

EECS 545 F07 Project Final Report

**Chemical Structure Image Extraction from Scientific Literature
using Support Vector Machine (SVM)**

Jungkap Park, Yoojin Choi, Alexander Min, and Wonseok Huh

1. Motivation

In chemical/biological research, the chemical structure database is widely used to find the detailed information of molecules of interest. However, current chemical structure database does not provide link to the relevant literature which may contain more up-to-date information. Hence, by annotating each molecule in the database with one or more relevant links to the scientific literature, the database would be a more useful resource to bio/chemical research scientists [1]. In general, the chemical structure diagram in a scientific article can be considered as keywords representing the main content of the article. Therefore, it is necessary to develop an automated system for extracting and recognizing chemical structure diagrams. In this project, we plan to develop an efficient method to extract chemical structure diagrams using Support Vector Machine (SVM) as a first step towards constructing such system.

2. Problem Statement

In this project, the main objective is to devise a classification algorithm which accurately identifies chemical structure diagrams. Chemical structure diagrams have some unique topological characteristics such as lines, hexagons, and pentagons while other images do not share those characteristics as shown in Fig. 1. Based on this observation, we define a feature vector where each element represents the unique characteristics of chemical structure diagrams. Then, thus defined feature vector may include following feature information.

- Color histogram
- Connectivity histogram
- Text information (i.e., atom symbols)
- Line information (via Hough Transform (HT))
- Existence of hexagon/pentagon structure (via Generalized Hough Transform (GHT))

The detailed implementation and preliminary results will be discussed in the next section.

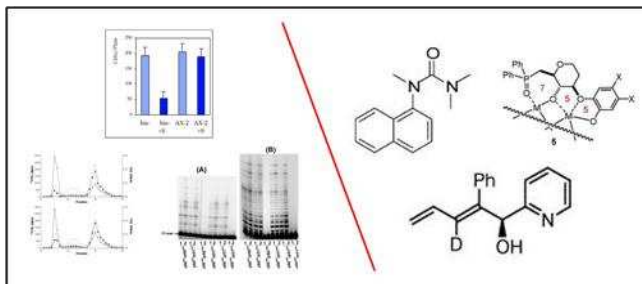


Fig. 1. Classification of chemical structure diagrams

3. Implementation

In this section, implementation methods and some simulation results for feature selection/extraction are presented. For illustration purpose, we use sample images as shown in Fig. 2. for chemical and non-chemical diagrams.

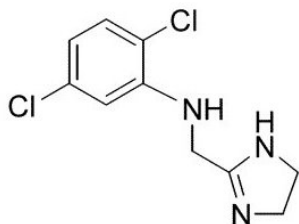


Fig. 2. (a) Chemical image

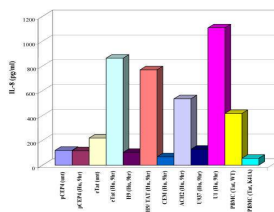


Fig. 2. (b) Non-chemical image

(1) Color histogram

Chemical structure diagrams are usually black-and-white images, and large portion of the image is white space. On the other hand, non-chemical images may contain various colors and intensities. Therefore, extracting the color histogram is not only an important feature to distinguish the chemical images but also an efficient method since extracting the color histogram is very simple and fast.

Implementation

We use hue-saturation-value (HSV) space to represent the color component of each pixel [2]. In color histogram, the number of bins for each color component (i.e., H, S, and V) is fixed to 16 so that the dimension of the color histogram is 16^3 . Thus obtained histogram will be used as a feature element in SVM.

Results

An example of color histogram (log scale) for chemical and non-chemical are shown in Fig. 3. As you can see in the figure, the color histogram of the chemical image is periodic while the histogram of non-chemical image is more distributed.

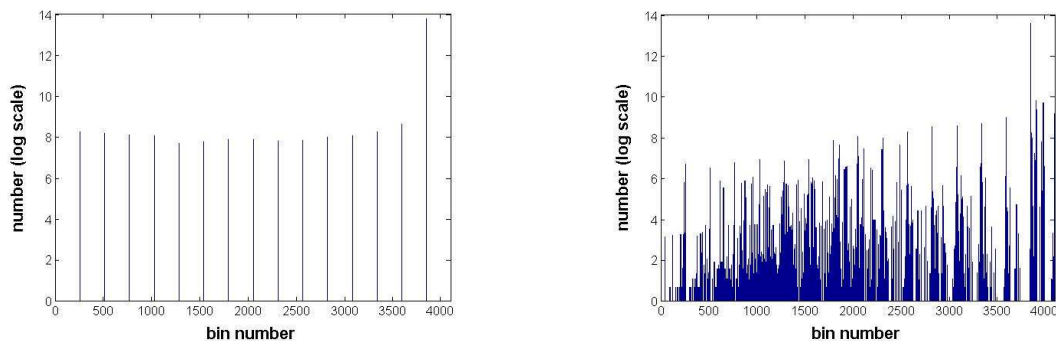


Fig. 3. (a) Color histogram of chemical image (b) Color histogram of non-chemical image

(2) Connectivity histogram

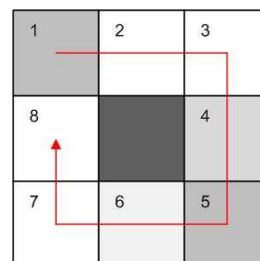
In most cases, a large portion of chemical structure images are white spaces in comparison to non-chemical images. Based on this simple observation, we introduce *connectivity* of an image as a feature element for SVM. Connectivity is defined for each pixel and represents the number of neighboring pixels those are not an empty (i.e., white) pixel.

Implementation

Before we check the connectivity of each pixel, we convert the RGB image to gray-scale image by using the following equation:

$$I = 0.2999 * R + 0.5870 * G + 0.1140 * B$$

Then, for each pixel, we investigate the intensity (I) of eight neighboring pixels (see the right figure). Each neighboring pixel is considered as having connectivity if the intensity of the pixel is less than 255 (i.e., not white). For example, the intensity of the center pixel of the right figure is 4.



