



Understanding Recognition Technology

An Insider's Guide

Setting the Right Expectations for
a Successful Implementation

www.parascript.com

TABLE OF CONTENTS

INTRODUCTION

Chapter 1: Understanding the Recognition Process.....	3
.....	
Chapter 2: Understanding Errors and Rejects.....	4
.....	
Chapter 3: Confidence Value and The Operating Point.....	7
.....	
Chapter 4: Common Misconceptions.....	10
.....	
Chapter 5: Context and Business Rules to Get the Most Benefits from Data.....	12
.....	
Chapter 6: Why Results May Not Meet Expectations.....	15
.....	
Conclusion.....	17
.....	

APPENDICES:

Appendix A: Mathematical Model to Optimize the Tradeoff Between Error and Rejects.....	18
Appendix B: How to Choose a Threshold Value.....	20

INTRODUCTION

Companies, organizations and government entities worldwide use recognition technology to process documents, forms, checks, mail and other important business content with high speed and accuracy. Recognition software employs different technologies; OCR (Optical Character Recognition) recognizes data that is in the form of machine print; ICR (Intelligent Character Recognition) recognizes handwriting, both handprint and cursive writing; among other state-of-the-art image analysis and pattern recognition technologies.

In order to get the most out of a recognition software implementation or upgrade, it is crucial to understand how recognition technology works and to set appropriate expectations. Recognition technology employs sophisticated algorithms and neural networks, referred to as artificial intelligence that “thinks” differently than humans and makes different types of errors. It is not intended to replace human labor completely, but rather to increase productivity and decrease the amount of manual intervention required. Ultimately, reducing manual intervention reduces the costs associated with forms and document processing.

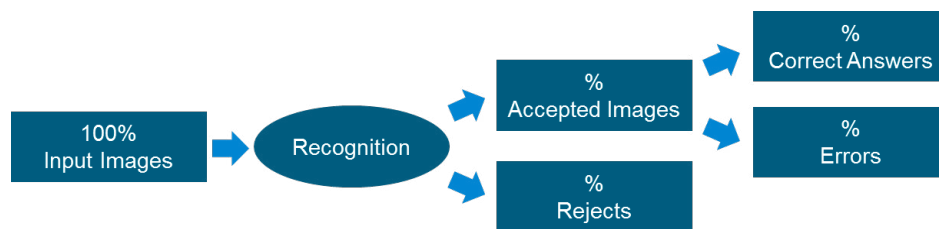
This e-book addresses the deeper technical aspects regarding how recognition software works, explains operational details, provides examples, and identifies how and when human involvement is required in the process.

CHAPTER 1:

Understanding the Recognition Process

Understanding the recognition process and how recognition technology approaches it differently than humans is key to a successful implementation.

The typical process is illustrated in the figure below. All input images are fed to the software. After recognition, processed data is divided into two streams: accepted answers and rejected answers.



Accepted images are those that have high probability of correct recognition. Rejects include words for which the recognizer cannot guarantee the required recognition accuracy. How does the software know what this required accuracy is and how was that determined in the first place? We'll analyze that in the next chapter.

The primary measure of the recognizer's productivity is the percentage of answers that were accepted (accept rate) or the percentage of answers that were rejected (reject rate). However, this measure is not sufficient to judge the accuracy and productivity of the recognizer because it does not allow us to estimate how many errors are present among the accepted answers.

Another useful measure characterizes the accuracy of accepted data: the percentage of words accepted that were recognized correctly or the percentage of words accepted that were recognized with an error (error rate). Error rate can be calculated as a percentage of the entire number of fields processed by the recognizer or as a percentage of the number accepted of words. We will not elaborate on the specifics of each method here. Each method has its advantages for different kinds of analysis.

CHAPTER 2:

Understanding Errors and Rejects

There are three possible outcomes when recognition engines attempt to read any data: the correct answer, error, or reject. It is important to understand errors and rejects and how to find the right balance between them.

Errors

Errors refer to the instances when a recognition engine reports an incorrect result. The problem with determining errors is that there is no way to find out that it is in fact an error unless there is a second opinion or external, verified information from sources other than the recognized image.

