges for crystals, but does not attempt to classify the images. A custom-built image acquisition and image processing system is reported in [2], where Jurisica et al. detect drop boundaries by fitting a conic curve and classify images based on spectral analysis. No specific error rates are reported. Instead, the correlation between extracted features and crystallization results is demonstrated by example.

In [1], Wilson uses the Sobel edge detector to locate drop boundaries and detects objects with high circularity; the images are then classified into three categories based on features of edge pixels. The reported accuracy rate is around 75%. In [4], Spraggon et al. apply the Canny edge detector and circle fitting to drop boundary detection and use a self-organizing neural network for classification, achieving 25% for both false negative and false positive rates. They use both GLCM-based texture features and geometric features related to straight lines. More recently, a probabilistic graphical model is used for drop boundary detection and for classification. Features are obtained from correlation filters and Radon transform (similar to Hough transform, with polar coordinate parameterization) [5]. A balanced error rate of 15% is achieved for binary classification: crystal-positives versus crystal-negatives.

The best result so far is reported in [6], where Bern et al. propose a line tracking algorithm for drop boundary detection and a decision-tree classifier with hand-crafted thresholds operating on geometric features. The classification achieves a false negative rate of 12% and a false positive rate of 14%. However, it is mentioned that the current feature-detection algorithms fail to capture the difference between swirly precipitates due to convective currents and micro-crystals, and that new features representing global characteristics of the images are needed.

3. SYSTEM OVERVIEW

The overall structure of the crystallization imagery classification system is illustrated in Fig. 3. Each image is pre-processed by a drop boundary detection algorithm, which provides a mask for pixels inside the drop area. We use the

line-tracking algorithm in [6] for this purpose. After that, feature vectors are extracted from the pixels within the drop boundary. They can be either geometric features or texture features, or a combination of both. These feature vectors are fed into a classifier, which is trained off-line using human annotated images. The classifiers in use are the C5.0 automatic decision tree provided in [7] and the publicly available SVM-Light [8]. To gain a better understanding of the feature vectors, automatic feature selection is also performed, either according to the statistics of the feature vectors, or during the initial stage of classification.

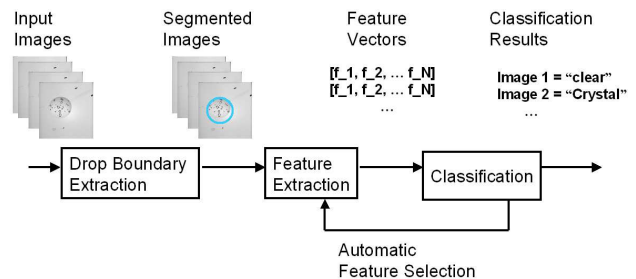


Fig. 3. System diagram

4. FEATURE EXTRACTION

As discussed in Section 1 and 2, it is hard to extract features that distinguish between precipitates and crystals. On the local scale, while features based on local pixel gradients can capture short edges of the crystals and generally work well, they may get confused by swirly precipitates as shown in Fig.1. On the global scale, both crystals and precipitates may contain significant high frequency components, therefore spectral analysis may not work very well either. Alternatively, we observe that images containing crystals tend to have a crispy texture whereas those dominated by precipitates appear to be more fluffy. Therefore, a combination of geometric and texture features is expected to have the benefit of both.

4.1. Geometric Features

Features that have been tried in previous researches are mostly related to the geometric characteristics of a local region of the image (e.g., straight-line detection from Hough transform, conic curve fitting or line tracking). We use the features in [6] for classification and for comparison with the texture features. A brief description for each feature is listed below; interested readers are referred to the original paper for further details:

- F1** Average directional gradient¹ within the best block².
- F2** Average directional gradient of all pixels inside the drop.
- F3** Peak value of the Hough transform line detection.
- F4** F3 normalized by the directional gradient of background pixels within the drop boundary.
- F5** A em curve score evaluating the straightness and smoothness of the curves formed by the selected pixels.
- F6** F5 normalized by F1.
- F7** Standard deviation of the grey levels in the best block.