Results

The normalized connectivity histograms for chemical and non-chemical images are shown in Fig. 4. For chemical image, the ratio of pixels with connectivity 0 is dominant while other connectivity components are almost zero. On the other hand, it is shown that non-chemical image has significant fraction of connectivity 8 components which means that a large portion of the image is non white pixels as expected.

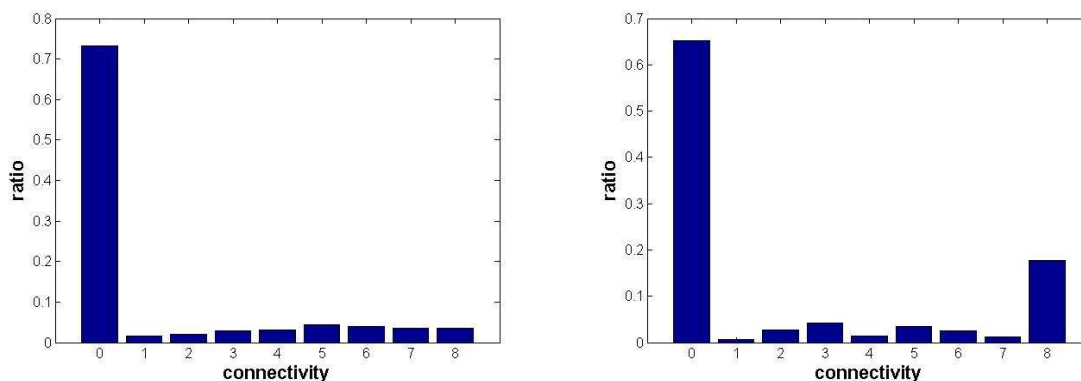


Fig. 4 (a) Connectivity hist. of chemical image (b) Connectivity hist. of non-chemical image

(3) Text information

Chemical images mostly contain short characters (i.e., atom symbols) in the chemical structures. For example, certain atom symbols in the periodic table such as 'C', 'O', and 'N' appear frequently in the chemical images as a single character or in a combined form (e.g., 'NH' or 'CO') as can be seen in Fig 2(a). Based on this simple observation, we exploit the character information in the input image in classifying chemical structures.

Implementation

We implemented an algorithm that accurately identifies the embedded characters in the given image for chemical diagram classification. The one of the main challenges in character recognition is to localize the characters. To solve this problem, we first identify the connected components in the image, based on pixel connectivity. Then we filter them based on the typical ratio (i.e., width/height) of a character. These component blocks are passed to the ‘GOOCR’ [9], which is an open source character recognition program. The ‘GOOCR’ returns the character contained in the input block components with high probability. We use thus-obtained character information as a feature element for SVM to classify chemical images. For example, if a large portion of identified characters belong to the predefined chemical symbol table (e.g., ‘C’, ‘OH’, ‘CO’, etc), the image is highly likely to be a chemical image.

Likelihood Analysis

To exploit the character information for chemical structure classification, we need a criterion to determine whether the identified characters are chemical symbols or general text. Therefore, we calculate the likelihood of chemical symbols based on predefined, frequently-used chemical symbol table, which is built by us based on the chemical structure drawing software, *JChem* [10]. We denote by $Table_chem = \{T^i\}_{i=1}^{864}$ the chemical symbol table of 864 frequently used chemical symbols. The detailed procedure of the calculation of the likelihood is as follows. For brevity, we omit the superscript if there is no confusion. Given the identified word (or string) ‘S’, ‘S’ = ‘S₁ S₂ ... S_n’, we calculate P(S|T) for each T in the chemical symbol table, the probability of recognizing ‘S’ assuming the true word is ‘T’, ‘T’ = ‘T₁T₂ ... T_m’. Here, the S_j and T_k represent each character. Let P_{S_jT_k} = the probability of recognizing T_k as S_j, P₂ = the probability of recognizing a separate part as a separate part, and P₃ = the probability of recognizing a connected part as a connected part. Our empirical results show that P₂ ≈ 0.95 and P₃ ≈ 1, respectively. Since P₃ ≈ 1, in almost all the cases, n, the length of S, is smaller than m, the length of C. Thus, we assume n < m. Then, we can calculate P(S|T) as follows:

$$P(S|T)$$

$$= \sum_{j=2}^{m-n} (P_{S_1 T_j} \dots P_{S_n T_{j+n-1}}) (1 - P_2)^2 P_2^{(n-1)} + \sum_{j=1, N-n+1} (P_{S_1 T_j} \dots P_{S_n T_{j+n-1}}) (1 - P_2) P_3 P_2^{(n-1)}.$$

Since the exact frequency of each chemical symbols is not known *a priori*, we assume that P(T) is equi-probable for all $T \in Table_chem$ and $\sum_{T \in Table_chem} P(T) = 1$. Then, we calculate the maximum-likelihood (ML) of all the words found in a given image. The ML can be expressed as:

$$ML = \max_{T \in Table_chem} P(S|T)P(T)$$

Once we calculate the likelihood for all the identified words, we use the **mean** and **variance** of them as our SVM parameter. We expect that the mean of chemical images are higher than that of non-chemical images, while variances of chemical images are small than that of non-chemical images. The simulation results verify our expectation.

Results

An example of text extraction process of a chemical image is shown in Fig. 5. Fig. 5(a) shows the original chemical structure. The blue boxes in Fig. 5(b) represent the identified components which satisfy the typical ratio of a character. Then the 'GOOCR' returns the exact character contained in the components as shown in Fig. 5(c).