Errors are not exclusive to automatic recognition. Humans also make errors, but of a different type than automatic recognition errors. Many human errors are related not to recognition, but to typing, also called keying. Combining automatic recognition with manual data validation greatly reduces error rates. Advanced recognition technology uses voting algorithms, which significantly reduces error rates, since the final result is a combination of several engines. Using voting has proven to achieve much lower error rates than those provided by double keying (data verification done by two separate individuals). But even these algorithms do not eliminate errors completely and there will be some errors that require handling by the solution.

If there is a large deck of test images with known actual values, called truth data, statistical measurements of how often these errors occur can be obtained. The statistical estimate for the frequency of errors is called error rate. The number of test images required to determine the error rate depends on the accuracy of the project. The set of images should be representative of real application images, not just in quantity, but also in quality. They should include the types of images and the quality of images that are encountered in a real stream of documents. There should be at least 10 errors that occur among the accepted answers to make measurements of the error rate

“Reducing rejects by increasing error rates allows companies to process more information automatically, without the need for human intervention. This lowers processing costs.”

statistically significant. For example, if seeking to measure an error rate at around 1%, the total amount of images required is at least 1000. If the required error rate is 0.5%, the minimal size of the set should be 2000 images.

Rejects

Rejects refer to the situation when the answer has a confidence value below the established threshold. In many cases, the rejected answer is correct, which makes data validation easier and faster.

Rejects may be caused by the inability to process some specific input or by the need to reduce error rate to the level the application can tolerate. Rejected items are typically processed manually or require recapturing of input data.

Tradeoff Between Errors and Rejects

Depending on the application needs, users can either reduce rejects by increasing error rates or reduce error rates by increasing reject rates.

Reducing rejects by increasing error rates allows companies to process more information automatically without the need for human intervention, which results in lower processing costs. This is appropriate for applications where having data with some errors is not detrimental. One example for this type of scenario is processing magazine subscription forms.

On the other hand, reducing error rates by increasing reject rates results in more data to be validated, which is more suitable for applications where every error counts or

there is very little room for errors, such as in check processing or documents that contain social security numbers or other sensitive data.

The operating point determines the right balance between errors and rejects for a specific application.

For a detailed explanation of how to optimize the tradeoff between error and rejects, see Appendix A.

CHAPTER 3:

The Importance of the Operating Point

The operating point is a critical number since it is used to build a business case, establish an ROI, and set the benchmark to measure against. It is composed of two numbers, the read rate and error rate. In order to understand the importance of the operating point and how it is determined, it is first essential to understand “confidence value”.

Confidence Value

There is confusion among those using recognition technology regarding how to judge the accuracy of the technology. Creating a metric is critical.

When performing recognition or verification, the software evaluates an image and provides an answer of recognition with an internal metric associated with it. This metric is calculated using a very complex algorithm and is, somewhat confusingly, called “*confidence value*”. A confidence value is a certain number within a selected scale, for example it can range from 0 to 100 that indicates the reliability of the recognition answer.

As humans, we can easily confuse a confidence value of 90 as 90% confident. The software does not see it that way. It is simply a reflection of a grade impacted by many different variables. These are all *relative and specific to the application at hand*. **It is very important not to confuse with a percentage. The scale could easily be from 15 to 178. However, the greater the confidence value, the more confident the recognizer is about the particular answer.** Based on the example above, to the software a confidence value of 90 is of little real meaning until it is compared to a large dataset (thousands). This comparison is done to set the required threshold by the application.



The operating point is the most critical number in recognition.

The following are some rules and information for using confidence values:

- A certain confidence value can be chosen as a “threshold” depending on the application.
- Answers that have confidence values lower than or equal to the threshold value are unreliable. In this case, the field is rejected and must be processed manually.
- Answers with the confidence values above the chosen threshold are accepted.
The percent of the accepted answers to total number is called the read or accept rate.
The number of correct results among the accepted answers expressed in percent is called accuracy. The number of erroneous results among the accepted answers expressed in percent is called error rate.

The higher the chosen threshold, the lower the number of the accepted answers, but the higher the accuracy of the accepted answers.

Operating Point – The Most Critical Number in Recognition

While confidence value is critical to defining the business case, it is really a developer's metric, and often misleading. The number that is critical is the “*operating point*”. In recognition the operating point is the critical number because it builds the business case, establishes the ROI, and sets the benchmark to measure against. It is composed of two numbers: read rate and error rate.

