TRIPLE: A Multi-Strategy Machine Learning Approach to Target Recognition

Bir Bhanu and John C. Ming

Honeywell Systems and Research Center
3660 Technology Drive, Minneapolis, MN 55418

ABSTRACT

Current target recognition systems are unable to modify their behavior based on the dynamic environmental changes which occur around them. In order to perform robustly in unconstrained, outdoor environments, a target recognition system must be able to adapt its representation of this dynamic environment while maintaining acceptable performance levels. Machine learning technology offers some promising solutions to many of these problems faced in the target recognition scenario. Learning allows a system to use situation context, to adapt its representation of the changing environment, and to improve the system’s recognition performance over time. This paper describes an innovative system which combines learning and target recognition into an integrated system which accomplishes the above tasks. This system is called TRIPLE: Target Recognition Incorporating Positive Learning Expertise. It uses two machine learning techniques known as explanation-based learning and conceptual clustering, combined with a knowledge-based reasoning system, to provide robust target recognition. A complete description of the TRIPLE system, as well as a simple example showing the system’s behavior, is presented.

1. INTRODUCTION

Target recognition systems currently lack the ability to adapt to changing environmental conditions and to modify their behavior based on the context of the situation in which they are operating. In order to be effective in dynamic outdoor scenarios, a robust recognition system should be able to automatically acquire necessary contextual information from the environment. Most target recognition systems lack this capability. Their performance begins to quickly degrade when subjected to problems such as variable lighting conditions, image noise, and object occlusion.

Due to recent advances in machine learning technology, some of the problems encountered in the target recognition domain seem to be resolvable. Learning allows an intelligent recognition system to use situation context in order to understand images. This context, as defined in a machine learning scenario, consists of a collected body of background knowledge as well as environmental observations which may impact the processing of the scene. The vision-learning cycle of an advanced target recognition system would involve the following steps:

Sense - acquire an image and apply initial image processing algorithms.

Understand - using present background knowledge and scene observations, determine those objects of interest in the image.

Act - based on the objects found, act according to global system goals.

Update - modify system knowledge using image observations so performance will improve next time.

This cycle repeats each time the recognition system is required to produce results. Because the system dynamically reacts to the appropriate stimuli in the environment, it continuously adapts its internal knowledge to maintain acceptable performance levels.

In addition to the vision-learning cycle described above, other desirable features to be incorporated in an advanced target recognition system are: (a) the models used by the system to represent targets, contexts, and other system knowledge should be dynamic data structures. This specification allows the learning component to quickly modify the behavior of the system by changing the data on which it operates; (b) most data should be of a symbolic, qualitative nature, thus avoiding the problems encountered in dealing with quantitative information. Using qualitative information, we do not have to rely on obtaining precise geometric representations of target; (c) the system has to be able to handle problems such as imprecise segmentation, occlusion, noise, etc. Since advanced target recognition systems are required to operate in real world situations, they must be capable of handling these image problems that will inevitably occur; (d) the system should exhibit improved performance over time. This improvement may come in the form of faster recognition times, improved recognition accuracy, and higher confidence in system results.

Machine learning will facilitate two main breakthroughs in the target recognition domain: automatic knowledge base acquisition and continuous knowledge base refinement. The use of learning in the knowledge base construction will save the user from spending the enormous amount of time necessary to derive target models and databases. Knowledge base refinement can then be incorporated to make any necessary changes to improve the performance of the recognition system. These two modifications alone will serve to significantly advance the present abilities of most target recognition applications.

To validate the concept of a target recognition system with integrated machine learning capabilities, this paper presents an overview of a new approach to target recognition. The system currently under implementation is called
TRIPLE: Target Recognition Incorporating Positive Learning Expertise. The system uses a multi-strategy technique; two powerful learning methodologies are combined with a knowledge-based matching technique to provide robust target recognition. Using dynamic models, TRIPLE can recognize targets present in the database. If necessary, the models can be refined if errors are found in the representation. Additionally, TRIPLE can automatically generate a new target model and recall it when that target is encountered again. Finally, since TRIPLE uses qualitative data structures to represent targets, it can overcome problems such as image noise and occlusion.