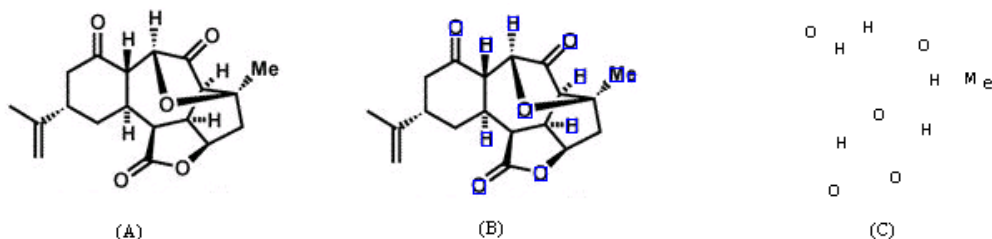


Fig. 5. (a) Original chemical image (b) Segmented image including characters (c) GOOCR output

(4) Line information

The lines in a chemical structure image tend to have uniform length, so the weights of each line in Hough space for a chemical structure image would have similar value. Also, the lines in a chemical structure image are usually shorter than the lines in a non-chemical structure image. Thus, we can exploit this line information from HT as a feature vector to classify chemical structure images. In order to quantify this line information, we used the mean and variance of normalized weights of the lines in Hough space.

The line angle information also can be used as one of the features because there are some patterns in the angle distributions of the lines in chemical structure images. For example, the angle difference of the lines is usually 60° , 120° , 72° , or 108° in chemical structures due to some common structures (e.g., hexagon/pentagon). In this work, we use the angle histogram with 18 bins (each for 20°).

Implementation

Since we are only concerning the length of a line, the thickness of a line should not affect the weight in Hough space. So, before HT, we skeletonize images and make all the lines have the same thickness of one pixel. As a skeletonization method, we use morphological thinning that successively erodes away pixels from the boundary until no more thinning is possible. Fig. 6 shows an example of skeletonization.

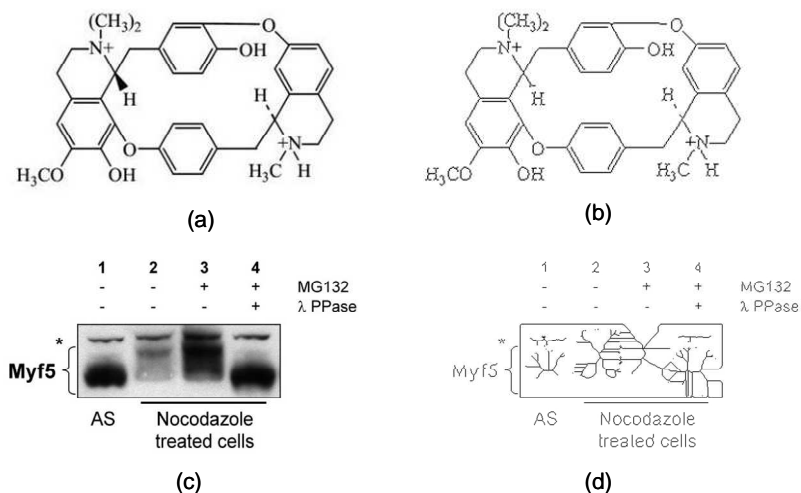


Fig. 6. (a) original chemical image, (b) skeletonized image of (a), (c) original non-chemical image, and (d) skeletonized image of (c).

For line extraction, several variants of HT, which are known as a standard technique for line detection in digital image processing, are used. In its basic form, HT detects lines by mapping the image in the Cartesian space to the polar Hough space using the normal representation of a line in x-y space (Fig. 7. (a) and (b)). So, for all i , where i is the index of a pixel in the image, we have:

$$x_i \cos \theta_i + y_i \sin \theta_i = r_i$$

Since a pixel corresponds to a sinusoidal curve in the Hough space, collinear pixels in the x-y space have the intersecting sinusoidal lines. Therefore, all possible lines passing through every arbitrary pair of pixels in a chemical diagram image are identified by checking the intersection points of curves in the Hough space. Fig. 7. (c) and (d) show the detected line and the corresponding Hough space.

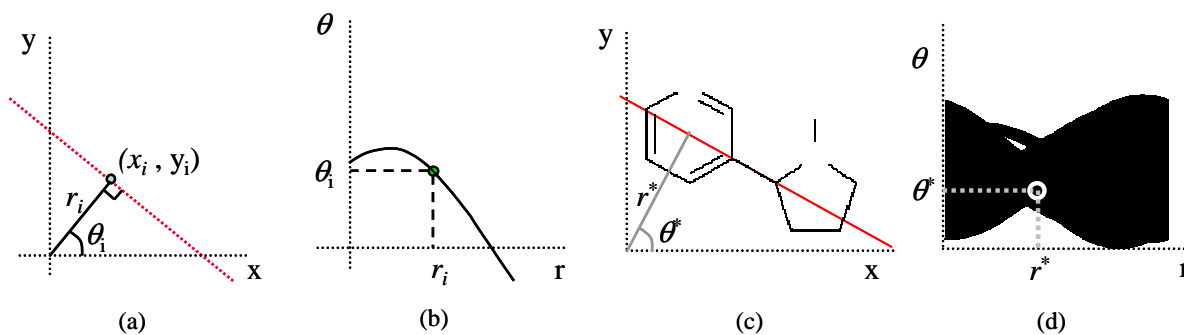


Fig. 7. (a) Cartesian Image Space, (b) Polar Hough Space, (c) Example of HT applied to a chemical structure image, and (d) Hough Space corresponding to (c)

However, the conventional HT does not use pixel connectivity and line width information as important features for line extraction. To correct these problems, we employ the modified Hough Transform [4], where each pair of pixels is assigned a weight based on the probability that the two pixels are originated from a single line segment. In their method, the weight of a cell in Hough space is accumulated for multiple line segments which have the same (r, θ) value, and therefore, we cannot match the weight in Hough space with the length of a single line segment exactly. To overcome this problem, we did not accumulate the weight in Hough space, but we only updated the maximum value of the weight. One more thing that we need to consider in HT is the normalization of weights because each image has different size. Since the diagonal line in an image has the maximum weight in Hough space, we use this value as a normalization factor.

