

Optical Linear Equation Analysis Using Support Vector Machines

Drew Schmitt and Nicholas McCoy

I. INTRODUCTION

There are numerous applications for the optical recognition and analysis of linear equations. Potential applications include a tool for students to visualize the solution of blackboard equations in the classroom. The steps of our proposed algorithm are shown below

1. Perform image processing techniques.
2. Classify extracted patches using SVM.
3. Reconstruct the equation using centroid and bounding box information.
4. Visualize using WolframAlpha.

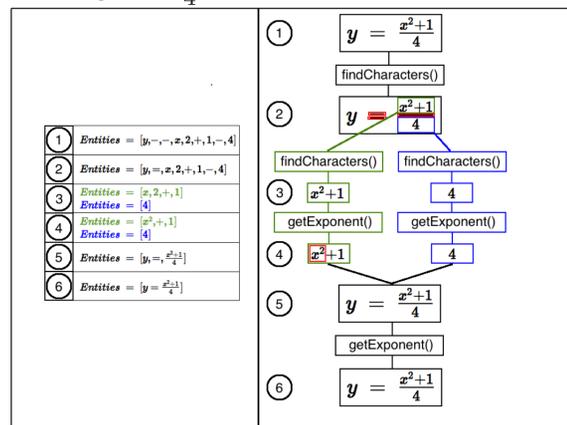
IV. EQUATION CONSTRUCTION

We first process all special characters, such as '-' and 'f' symbols to find their adjoining arguments. Recursively run this procedure until all entities are constructed. Sort items within each entity by the x-coordinate of its centroid.

For each item, recursively attempt to pair it with its righthand neighbor as a base-exponent pair, x^y , according to the relative centroid locations and bounding boxes of the two.

Run this procedure until all characters have been visited while constructing a unified string representing the underlying equation.

The Figure below details the flow of this recursion for the input image containing the equation $y = \frac{x^2+1}{4}$.

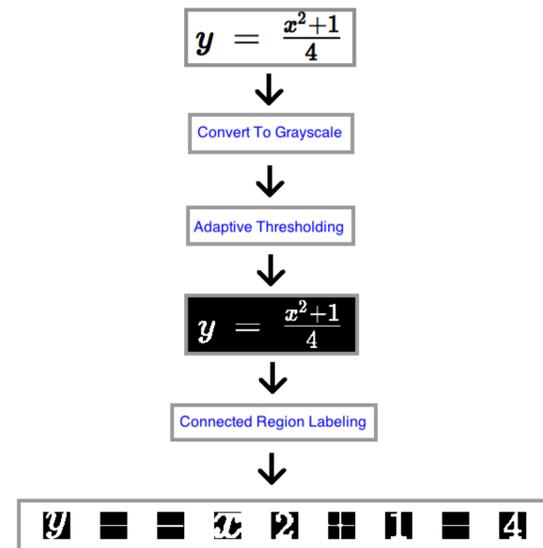


II. IMAGE PROCESSING

The first step is to convert the incoming RGB image to a binary image using locally adaptive thresholding based on Otsu's method.

Connected regions are then located and labeled. Each connected region represents a character in the equation. Properties such as the region's area, centroid location, and bounding box are extracted.

The Figure below provides a visual representation of the general flow of the image processing steps used to extract the character patches from the equation $y = \frac{x^2+1}{4}$.



III. SVM

Given that an SVM is a binary classifier, and it is often desirable to classify an image into more than two distinct groups, multiple SVM's must be used in conjunction to produce a multiclass classification.

A one-vs-one scheme can be used in which a different SVM is trained for each combination of individual classes. An incoming image must be classified using each of these different SVM's. The resulting classification of the image is the class that tallies the most "wins".

For the training data $\{(x_i, y_i)\}_{i=1}^m$, $y_i \in 1, \dots, N$, a multiclass SVM scheme aims to train $\binom{N}{2}$ separate SVM's that optimize the dual optimization problem

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m y^{(i)} y^{(j)} \alpha_i \alpha_j K(x^{(i)}, x^{(j)}) \quad (1)$$

using John Platt's SMO algorithm. In Equation 1, $K(x, z)$ corresponds to a linear kernel function.

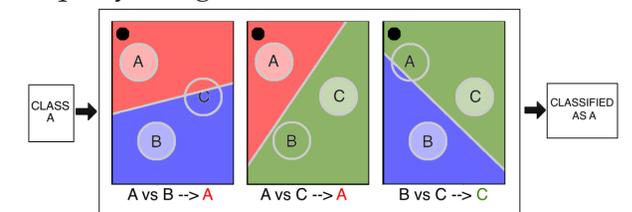
Upon classification, a query image is then counted towards the votes of classes p and q , respectively, using the following scheme

$$\text{votes}_{p+} = \left(1 - \text{sgn} \left\{ \sum_{i=1}^m \alpha_i y^{(i)} K(x^{(i)}, z) \right\} \right)$$

$$\text{votes}_{q+} = \text{sgn} \left\{ \sum_{i=1}^m \alpha_i y^{(i)} K(x^{(i)}, z) \right\} \quad (2)$$

where $\text{sgn}(x)$ is an operator that returns the sign of its argument and z is the query image reshaped into a vector of length N^2 by 1. The query image, z , is then classified as the class that obtains the most votes after passing it through all $\binom{N}{2}$ SVM's.

When the query image is of class A , the A -vs- B and A -vs- C SVM's will be queried with the input patch. If the query patch truly contains an object of type A , then A will win in both of its contests. The Figure below represents this concept visually where the black dot is the query image.



V. RESULTS

The Figure on the near right shows a successful classification output for the input image containing the equation $y = \frac{x^2+1}{4}$.

The Figure on the far right shows the success rate vs. variance of additive Gaussian noise added to the input image. Moving along the x-axis, the input images are corrupted with additive Gaussian noise with a larger variance.

Future work for this research should focus on making the algorithm more robust to noise.