4.2. Texture Features

We base our texture feature extraction on the gray-level-cooccurrence-matrix (GLCM) defined in [9]. More specifically, a GLCM gives a joint histogram of the quantized gray-level value pairs of two image pixels bearing a certain spatial relationship. The most widely used GLCMs are the ones for two neighboring pixels, aligned horizontally, vertically, along the diagonal or the counter-diagonal directions. A collection of different properties such as contrast, correlation and entropy are calculated for each GLCM. To make the features orientation-invariant, the average and variance of these features, over GLCMs of different orientations, are used as the final texture features. For each image, four GLCMs are computed from the pixels within the drop boundary, corresponding to neighboring pixels aligned horizontally, vertically, diagonally and counter-diagonally.

For each GLCM, we compute 12 out of the 14 features suggested in the original paper. The average and variance of these features over the 4 GLCMs are then calculated. Therefore, we obtain a 24-dimensional feature vector for each image. The texture features are listed below:

- F1** Measuring the homogeneity of the image.
- F2** Measuring the amount of local variations.
- F3** Measuring the gray-tone linear-dependencies.
- F4** Moment calculated from the difference in gray-tone values of the adjacent pixels.
- F5** Expectation calculated from the distribution of the sum of the gray-level values for the two pixels.
- F6** Variance calculated from the distribution of the sum of the gray-level values for the two pixels.
- F7** Entropy calculated from the distribution of the sum of the gray-level values for the two pixels.
- F8** Entropy calculated from GLCM; textures with more irregular patterns receive higher scores.
- F9** Variance calculated from the distribution of the difference of the gray-level values for the two pixels.

¹Directional gradient refers to the component along gradient direction at each pixel discounting the effect of drop boundaries.

²The best block refers to a 25×25 block containing the most image pixels with high gradient values.

- F10** Entropy calculated from the distribution of the difference of the gray-level values for the two pixels.
- F11** Similar to F3, based on entropy calculations.
- F12** An alternative formula to F11, also based on entropy calculations.

Interested readers are referred to [9] for the exact formulas. Although the features are calculated to capture some statistical property of the GLCM, their exact meanings are rather vague. They are, nevertheless, included in the initial classification stage to ensure preservation of enough information from the original image data. Their effectiveness can then be automatically examined during the classification stage.

5. CLASSIFICATION ALGORITHM

5.1. Automatic Decision Tree

The C5.0 classifier is a publicly available commercial software for data mining [7]. It automatically extracts classification rules in the form of a decision tree from given training data. It also supports adaptive boosting, based on the idea from Freund and Schapire [10], where more than one classifiers are generated, and the final classification result is voted by all classifiers. In addition, C5.0 provides an option called “winnowing”, which pre-selects the more differentiating features for designing the decision tree. This option is used for automatic feature selection.

5.2. Support Vector Machine

The SVM method hinges to two mathematical operations: nonlinear transform of an input vector into a high-dimensional feature space and construction of an optimal hyperplane for separating the transformed feature vectors [11]. In the first operation, the original feature vectors are mapped into a higher dimensional space where the two classes may be linearly separable via an inner-product kernel. Most commonly used kernels are the linear and radial basis function (RBF) kernels. In the second operation, an optimal separating hyperplane is constructed to maximize the margin of separation between positive and negative samples. We use the implementation of SVM-Light [8] for our experiments and try with both the linear and RBF kernels.

6. EXPERIMENTAL RESULTS

Experiments are performed over a dataset of 520 human annotated images at the website of the Joint Center for Structure Genomics (JCSG) [12]. The dataset consists of 130 samples in each of four categories defined by JCSG. For binary classification, the first two categories are labeled as “Failure”, and the last two as “Success”.

We test on the geometric features, the texture features

and the combination of both. The dimensions of the feature vectors are 11, 24 and 35, respectively. Public-domain implementations of the SVM algorithm (SVM-Light [8]) and the automatic decision-tree algorithm (C5.0 [7]) are used.