Results

Fig. 7 shows the mean and variance of the weights of the lines for 30 chemical structure images and 20 non-chemical images. As we expected, the variance of weight of chemical structure images has a tendency to be smaller than the variance of non-chemical structure images. Also, the mean value of weight of chemical structure is also smaller than non-chemical structures'. This is because the lines in a chemical structure image are usually shorter than the lines in non-chemical structures such as graph, diagram, and picture.

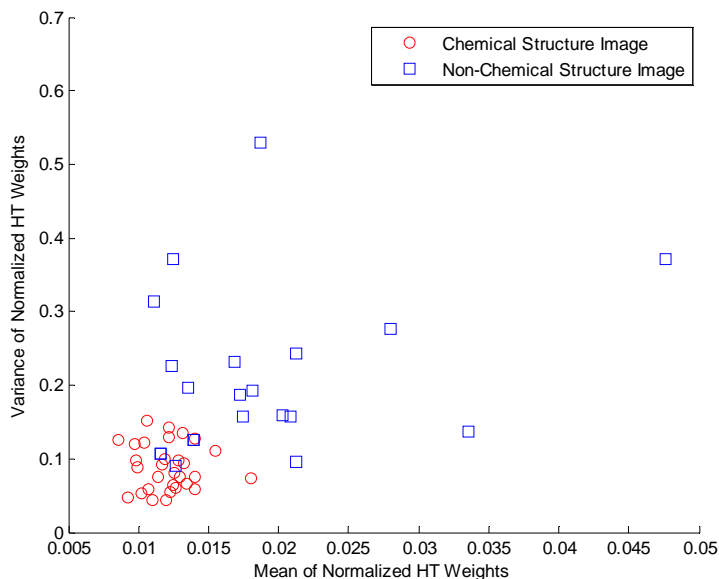


Fig. 7. Mean and variance of the normalized weights of the lines in Hough space

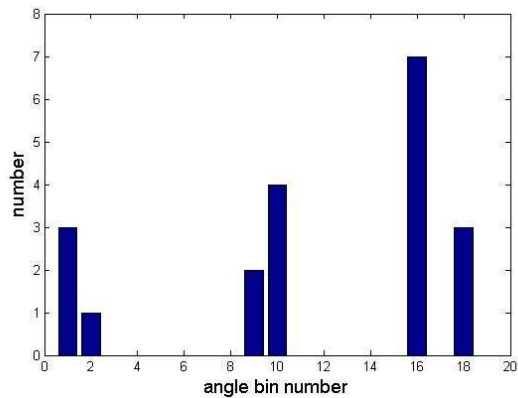
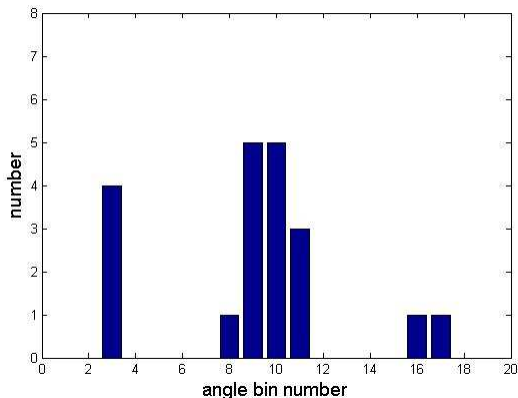


Fig. 8 (a) Line angle hist. of chemical image

(b) Line angle hist. of non-chemical image

The line angle histograms for chemical and non-chemical images are shown in Fig. 8. Since there is an hexagon structure in the chemical image, we can see that lines with an angle with multiples of 60° ($=$ bin num. 3, each for 20°) is observed in Fig. 8(a). On the other hands, in Fig. 8(b), we can see that lines with $0^\circ, 90^\circ,$ and 180° are observed since the non-chemical images contains a graph (i.e., x-and y-axis).

(5) Existence of hexagon/pentagon

Hexagons and pentagons are one of the unique characteristics of chemical structure diagrams. Therefore, the existence of hexagon/pentagon can be a strong indication of chemical structure diagrams. Therefore, we consider the existence of hexagon/pentagon structures as a feature for chemical image classification.

Implementation

While the Hough Transform (HT) can be used to detect analytically defined shapes (e.g., lines, circles, etc), the generalized Hough Transform (GHT) [7] enable us to detect an arbitrary shaped objects. Unlike the HT, several sample points are used in GHT. From each sample point (x_c, y_c) , the distances and angles to the object boundaries are measured. Thus-obtained information is used for object detection. We implemented the GHT for hexagon/pentagon detection using C++.

Results

Fig. 8 shows an example of hexagons/pentagons detection steps based on the GHT using our developed tool.

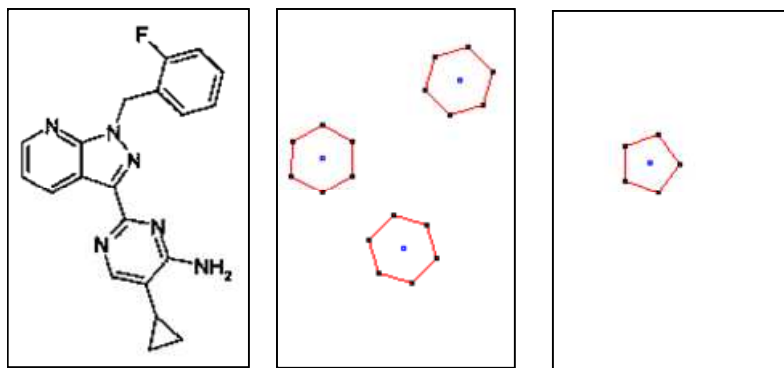


Fig. 8. (a) Chemical image (b) Detected hexagons (c) Detected pentagon.

(5) Feature Extraction

To extract the above mentioned feature information from the input image (chemical or non-chemical), we developed a tool using C++ as shown in Fig. 9. A brief description of the use of this tool is described as follows. First, we can select an input image file using the ‘open’ button. There are 6 functional blocks: (i) ‘color histogram’, (ii) ‘connectivity histogram’, (iii) ‘character reorganization’, (iv) ‘line histogram’, (v) ‘line angle’, (iv) ‘hexagon/pentagon’.