The two main learning components of the TRIPLE system are Explanation-Based Learning (EBL) and Structured Conceptual Clustering (SCC). Explanation-based learning provides the ability to build a generalized description of a target class using only one example of that class. Structured conceptual clustering allows the recognition system to classify targets based on simple, conceptual descriptions rather than using a predetermined, numerical measure of similarity. While neither method, used separately, would provide substantial benefits to a target recognition system, they can be combined to offer a consolidated technique which employs the best features of each method.

The remainder of this paper will provide the reader with most of the details of the TRIPLE system. To fully explain some of the machine learning techniques which are being utilized, section 2 presents a brief overview of machine learning, including the explanation-based learning and conceptual clustering methods. Following the coverage of these techniques, section 3 outlines the approach to target recognition used in TRIPLE. Section 4 provides an example of TRIPLE’s abilities using automobiles as the targets to be recognized. Finally, section 5 presents the conclusions of the research to date and discusses future plans for the current system.

2. MACHINE LEARNING TECHNIQUES

The ability to reason and the ability to learn are the two major capabilities associated with an intelligent being. Today, machines have been given the ability to reason by developing algorithms which duplicate, to a limited extent, a reasoning process. Unfortunately, machines do not have the ability to modify or update this reasoning process if the situation so dictates. Nor do they have the ability to learn new concepts which are encountered in the processing of information. The lack of both of these abilities has severely limited the effectiveness of computers to work in unconstrained environments.

Fortunately, this situation is slowly beginning to change. Progress is being made in the development of algorithms which can modify their behavior as the environment in which they operate changes. Machine learning now exists as a valid portion of the AI and psychology fields, has dedicated conferences and journals, and is being added as a feature in a broad variety of computer science applications. While many of these additions are very superficial and do little to improve the performance of the system to which they are added, research has led to fundamental advances in several areas of machine learning technology.

Advances in the machine learning field originated in the late 50’s. Rosenblatt12 developed the perceptron, a self-organizing system, although it never achieved the success anticipated by many of its advocates. Following this effort, researchers concentrated on tasks such as learning from examples and language acquisition which relied on significant amounts of data and time to search the problem space. Also known as inductive learning systems, these learning from example techniques were applied in a wide range of application domains. The most influential research performed in the area of inductive learning was accomplished by Winston.16 His structural learning system formed concept descriptions from a set of carefully selected examples of the concept as well as "near misses". Near misses represent concepts that are similar to the one being learned, differing only in a small number of very significant details. Positive examples serve to generalize the concept while near misses provide the necessary amount of specificity. Many inductive learning systems followed the early work done by Winston. Dietterich and Michalski15 present a general comparison of various systems which incorporate learning from examples.

In most systems which utilize inductive learning, a method known as generalization is used to extract the common features which characterize a group of objects. Generalization has been used in various AI contexts for many years, although the results have been difficult to compare due to substantial differences in implementation and domain of application. Mitchell10 casts the generalization problem into a search framework and compares various approaches to the problem. The search space consists of the possible generalizations that can be constructed for a given problem. Methods of generalization can then be characterized by a search strategy such as depth-first, breadth-first, or version space technique.

Connell and Brady2 developed a system which learns the descriptions of two-dimensional objects including aerial views of airplanes or silhouette images of various hand tools. This technique produced structured production rules which were used to recognize subsequent instances of similar objects. Using inductive generalization techniques which allow for disjunctions, Connell and Brady's method was one of the first systems which could learn from real image data.

The trend in machine learning has been to incorporate techniques which can derive the maximum amount of information from single examples, using analytic methods rather than empirical ones. Current research is now directed at developing programs which provide learning from observation and discovery. Explanation-based learning (EBL) and structured conceptual clustering (SCC), both of which are used in the TRIPLE system, are emerging methodologies which employ a high level of inference. EBL, classified as a learning by observation technique, uses inference to construct a useful concept description from a single example of that concept. SCC, which is also a learning from observation method, employs an even higher level of inference since it does not rely at all on any user input to classify a group of targets into conceptually simple groups. These techniques will now be discussed.

2.1 Explanation-Based Learning

Most of the early systems which utilized learning from examples were able to achieve impressive results compared to methods which did not use any form of machine learning at all. However, it was discovered that the user may find it difficult or impossible to provide the learning mechanism with enough examples to properly generalize the concept description. Additionally, the system was unable to justify the generalization which was produced from a set of examples; the user could not obtain a description of how the observation had been reached. In order to aid the development of learning systems which, using applicable background knowledge, could generate a concept description
from a single, user-provided example. At the same time, the system also created an explanation as to why the example yielded that particular generalization. Initially referred to as explanation-based generalization (EBG), this technique is now commonly called explanation-based learning (EBL).