In the example below, see an operating point of 85% read rate @ 1% error rate. This means that out of 100 documents, software will successfully read 85 and is likely to produce an error on 1. The other 15 documents would fall below the required accuracy and be passed to a human for review. For comparison, a human typically produces errors at a 3% rate.

85% read @1% error

Sample Operating Point

Additionally, it is not accurate to assume that humans will error on the same 1% that the software does. This example illustrates that the way to produce the most accurate and most efficient results is to blend usage of both software and trained personnel.

Using Confidence Values to Achieve the Required Accuracy and Read Rate – The Operating Point

In order to determine an operating point, a data specialist will (among other things) run the images for recognition, sort the confidence values from largest to smallest and compare results against the truth data. At this point, a clear line will be defined in the data where confidence values take their meaning, and higher confidence values start to have relative value against lower confidence values, as seen by the ratio of correct answers to incorrect answers. The “sweet spot”—a balance between read rate and error rate—becomes the operating point, and can be adjusted to the tolerance levels of the company or organization.



For a detailed, step-by-step description of how to find the operating point and the right threshold, see Appendix B: How to Determine a Threshold Value.

In summary, to achieve good recognition results requires a combination of many different factors that are assessed over a large statistical dataset. This dataset is used to produce the operating point, which is the bottom line for developing a successful business case and implementation of recognition or signature verification.

CHAPTER 4:

Common Misconceptions About Recognition

Introduction

There are two common misconceptions regarding automated recognition technology: (1) that all items in a stream have an equal level of difficulty and (2) that values should be recognized first and then use rules and context. Errors and inefficiencies that can result from employing these misconceptions can be resolved with appropriate tuning.

Misconception #1:

All items in a stream have an equal level of difficulty for recognition, regardless if they are processed automatically or manually.

The items that cause the most errors and rejects are not random. It makes sense that the items with the highest probability for rejection are ones that are the least standard and therefore are the most difficult to read. This applies to both automated and human (or manual) verification. This means that it requires more effort to key a rejected item than an average item in a stream and operators make more errors on rejected items than on the whole stream of documents. Therefore, it is wrong to assume that a rejected item could be recognized equally as well as an average item.

To solve this issue, recognition engines can be tuned specifically for the particular goal of the application. For example, to efficiently read large numbers (such as a 10-digit account number) in a stream of documents, the engine is tuned so that smaller numbers, such as 3 digits in that stream are likely to be rejected. If the goal of the application is different, for example to detect all small numbers in the stream, it would be inefficient to use the engine tuned for 10 digit account numbers. There are methods and approaches designed to detect items that meet the particular goal of the application to enhance the solution and its efficiency.

“ Common misconceptions relate to the level of recognition difficulty and the use of context and rules.

Misconception #2:

It is more effective to first recognize values and then use rules and context.

It is usually easier to solve the problem of detecting whether certain values meet specified conditions than to recognize the specific values. For example, recognition engines may be unable to read the value of a particular character in an image. However, if the main goal of the application is other than to correctly recognize each character, the software can deliver a confident and correct answer that meets the needs of the application. For example, the task of detecting CAR/LAR mismatch on a check can be solved more efficiently if the engine is specifically targeted to find a discrepancy between these two fields, rather than reading the contents of each field and then comparing the results from each item (CAR and LAR). The latter case will more likely contain more errors. Another example of using rules and context to improve recognition is when there is additional input from other sources, such as in the case of mobile deposits where a user keys in the dollar amount. Recognition engines would be able to provide better answers using this information during the recognition process.

MYTHS
MYTHS
MYTHS
FACTS

This process is especially important if the error rates tolerated by the application are much lower than the error rates that can be achieved with automatic recognition or manual keying. Using recognition with higher error rates as a basic building block for any solution that has much lower error tolerance may result in a very unstable solution. Conversely, addressing the target condition directly usually results in a much more efficient and more robust solution.

CHAPTER 5:

Context and Business Rules to Get the Most Benefits From Data

Introduction

The effective use of context and business rules are tools users can use to process information faster; reduce reliance on manual data entry and related costs; and ultimately derive the most benefits from their data.