By clicking the buttons in the ‘Feature Extraction’ area, our scheme will extract feature information from the input image and record the result in the output text file. For example, if we click ‘Color Histo’ button, our algorithm extract ‘color histogram’ and ‘connectivity histogram’ data from object images and record the output in the text file. Fig. 9 shows the output of ‘Char. Recog’ button. When we click ‘Batch All’ button, our algorithm extract all 6 features’ output data and record them in the output text file. Then, the collected feature information will be used in the SVM- and KNN-based chemical structure classification schemes.

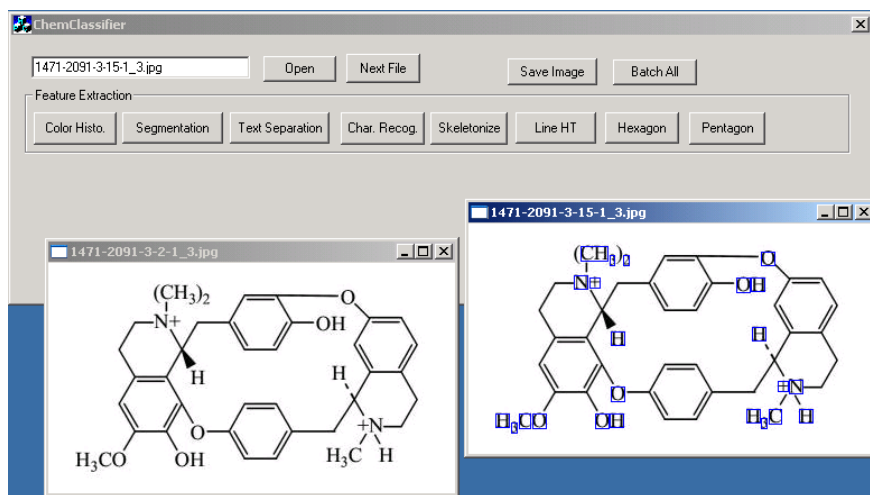


Fig. 9. A Developed tool for extracting image features

4. Simulation Results

In this section, we evaluate the efficiency of our scheme via simulation using *libsvm* [11].

We use the radial basis function (RBF) for SVM kernel. That is,

$$K(u, v) = e^{-\gamma * |u-v|^2}$$

We use 400 chemical (non-chemical) images for the training set and another 400 chemical (non-chemical) images for the test set, respectively. First, we decide the optimal bandwidth (γ) for our RBF kernel via K-fold cross validation. We evaluate the performance of our SVM-based scheme with various numbers of training data. To demonstrate the efficiency of our scheme, we compare the performance of our SVM against K-nearest neighbor (KNN)-based technique. We use *test error rate* as a performance metric throughout the simulation. Each simulation result is an average of 20 runs. The features that we used in the simulation are: (i) color histogram, (ii) connectivity histogram, (iii) mean and variance of weights of lines in Hough space, (iv) angle histogram, and (v) mean and variance of character likelihood. Note that we did not use hexagon/pentagon detection as a feature due to large computational overhead.

(1) SVM Model Selection

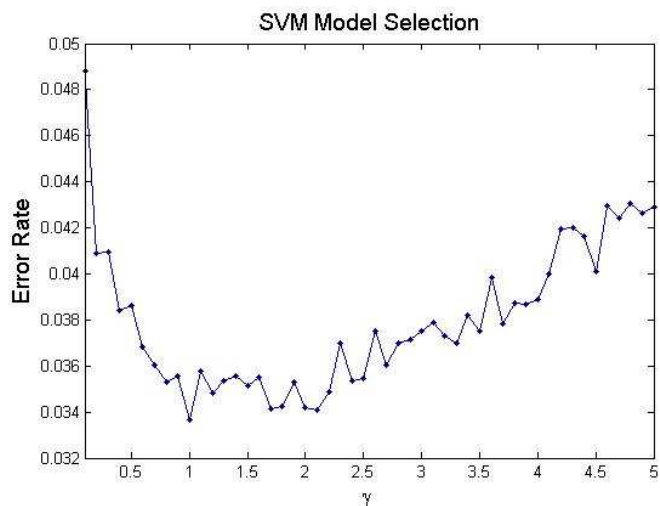


Fig. 10. Error rate with various bandwidth γ

Fig. 10 shows the simulation results of the K-fold cross validation under various bandwidth (γ) values. We set $K=10$ for this simulation. As shown in the figure, if γ is too small ($\gamma < 1$), the performance degrades as γ decreases due to over-fitting. On the other hand, if γ is too large ($\gamma > 2$), the performance also degrades as γ increases. Based on this observation, we use $\gamma=1.5$ for the rest of our simulation. Note that we also tested different K values (i.e, $K=3,5,7,9$) for KNN-based scheme, and we observed that there is not much of difference between them. So we set $K=5$ for the rest of our simulation.

(2) Effect of Number of Training Data

Here we investigate the effect of number of training data on the performance. We compare the performance of the two classification schemes: (i) SVM and (ii) KNN. The simulation is carried out 10 times and the average values are plotted in Fig. 11.