The 10-fold cross validation method is used for evaluation. The entire dataset is randomly divided into 10 disjoint sets and the result is averaged over 10 folds of experiments using 9 sets for training and the remaining one for testing. False-positive and false-negative rates are calculated with respect to human annotation results.

The classification results are summarized in Table 1. For the C5.0 classifier, the boosting option is also tested for improved performance. For SVM-Light, both linear and RBF kernels are tested. Only results from the RBF kernels are presented, since they are superior to those from the linear kernel. In general, the texture feature outperforms the geometric features. The C5.0 classifier with boosting gives the best results using both feature sets: false positive at 14.6% and false negative at 9.6%.

As mentioned before, the actual meaning of many of the feature vectors may be quite vague, and their effectiveness for classification need to be further evaluated. The winnowing option in the C5.0 classifier pre-selects the several important feature vectors and only uses these features to build a decision tree, with very small sacrifice in classification performance. Among the geometric features, the ones selected by the classifiers are F2, F4, F5 and F7. The selected texture features are F2, F3, F5 and F9.

7. CONCLUSION

We investigate the effectiveness of several new mathematical tools, including texture feature extraction and the SVM classifier, for the classification of protein crystallization imagery. The performance is compared and combined with existing algorithms such as geometric feature extraction and automatic decision-tree classification.

Experimental results from 520 human annotated images

Table 1. Experimental results using three different classifier configurations: SVM-Light, regular C5.0 and C5.0 with boosting. F.N. refers to the false negative rates and F.P. refers to the false positive rate.

		Texture	Geometric	Combined
SVM	F. N.	19.62%	36.54%	17.69%
	F. P.	30.77%	30.77%	29.23%
C5.0(A)	F. N.	19.2%	31.2%	15.4%
	F. P.	14.2%	12.3%	14.6%
C5.0(B)	F. N.	17.7%	28.5%	14.6%
	F. P.	12.7%	10.4%	9.6%

validate the benefits of texture features, and show that the SVM classifier yields comparable performance to other classifiers reported in literature. For binary classification between failed and successful protein crystallization trails, the best results are false positive rate at 9.6% and false negative rate at 14.6%, achieved by the C5.0 automatic decision tree classifier with boosting using both geometric and texture features.

8. ACKNOWLEDGMENTS

The authors would like to thank Professors Sabastian Thrun, Gary Bradsky and Dr. Daniel Russakoff at Stanford University for helpful discussions.

9. REFERENCES

- [1] J. Wilson, "Towards the automated evaluation of crystallization trials," *Acta Crystallographica D*, vol. 58, 2002.
- [2] I. Jurisica et al., "Intelligent decision support for protein crystal growth," *IBM Systems Journal*, 2001.
- [3] W. M. Zuk and K. B. Ward, "Methods of analysis of protein crystal images," *Journal of Crystal Growth*, vol. 110, 1991.
- [4] G. Spraggon, A. Kreusch S. A. Lesley, and J. P. Priestle, "Computational analysis of crystallization trials," *Acta Crystallographica D*, vol. 58, November 2002.
- [5] Cumbaa et al., "Automatic classification of sub-microlitre protein-crystallization trials in 1536-well plate," November 2002.
- [6] Marshall Bern et al., "Automatic classification of protein crystallization images using a line tracking algorithm," *Acta Crystallographica D*, 2003.
- [7] "C5.0," <http://www.rulequest.com/see5-info.html>.
- [8] T. Joachims, "Making large-Scale SVM Learning Practical," *Advances in Kernel Methods - Support Vector Learning*, MIT-Press, 1999.
- [9] R. Haralick, K. Shanmugam, and I. Dinstein, "Textural Features for Image Classification," *IEEE Transactions on Systems Man Cybernetics(SMC-3)*, 1973.
- [10] Y.Freund and R.E. Schapire, "A short introduction to boosting," *Journal of Japanese Society for Artificial Intelligence*, vol. 14, 1999.
- [11] Simon Haykin, "Neural networks: a comprehensive foundation," *Prentice Hall*, 1999.
- [12] "Joint Center for Structural Genomics," <http://www.jcsg.org>.