Benefits of Using Context and Rules

Considering all of the ways that businesses rely on data, the benefits of increasing the accuracy of information and processing it more quickly can be significant. Efficient and accurate processing enables companies to improve internal and external transactions; save time; increase collections; improve customer service; improve collection rates and reduce time spent researching problems due to incorrect data. Most organizations stand to gain both monetary benefits and increased business efficiencies. In some industries accuracy rates can have an even greater impact including healthcare, where disease codes are used frequently, and in banking where each digit must be correct.

Context and business rules are two tools that companies can use to process information faster and improve data quality. These “magic bullets” perform some of the “thinking” or logic for organizations processing data. They determine how information is processed, increasing the accuracy of the data recognized by the software, speeding up the process and reducing the amount of required manual data entry. These rules allow more information to be recognized automatically during the “first pass” of processing, transfer the same context to those entering data manually (keyers) during the second phase of data entry, and help to identify and verify low confidence fields during the final validation stage.

Using Context

Context plays a significant role in the recognition process by helping to explain the characteristics or properties of data within a field. When humans read handwriting or print, they look at entire words — and even the entire document — to correctly identify the content. Knowing a range of probable meanings makes the task of reading much easier. This is why recognition engines use context as an effective and flexible tool to compensate for the inherent ambiguity of handwriting and to improve recognition accuracy.

Rules for context can be applied seamlessly during recognition (for instance, to tell the engine that the next character must be a 0 (zero) and not an O), during data entry (with the same numeric rule for keyers to follow, or by having the system set up so that individuals cannot enter anything to the contrary) and in validation to check one last time for accuracy. In many programs, a letter or number in a field may be highlighted to prompt verification of questionable information.

Using Rules

Business Rules are referred to as “the logic” and help to ensure that the data meets defined criteria. Rules apply “if/then” thinking or specific scenarios. Rules can be used to automatically populate fields with database lookup.

Example 1: If this field is X, then the answer must be Z in order to determine the next valid answer or to confirm accuracy of one or more fields

Example 2: Automatically populate fields with database lookup; if the code is 00123, then the name, address and phone are automatically x, y, z

Example 3: Eliminate errors, such as when an area code/address combination is known and can be used to check/repair a digit on an answer to determine if it should be a 3 or an 8

Example 4: Match ZIP codes with appropriate mailing addresses for address recognition

Example 5: Verify the numeric amount (i.e., \$108.35) on a check with the alphanumeric amount (i.e., One hundred eight and 35/100) for check processing applications

Both context and business rules can be used to increase accuracy during each stage of data capture—in character recognition, to determine answers, increase accuracy and reduce keying; during keying, with sample tests, and finally, in validation. Especially when used together, context and business rules offer greater speed and efficiency and reduce costs and time. Looking at their own data, most organizations can determine numerous ways to employ context and business rules for process improvement.

Getting Started

First, examine the type of form and the information you are collecting along with the volume of data that you need to collect. Assessing both these factors simultaneously will help to determine the best combination of speed and accuracy—and which should be weighted more heavily—for your organization's given needs.

Next, gather the individuals working closest with the data to go through each form, field by field, to consider:

- What fields are most important?
 - Which fields have the greatest negative impact on the organization if they are incorrect? Which fields provide the most valuable information to the organization?
- Which fields already have rules set in place?
- For those without rules, what kinds of context can be built for each form?
- Are some fields always fixed length?
- Do some fields contain a certain pattern of characters or numbers? For example, should dates and times be formatted in a certain manner?

Inventory should be taken at this time to assess whether existing tables or databases are available for use in employing business rules or can be created. These might include:

- Postal database (which already exists in some data entry systems),
- Common vendor/payee names,
- A list of part numbers or account numbers, and
- Common provinces and cities outside the U.S. to build out addresses.

CHAPTER 6:

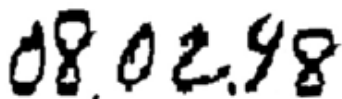
Why Results May Not Meet Expectations

The goal of every recognition engine is to produce the highest accuracy possible. However, the type and quality of images being recognized can and often produces significantly different results, depending upon which recognition technology is used. For example, in the case of constrained handprint, the more distinct the individual written characters are in the image, the more closely matched are the results of each method.