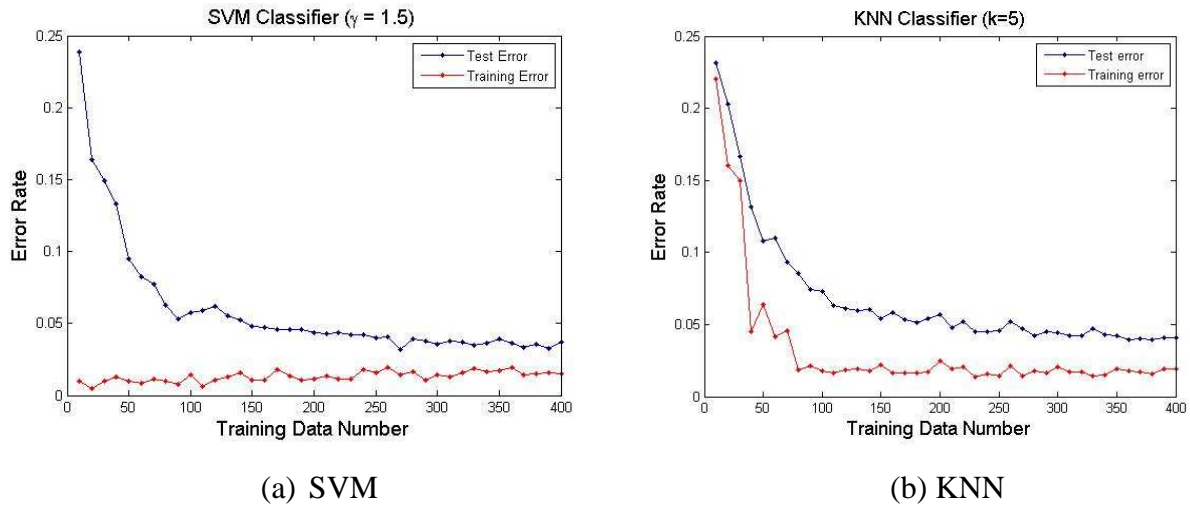


Fig. 11 Effect of number of training data on performance

Fig. 11(a) shows the performance of SVM with various numbers of training data. As one can see in the figure, the test error rapidly decreases as the number of training data increases, while the training error is almost the same as the number of training data increases. The SVM-based scheme shows 96.29% accuracy (385/400) in classification.

Fig. 11(b) shows the performance of KNN-based scheme ($K=5$) with various numbers of training data. Similar to the SVM, the test error significantly decreases as the number of training data increases. In this case, the training error also depends on the number of training images. The KNN-based scheme shows 95.93% (383/400) accuracy in classification.

(3) Performance Comparison

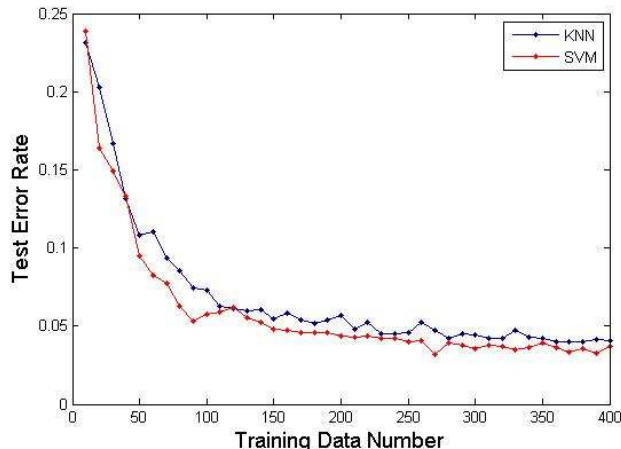


Fig. 13. Performance comparison of SVM vs. KNN

Fig. 13 shows the performance of SVM and KNN-based schemes. As one can see, the SVM-based scheme slightly outperforms the KNN-based scheme under simulated scenarios.

(4) Performance Measure for Each Feature

Table 1 shows the error rate of SVM-classifier when we use each feature separately. As you can see in the table, the line information feature gives a good performance over the other features. That means the line information contains the largest part of the chemical structure characteristics. Furthermore, when we use whole features for SVM-classifier, the performance is significantly improved.

	Color hist.	Connectivity hist.	Line length information	Line angle information	Character symbol	Total
Test Error	0.313	0.173	0.143	0.115	0.216	0.037
Training Error	0.311	0.164	0.135	0.102	0.214	0.015

Table. 1. Error rate for each feature

5. Conclusion

Chemical image classification is very important and fundamental problem in building the chemical structure database which will benefit the chemistry-related research society. In this project, we implemented an efficient SVM-based chemical structure classification tool using C++. We defined several unique features that commonly appear in chemical structure images as a basis for chemical image classification. Simulation results show that our SVM-based classification scheme shows over 96% accuracy in classification. We observed that each feature shows different performance in chemical structure classification. Therefore, future work includes finding optimal weights of features for performance improvement. In addition, it would be also interesting to investigate other features for more sophisticated chemical image classification.

References

- [1] Banville, D. L., "Mining Chemical Structural Information from Drug Literature", *Drug Discovery Today*, Vol. 11, pp. 35-41, 2006.
- [2] Chapelle, O. et al, "Support Vector Machines for Histogram-Based Image Classification", *IEEE Transactions on Neural Networks*, Vol. 10, pp. 1055-1064, 1999.
- [3] Mitra , S. et al, "Data Mining: Multimedia, Soft Computing, and Bioinformatics", *Hoboken: Wiley*, 2003.
- [4] Yang, M., "Hough Transformation Modified by Line Connectivity and Line Thickness", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, pp. 905-910, 1997.
- [5] E. Sojka, A New Algorithm for Detecting Corners in Digital Images, *Proceedings of the 18th Spring Conference on Computer Graphics*, pp. 55-62, 2002.
- [6] Fan, K., Liu, C., Wang, Y., 1994, "Segmentation and Classification of Mixed Text/Graphics /Image Documents," *Pattern Recognition Letters*, 15, pp. 1201-1209.
- [7] Ballard, D. H., "Generalizing the Hough Transform to detect arbitrary shapes", *Pattern Recognition*, Vol. 13, pp. 111-122, 1981.
- [8] Scholkopf, B. et al., "Estimating the support of a high-dimensional distribution", *Neural Computation*, Vol. 13, pp. 1443-1472, 2001.
- [9] GOCR, <http://jocr.sourceforge.net/>
- [10] JChem, <http://www.chemaxon.com/>
- [11] LibSVM, <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