The generalization process employed by EBL can be viewed as a search through the possible concept description space. The objective is to locate the correct definition of the concept being learned. To constrain the size of this search space, EBL relies on knowledge of the problem domain. Since the extra information present in a set of multiple examples is not provided, EBL must use some other type of knowledge to sufficiently generalize the single example. This information exists in the form of relevant background knowledge which is given to the system. Using the background knowledge, EBL is able to produce a valid generalization of the single example. Additionally, it creates a justification of the generalization in terms of the background knowledge used to produce that generalization. This justification is called the explanation of the concept example.

The origin of the explanation-based approach to machine learning can be traced back to the STRIPS system developed by Fikes et al. which learns generalized robot path planning motions. From this initial work, the creation of a concept description from a single example was then formalized by DeJong. In this paper, he introduces the term explanation-based generalization. As DeJong continued working on his system, others began work on their own extensions or changes to the initial EBG method. The most prominent of these was the research done by Mitchell, Keller, and Kedar-Cabelli. They proposed a standardized approach to explanation-based generalization. This technique creates an explanation structure, represented as a proof tree, which serves as the generalization of the concept. The generalization is a two-step process. The first step forms the explanation that separates the relevant and irrelevant feature values present in the training example. Second, the explanation is analyzed to determine the constraints (numeric values, numeric ranges, or enumerated values) on the feature values which will allow the explanation to apply in general.

In response to perceived inadequacies in the work by Mitchell et al., DeJong and Mooney proposed further revisions to the EBG system. DeJong and Mooney felt that the term explanation-based learning was more complete than explanation-based generalization since the approach seemed to be applicable to both concept refinement and concept generalization. This version of EBL serves as the basis for the target model creation and refinement component of the TRIPLE system described in section 3. While Mitchell et al.’s version of EBG produced the object generalization in a two step process, the EBL technique simultaneously forms an explanation of the training example and builds the generalized concept of the training example. In addition, EBL is capable of specializing a previously-defined, over-generalized object concept. This refinement ability is very valuable since it provides a partial solution to the problem of generalizing non-independent conjunctive sub-goals. In other words, after several passes over a concept description which may contain conflicting information, EBL is capable of properly representing this concept, while EBG would have failed. Since this effectively provides a form of explanation-based specialization as well as explanation-based generalization, DeJong and Mooney have aptly named the method explanation-based learning.

2.2 Structured Conceptual Clustering

Classification of similar objects has traditionally been accomplished using mathematical techniques such as numerical taxonomy and cluster analysis. Using a pre-defined set of object features or attributes, these techniques would compute clusters of objects; clusters are characterized by high intra-class similarity and low inter-class similarity. Clustering methods are generally unable to identify groups of objects which represent conceptually simple concepts since they rely on numerical measures of similarity. In addition, the results usually must be interpreted by expert data analysts to decipher the classification results.

Problems with numerical clustering techniques and the proposed solutions have been numerous. However, they do not address some of the fundamental problems inherent with the clustering methods. First, numerical clustering techniques are context free. They make no use of any contextual or background information while computing object similarities. Psychological tests have shown that humans make use of significant amounts of context when classifying objects. Second, most methods are unable to expand the feature space in order to discover new features which may yield ideal classifications. Simple, linear combinations of object attributes can often be used to locate intrinsic groups of data. Third, the techniques do not have the ability to select and evaluate object attributes when generating clusters. The attributes are simply used to compute distances between neighboring objects and clusters. Finally, the classification results still have to be interpreted because a characterization of the clusters is not produced.

The problems mentioned above have caused researchers to design systems which try to model the classification techniques used by humans. People normally group objects using a conjunction of attributes which represent conceptually simple ideas. At the same time, they also consider the context in which the objects act, which often determines important features which can be used in the classification task. Using a collection of background knowledge to provide context, Michalski developed a new version of clustering which identified groups of objects which represent the same type of conceptually simple ideas that humans tend to use. This approach to classification is known as conceptual clustering. Since it does not rely on a teacher to pre-classify the objects, conceptual clustering is superior to earlier systems which use learning from examples.