On the other hand, and this is far more typical, if the individual characters are more ambiguous and illegible or resemble others characters as in the case of unconstrained handwriting that can mix-up hand print and cursive styles, the greater will be the difference in the recognition rates and accuracy of the recognizers. To produce better results on “poor quality” images and hard-to-read characters, advanced handwriting recognition technology employs the use of dynamic vocabularies.

Example 1:

The image below extracted from a form field is the age of a dependent child under age 21. On the form, if the child is over 21 then the field is left blank. Because of this context information, it is clear that the date in the image is 08-02-98 even though the digit 9 looks more like a 4.

A handwritten date in black ink on a white background. The date is written as "08.02.98". The first two digits "08" are separated from the rest by a period. The next two digits "02" are also separated by a period. The final two digits "98" are written together. The digit "9" is written in a way that it looks like a "4" if not for the context.

Having defined the correct answer, consider how the results of recognition would differ without vocabularies, static vocabularies, and dynamic vocabularies.

A recognition engine working without a vocabulary will read 08-02-48, which is incorrect based on the context information. A recognition engine working with a static vocabulary (look-up table) to perform post-recognition validation will often fail to produce a correct



Advanced handwriting recognition technology employs the use of dynamic vocabularies to produce better results on poor quality images and hard-to-read characters.

answer as well. This is because it may produce only one answer (4) for the first digit of the year.

Recognition with dynamic vocabularies produces greater accuracy since the engine determines during the recognition process that there cannot be digits other than 9, 0, or 1 in the year position. Although the symbol looks like a 4, this number as an option is not possible and is excluded from the recognition process. The result is that a 9 will be the answer.

Example 2:

Recognition of letters written in handwriting presents even more of a challenge. In the image below, it is not clear if the first symbol is a “d” or a combination of a “c” and an “l”. Recognition without vocabularies would not produce a reliable result.

A handwritten word in cursive script, which appears to be 'clear'. The first letter 'c' is written in a way that could also be interpreted as a 'd' or a combination of 'c' and 'l'.

Use of a static vocabulary approach would require that both hypotheses are produced and stored until the end of the recognition process when validation based upon the look-up table takes place. If the word were longer, there would be more hypotheses to be stored and analyzed. The result would be slow and would potentially yield lower recognition accuracy.

Using dynamic vocabularies, there is no necessity to analyze and store all possible hypotheses of segmentation. If the dynamic vocabulary does not contain a combination of c and l at the beginning of the word, the only possible segmentation solution is “d”.

There are a lot of examples and stories of poor handwriting recognition and the lack of dynamic vocabularies is a big reason. But using the proper technology for the job, there are just as many good stories of successful use of handwriting recognition.

The conclusion is that dynamic vocabularies have the ability to produce higher recognition accuracy over what users have come to expect.

Conclusion

Companies and organizations with complex high volume processing requirements benefit from automated recognition solutions. The technology has advanced substantially so that current software options include sophisticated image analysis and pattern recognition algorithms.

Businesses want to gain the most value for their software by achieving the highest possible accuracy rates accompanied by lower costs. To accomplish this, users need to comprehend several important aspects of the technology that include error and reject rates, confidence values and how these statistics work together to produce the operating point; how vocabularies are used to identify errors; and appreciate the nuances in the software to use context and business rules to set criteria to improve accuracy. The more knowledgeable users are regarding what their solutions do and how to use them, the more likely their software will deliver more speed and accuracy.

Want to know more about recognition technology?

Check out Parascript solutions - *FormXtra* and *FormXtra Capture*. You can also Talk to an Expert by clicking on Live Chat at the top of the Parascript website - www.parascript.com.

Appendix A:

Mathematical Model to Optimize the Tradeoff Between Error and Rejects

Introduction

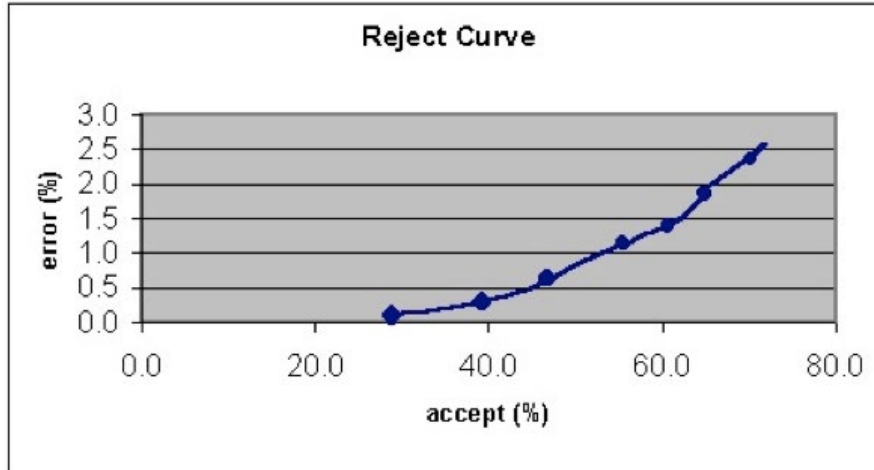
The difference between errors and rejects and the tradeoff between them was presented in Chapter 4. Here the reject mechanism is presented to understand the mathematical model to optimize the trade off between errors and rejects.

Reject Mechanism

The reject mechanism helps to guarantee the specified error level required by an application. Recognition engines usually return an answer accompanied by a value parameter called **confidence value**. The confidence value is a number within a scale, for example it can range from 0 to 100 or 0 to 1000, and indicates how confident the engine is that a particular answer is correct. If the engine is less confident that a recognition result is correct (confidence value is below the chosen **threshold**), the result can be rejected. Confidence values provide a flexible, controllable mechanism that allows tuning for specific needs. For example, if an application cannot tolerate an error rate higher than 1%, it is possible to choose a confidence value threshold so that answers accepted by the system as correct will contain no more than 1% errors (on average or with high probability). Those applications that are less sensitive to error rate but instead have a requirement to minimize expenses associated with data processing can use the reject mechanism to trade errors for rejects, i.e. to set up a solution that provides the biggest savings.

Reject Curve

The best and most accurate ratio between errors and reject is chosen using the following mathematical model. Both reject and error rate are functions of the confidence level threshold. Since rejects depend on the confidence level threshold monotonically, error rate could be considered to be a function of reject. This function is called **reject curve**.



Sample error-reject curve.

Each dot on the graph is associated with a certain confidence value that may be chosen as a threshold value. The higher the selected threshold level, the lower the number of errors in the accepted results.

Error Ratio

At each specific point of the curve reflecting dependence of error rate from reject rate there is an **error ratio** which reflects how many items should be rejected to eliminate 1 error or how many items will be excluded from reject if the engine is permitted it to make additional errors. Usually it is expressed as 1:N (one error for N rejects). In many applications, N is chosen in the range from 2 to 10. Mathematically this ratio is minus a derivative of a function "Errors from Rejects". If there is a processing cost associated with handling errors and a cost associated with handling rejects, the best choice for error rate versus reject rate corresponds to the point where this ratio is equal to the ratio between cost of rejects and the cost of errors.

Appendix B

How to Determine a Threshold Value

1. Prepare a representative set of images.

If the number of images in the set is not sufficient, the results of the investigation will not be representative. Some random variations in the set of images may occasionally alter the results of the analysis.

The number of images depends on the accuracy required for the project. Parascript recommends evaluating the error rate for at least 10 errors that occur among the accepted answers. For example, if an acceptable error rate for recognition of a field is 1% and at least 10 images make 1%, then the set of images represents statistically significant results (in this case, a single non-typical error changes accuracy by only 0.1%). Accordingly, the set should contain at least 1000 images. If the required error rate is 0.5%, the minimal size of the set should be 2000 images. The set of images should be representative, not just with regard to their number, but in the quality of the images of this particular field, the variety of character styles, and the variety of words that may be encountered in the field.

2. Run recognition on the images. In order to make a choice of a threshold value easier, prepare the following table (for example, using a spreadsheet program) and sort the results in decreasing order of the confidence value.

Statistical Information

N	Answer	Confidence Value	Expected Answer	Result Is Wrong	Number Of Images	Number Of Wrong Answers
---	--------	------------------	-----------------	-----------------	------------------	-------------------------

Each table row contains the following data for each recognized image:

N	The table row number
Answer	The result of recognition
Confidence Value	Its confidence value for the answer
Expected Answer	Correct ASCII value, determined by human processing of the image (truth information)
Result Is Wrong	Enter 0 or 1 for this column according to the following: 0 – The fields "Answer" and "Expected answer" are the same 1 – The fields are not the same

Number Of Images	The number of rows up to and including the current one
Number Of Wrong Answers	The sum of "Result Is Wrong" column

To determine a confidence value for the threshold value, build a new table extracting the following three columns from the first table (see Table A-2).