Appendix

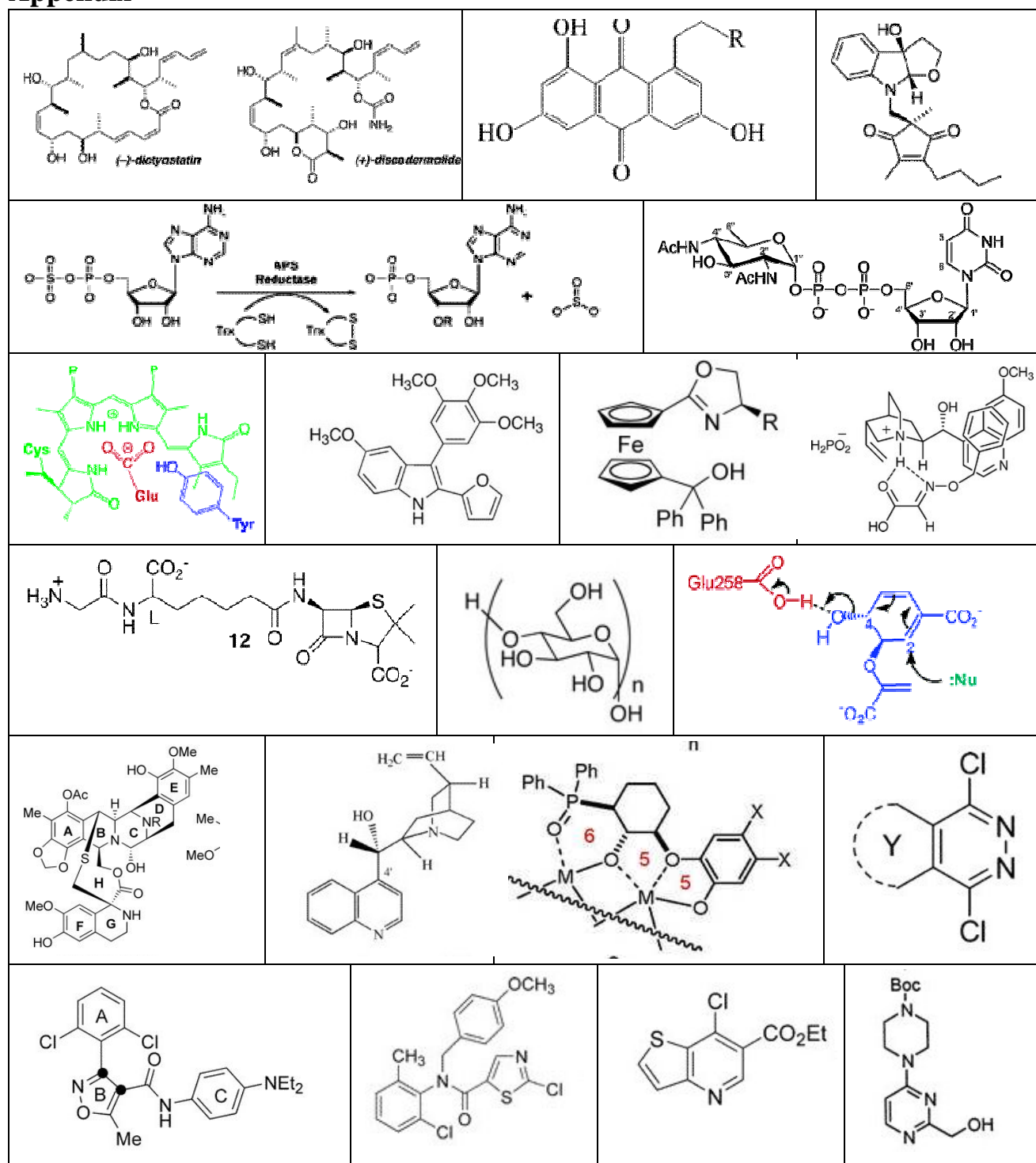


Fig. 15. Examples of chemical structure images used in the simulation

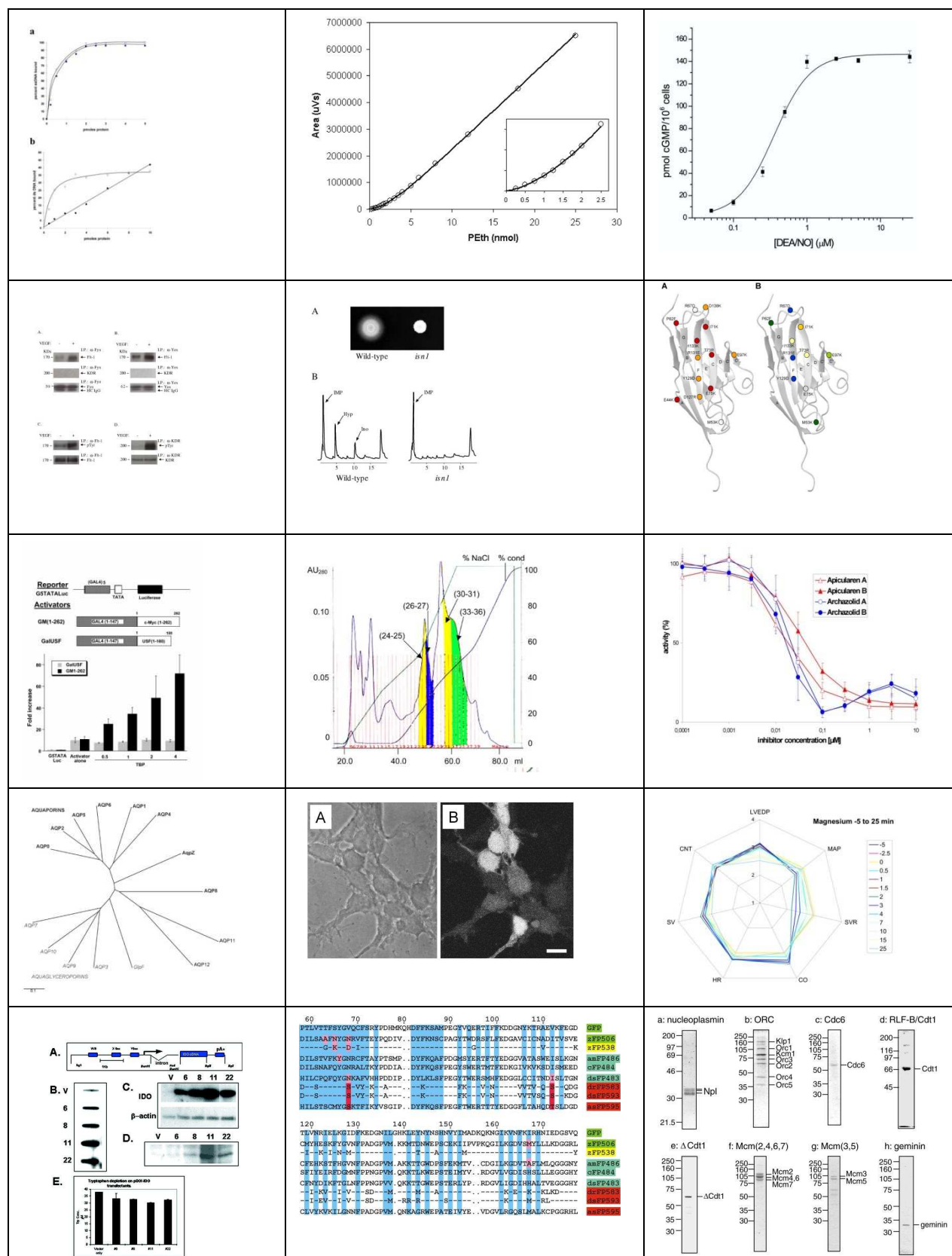
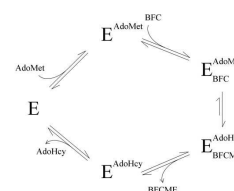
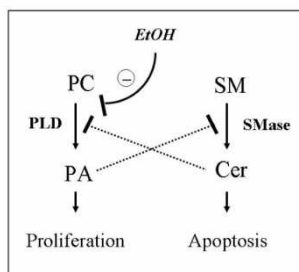
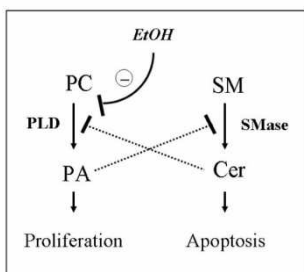
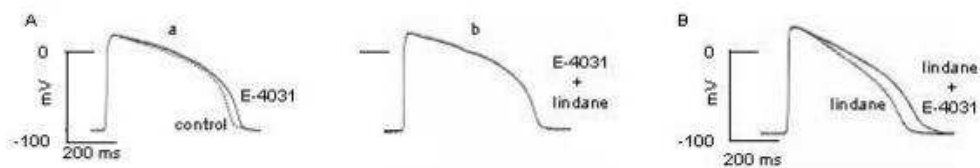


Fig. 16. Examples of non-chemical images used in the simulation