An implementation of the technique Michalski developed is presented in a paper by Michalski and Stepp. This method, known as the CLUSTER/2 program, constructs a classification of objects only if a given class can be specified by a conjunctive concept which uses selected object attributes. Quality measures such as the fit between the clustering and the observed events, the inter-cluster distance, total number of features used in the concept description, and the number of features which individually discriminate among all the clusters have been used to judge the quality of the selected object attributes.

To validate the performance of the CLUSTER/2 algorithm and compare its performance with classical clustering approaches, Michalski and Stepp tested the classification ability with 18 other numerical taxonomy methods. Only 4 of the 18 numerical methods were able to produce the conceptually appealing results of the CLUSTER/2 program. These results show that conceptual clustering achieves many of the classification goals used by humans.

As a further extension to their work, Stepp and Michalski constructed a new conceptual clustering system which
incorporates three main changes from the previous technique: objects are complex and require structural descriptions; relevant attributes may not be initially provided and should be dynamically determined in that case; and rules of inference are used to derive useful high-level concepts from the initial low-level information. This new version of conceptual clustering is called CLUSTER/S. To produce valid classifications, the system is provided with a general goal of classification. Using the supplied goal, the system then references the collection of background knowledge to determine the relevant attributes and features useful for clustering.

The background knowledge is organized into a network structure called a Goal Dependency Network (GDN). The information present in the GDN can represent general-purpose knowledge as well as domain-specific knowledge, both of which are necessary in the problem solving process. General-purpose knowledge is made up of fundamental constraints and criteria which specify the general properties of classification. The domain-specific information contains inference rules for deriving new descriptors and rules for determining which descriptors will be relevant. Given a high level goal of classification, the GDN specifies the related sub-goals and any associated object attributes which are relevant at that level in the network. If the relevant descriptors are not present, the background knowledge is used to derive new descriptors which are applicable. The Goal Dependency Network plays a vital role in the construction of meaningful classifications.

The ability of CLUSTER/S to handle complex, structural objects relies on the information present in the GDN. Using the goal of classification, the system is able to determine which structural elements of the object are most useful in selecting the appropriate classification scheme. The advantages of this approach include:

1. Ability to handle compound objects which require structural descriptions in order to be easily classified.
2. Use of goal-directed inference from information provided by the GDN.
3. Formulation of new object attributes using the GDN.
4. Allows the system to be model or data driven.

The ability to select appropriate classification features and to generate new attributes when necessary places the conceptual clustering technique into the area of machine learning referred to earlier as learning from observation.

In the domain of target recognition, most complex targets are represented as a structured collection of sub-parts. The ability to cluster such descriptions is very useful in this application. In the remainder of this paper, Stepp and Michalski's CLUSTER/S system will be referred to as Structured Conceptual Clustering (SCC) because of the ability to handle these structural descriptions.

3. TRIPLE TARGET RECOGNITION SYSTEM

Although machine learning has been used in many applications, very little effort has been made to combine several learning techniques together. Learning methodologies are used independently to provide adaptive ability and improved system performance. However, TRIPLE incorporates explanation-based learning and conceptual clustering into a multi-strategy learning approach to target recognition. By utilizing the capabilities of each learning method at appropriate steps in the recognition and learning process, TRIPLE overcomes the inherent limitations present in these learning techniques. EBL's main limitation is the matching time required when the number of target models becomes large. SCC's has problems with model biases when the number of object class examples is small. Combining the ability of EBL to characterize a target using a single training example with SCC's efficient method of organizing objects once they have been properly modeled yields an integrated learning system which handles the target recognition task.

Figure 1 shows the various components of the TRIPLE target recognition system. Basically, the characterization abilities of explanation-based learning are combined with the aggregation capabilities of the conceptual clustering technique. The EBL component is used to create and refine target models while the SCC component is used to structure the EBL-generated models into an efficient classification tree. The domain rules and facts relevant to the recognition problem are stored in the background knowledge database. The goal and sub-goal information is stored in a modified version of a Goal Dependency Network (GDN) originally used by conceptual clustering. The GDN has been adapted so that the EBL component can access the necessary information.