Statistical Information (modified)

Note: If there are several rows with the same confidence value, take the bottom one for investigation.

Confidence Value	Number of Images	Number of Wrong Answers
------------------	------------------	-------------------------

Once a confidence value is selected, the other two fields from the corresponding row have the following meaning:

Number of Images	The number of accepted answers for the chosen threshold
Number of Wrong Answers	The number of recognition errors that occur among the accepted answers

The confidence value associated with the required error rate is a threshold level to be determined. Table A-2 shows the relationship between the accept rate (or reject rate) and the error rate. The relationship can be displayed as a graph.

Figure 4.3 presents a sample error-reject curve. Each dot on the graph is associated with a certain confidence value that may be chosen as a threshold value (row of the table). The higher the selected threshold level, the lower the number of errors in the accepted results.

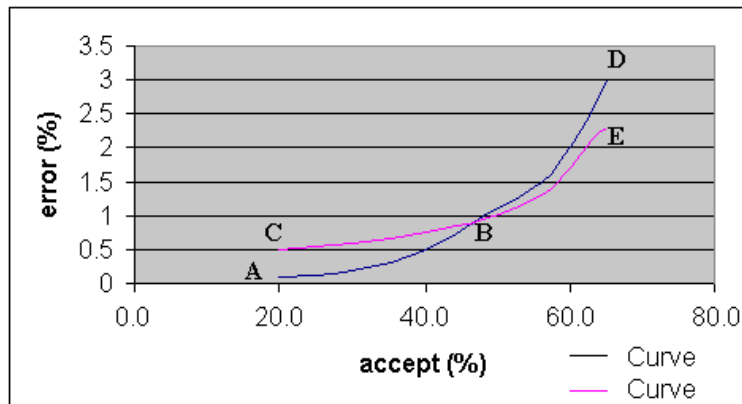


Figure 4.3 error-reject curve

The reject curve allows us to choose a threshold value more accurately than using a table because of the curve shape in the required error value.

The reject curve not only describes the effect of moving a threshold, it also allows an effective comparison of the performance of different recognizers or performance of the same recognizer working with differently-tuned parameters. The better the recognizer (or the better the parameters are tuned), the closer the reject curve is to the X-axis. However, it is not always possible to compare two curves, making it worth comparing different parts of them. In Figure 4.4 we plot reject curves of a recognizer working with differently tuned parameters (Curve 1 and Curve 2). It is clear that as application requirements change, one curve reflects better results. For example, if an application requires low error rates (for example 0.5-1%), reject curve 1 (section AB) shows better results than reject curve 2 (section CB). This means that the first option of tuning parameters of the recognizer is preferable. For higher error rates, curve 2 (section BE) shows better results than reject curve 1 (section BD).

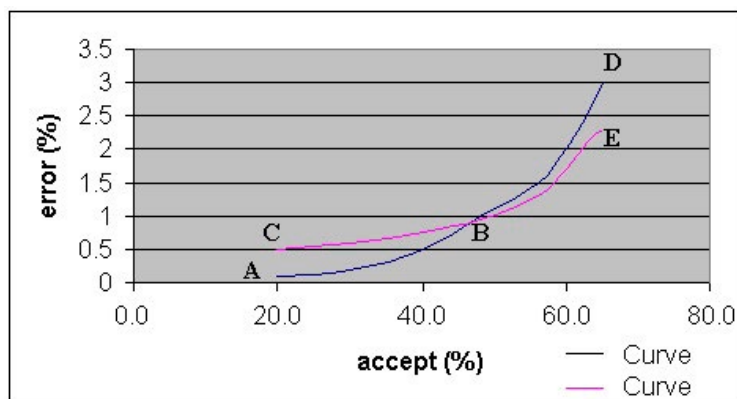
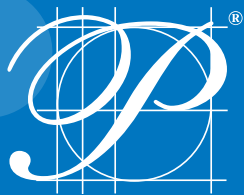


Figure 4.4 Reject curves of a recognizer working (Curve 1 and Curve 2)



PARASCRIPT®

www.parascript.com