(1) False alarm (Non-chemical → Chemical)



(2) Missing Detection (Chemical → Non-chemical)

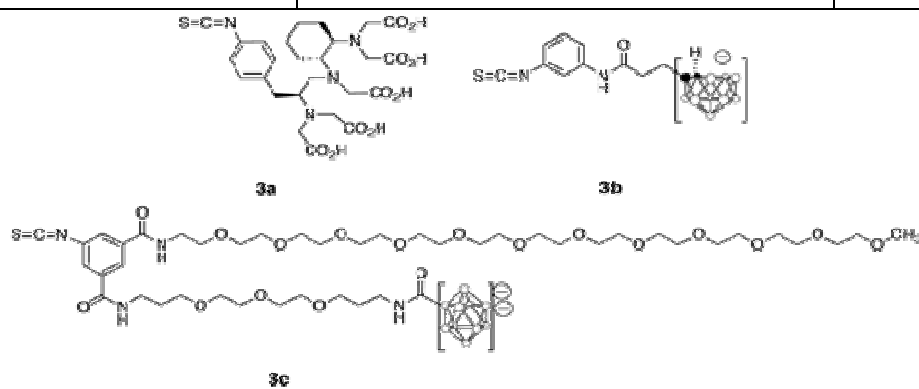
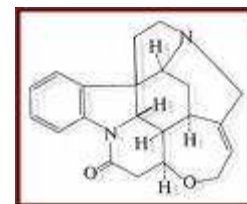
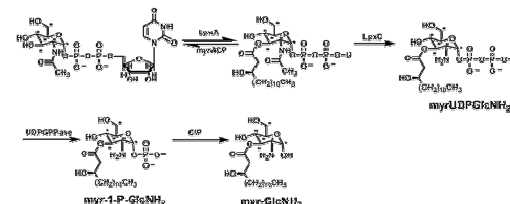
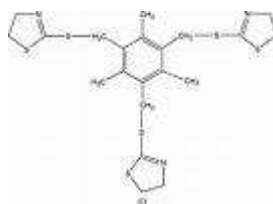
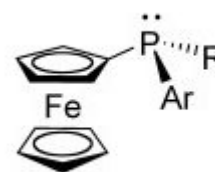
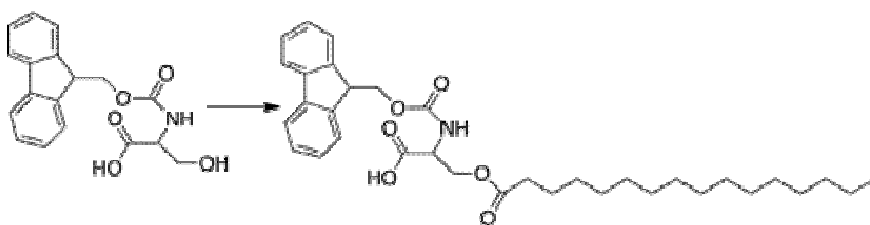


Fig. 17. Examples of chemical (non-chemical) images of classification failure