Figure 1: The TRIPLE target recognition system
The TRIPLE system, as indicated in Figure 1, is composed of six main components:

1. A system training set which initializes the target recognition process.
2. A target database, factual knowledge, and a goal dependency network. These items are combined to form the background knowledge pertaining to the selected recognition domain.
3. An explanation-based learning component which provides generalized target descriptions.
4. A structured conceptual clustering system which arranges the current set of target descriptions into a classification tree.
5. A model matching component which uses the existing classification tree to recognize an unknown target description.
6. A segmentation and feature extraction component which processes the images and provides the necessary object features and relationships.

Each of these elements will be described in detail in the following sub-sections, followed by a description of the recognition-learning cycle used by the TRIPLE system.

3.1 System Training Set

The initial input to the TRIPLE system consists of a set of target examples and a collection of background knowledge pertinent to those targets. For each target which will be recognized by TRIPLE, the user must supply a set of 1 or more representative examples for that target class. Each set of examples will be analyzed by the EBL system and a generalized target description will be created. Note that the input data for each target class need only consist of a single example since EBL is capable of producing a correct generalization from only one target. However, as will be discussed in section 3.3, EBL will be able to generalize a single conceptual description if provided with more than one example.

In addition to the examples for each target which will be recognized, the user must also provide the initial description of the goal dependency network and any factual knowledge which may be necessary when the learning systems attempt to make inferences on the target models. The GDN contains the goal hierarchy necessary for the recognition and learning systems to derive the applicable target attributes when recognizing new targets, refining existing models, or processing incomplete data.

3.2 Target Models, Factual Database, and Goal Dependency Network

The collection of background knowledge used by the TRIPLE system is composed of three main groups of data: target models, factual data, and a goal dependency network. The target models, which are originally built by the EBL component, are stored separately in the target model database in case they are needed for refinement purposes later. Although the target model information will also be represented in the classification tree, the actual schema which defines each model is kept in the database for use by the EBL system when characterizing a target class. The features and relationships which are labeled as relevant by the EBL component are tagged in the schema and will be used in the SCC system. The models are dynamically modified by the recognition and learning elements when missing or incorrect attributes and relationships are discovered.

The factual knowledge originally given by the user is maintained in the fact database. This knowledge consists of a set of rules and facts which define the domain in which the recognition process is operating. For example, if the recognition domain is automobiles, the factual knowledge may contain information regarding plausible locations, orientations, uses, and other contextual data for different type of vehicles on the road. This information is used in conjunction with the learned target models by the EBL, SCC, and recognition components. As the various components attempt to apply this information, inconsistencies or gaps may be found. These problems will propagate the assertion of new facts and the retraction of incorrect or useless facts. Thus, the factual database will be as dynamic as the models which depend on this data.

The modified version of the GDN in the TRIPLE system is primarily used by the EBL component when determining attribute relevancy. Although it was originally designed to be used by the SCC system, the GDN information is used in the EBL phase of the recognition loop. The version of the GDN used in the EBL phase contains two high level goals: creation of a new target model and refinement of an existing target model. The selection of the appropriate goal is described in the next section. Using the goal which is received, the GDN accesses the goal-subgoal hierarchy and selects the attributes and relationships which are useful in characterizing the particular target. This alternative application of the GDN information is possible since the main responsibility of the GDN is to provide the SCC component with a set of relevant target features; this task is now accomplished by the EBL system.

3.3 Explanation-Based Learning

The explanation-based learning component of the TRIPLE system is responsible for two critical functions:

(a) Processing the training examples provided by the user.
(b) Explaining target recognition failures to improve system performance.

Each of these tasks will now be described in detail.

(a) Processing training examples:

The first step in the target recognition process is to acquire and represent a set of target models. This job has traditionally been very difficult due to the amount of work necessary to generate a correct model. The TRIPLE system uses the power of EBL to simplify the modeling process. Since EBL can generalize a target description from a single example, the user merely provides a set of target attributes and relationships in the training set. From this data, EBL selects the relevant attributes and relationships using the modified version of the GDN. Other applicable background knowledge which provides useful data transformations is used here as well. These transformations are used to infer high-level target attributes from the various input attributes.

The schema which is used to store this new target will still retain the entire list of attributes and relationships which were originally provided to the EBL system. The relevant attributes which have been selected will be tagged in order to separate them from the rest of the attributes. The remaining attributes are retained in case model refinement is ever needed. Some attributes will be indicated in the GDN as statistical attributes that the recognition component uses to provide another adaptive capability to the recognition system.

If any of the attributes are statistical, they will be initialized
at this time. This process will be more fully described in section 3.5.

Since all training examples are processed sequentially before the target recognition process begins, the training phase also insures that the target generalizations are not identical. This situation can result from a collection of background knowledge which is not diverse enough to distinguish the various training examples. Alternatively, the set of target attributes selected by the user may not be broad enough to individually characterize each target class. In either case, the EBL system performs a simple comparison against all existing schemata during the training phase. If an identical match occurs, the user will be notified by the TRIPLE system and must then supply the additional data in order to separate those target classes. After the training phase, this problem will not occur again because the unknown targets are first checked against all existing models before the learning loop is ever entered. Thus, it is impossible for EBL to create the same generalization twice, except during this stage.

An additional extension to the standard EBL technique is the capability to generate a single explanation from more than one representative example. This ability is useful when the user is trying to characterize a target model that may not be sufficiently captured in a single training example, such as when modeling a 3D target. Using only a single example in this situation could bias the resulting generalization towards that particular example. In order to prevent the possibility of this bias, the EBL component of the TRIPLE system allows the user to create a single target model with up to 3 examples of that target. There are two ways to accomplish this goal. First, each of the examples can be individually generalized and these generalizations can be intersected to provide a single model. Alternatively, the similarities of each example can be located and used to provide a generalization. The first method is preferable since it is easier to intersect the generalizations due to the smaller number of remaining attributes and relationships. Multiple examples will hopefully eliminate any biases which may be implied from using only a single example.

Using the above modifications to the standard EBL technique, the target model training task has been effectively automated in TRIPLE system design. After characterizing each of the targets, a copy of the schema will be stored in the background knowledge database and another copy will be sent to the SCC component for integration into the classification tree.

(b) Explaining recognition failures:

The main task of the EBL component is explaining the various types of recognition failures that occur. EBL must be provided with two pieces of information in order to properly explain these failures. First, the system must have the target description which caused the recognition failure. Second, the type of failure which resulted from this target must also be determined. The failure type is used as the goal concept used in the generalization process. Most EBL systems require that the user provide the goal concept from which the generalization can be derived. However, the TRIPLE system does not require the user to provide this kind of information. Instead, the model matching component determines the appropriate goal concept and sends it to the EBL system. The goal concepts are not modified when the recognition system is not able to properly recognize a new target. TRIPLE's EBL component is designed to process two main types of recognition failures:

(1) Failures caused by the presence of a new target.

(2) Failures due to an incomplete or incorrect target model.

Failures which result from encountering a new model are handled in the same manner as when processing the user-supplied training examples. This process was described previously in this section.

If an incorrect model causes the recognition failure, EBL can explain this event and alter the model to avoid the problem in the future. There are several types of target model errors which EBL can effectively handle. If the current model contains an attribute target attribute and relationship which is not relevant, EBL will remove the relevant attribute or relationship tag from that feature in the model schema. If the model contains an attribute whose value is not useful, EBL can modify the value. Finally, if a target model is not using an attribute or relationship which is relevant, it can be tagged as relevant in the model schema. Since these types of recognition failures are located and labeled by the model matching component, EBL can minimize the search space when trying to locate the incorrect data.

3.4 Structured Conceptual Clustering

Normally, conceptual clustering applications classify hundreds of examples simultaneously, many of which belong to the same class, thus requiring the use of a similarity-based method to correctly characterize the targets present in a given cluster. The similarity-based method performs the characterization task defined in Section 4. However, in the TRIPLE system, SCC receives only one pre-characterized example which represents a target class from the EBL system. The EBL component has already performed the task usually handled by the GDN; all relevant target models and relationships have been identified. Since SCC relies on conceptual simplicity as a quality measure, instead of numeric quality values which depend on the number of samples in a cluster, it can produce a valid clustering which contains only one sample per cluster.

The measure of simplicity determines how precisely the classification tree distinguishes various targets. For example, assume the target recognition domain is automobiles. If the measure of simplicity is very coarse, it may be impossible for the system to separate different types of cars, although cars and trucks are distinguishable. As the measure of simplicity becomes more complex, different types of cars can be identified (sedan, coupé, etc.).

To provide greater efficiency in the conceptual clustering process, SCC does not always create a new classification tree. If a model refinement operation has been selected, only a portion of the classification tree has to be changed. By tracing the classification tree until the branch at which the incorrect feature is located, SCC can merely re-cluster the rest of that branch. Since all target models will be isolated along only one branch of the tree, there is no danger in rearranging individual branches of the tree.

If a new model is added to the classification tree, the tree will have to be reconstructed since new attributes may be present in the new model which are not present in the tree. In addition, the new model may interact with the existing target models in such a way that features which were not previously useful as important classifiers can now be used to distinguish target classes. Thus, the conceptual clustering process will be applied to the entire group of target models, including the new target which has been created. The classification tree can then be passed to the model matching component for use in identifying unknown targets.
3.5 Knowledge-Based Model Matching

The model matching component of the TRIPLE system uses the classification tree produced by SCC in order to identify unknown targets. The model matching component has access to the knowledge base so that any necessary data transformations or inferences can be made. This component has two functions within the TRIPLE system: recognition of targets which already exist in the model database and identification of the two types of recognition failures which cause the system to enter the learning loop.

When an unknown target is received by the recognition system, it is first checked against all items currently in the model database. This matching procedure is done by using the classification tree. Since the model schemata are represented in a search tree format, the recognition process is far more efficient than comparing the new target with each model individually. Because some of the features present in a target model may be missing in the image observations due to segmentation problems or occlusion, the model matching component monitors the parsing of the classification tree to decide on the type of recognition or failure which has occurred. Each of these possibilities will now be briefly discussed.

Complete Matching: If the model matching component can successfully traverse the classification tree, match all necessary attribute and relationship criteria, and arrive at a leaf node which contains a target model, that model has been correctly matched. During the parsing of this tree, the recognition element has some flexibility in terms of slight variations in attribute values and a small number of missing attributes. However, if any attributes are missing, they must be minor in nature; global target features cannot be safely ignored. Confidence in a successful match is computed based on the amount of variation in attribute values as well as the significance of any missing attributes.

Incomplete Matching: If the model matching component can correctly parse the high level nodes of the classification tree (these nodes specify global attributes of the target models) prior to exhausting all available image data, all targets present in any leaf nodes at the end of that branch can be hypothesized as valid models for the data which is present. Matching global features and relationships implies that the target is lacking the details necessary for a precise identification. However, this information can guide the segmentation and feature extraction algorithm which can be prompted to provide additional information on the target being identified. If no more feature information can be obtained, the model matching component provides the complete set of possible target identifications.

Occlusion: If the recognition process is not able to match any high level classification attributes, but can recognize many low level attributes and relationships, the presence of occlusion is very probable. In this case, a confidence level can be computed based on the reliability of the features which lead to the matching. For example, if the global features of a car (length, height, etc.) cannot be matched, but the shapes of the hood, bumper, headlights, and fenders are present, the recognition system can use this information to identify the target as a car. A confidence threshold is used to insure that the model matching component does not match ridiculously simple features and report a valid matching.

Model Refinement: Model refinement is determined by a process which is very similar to complete recognition. The parsing of the classification tree should proceed without missing more than a few target attributes or relationships. However, the failure is encountered when comparing the values stored in the attributes themselves. While the attribute itself may be present in the target model, the values in the model attributes may be different than the values in the image attributes. This situation implies that the target model contains most of the correct attributes but needs to be refined by updating the values present in the attribute variables. In addition, any missing or extra attributes present in the target model can be inspected during the refinement process to determine whether they should be included or removed, respectively.

New Target Model: This situation is encountered when the model matching component cannot apply any meaningful portions of the classification tree to the data received from the image. The image data is passed to the learning loop of the TRIPLE system for characterization and integration into the classification tree. If the data contains enough high level and low level details to properly characterize a new target model, it will be processed by the EBL system and incorporated into the classification tree. Otherwise, the EBL system will report the failure to understand the data and the object will be classified as unknown.

The recognition element updates the statistical variables for the given target model if and when that model is used to recognize a target. This process allows the system to gradually determine that certain features have a higher utility since they are always used in the recognition process. In addition, the variable values of certain attributes can be slowly modified by the recognition process in order to overcome any initial bias that may have been derived from the initial construction of the target model. This factor is especially useful for those models which are activated by the learning system during normal operations. While the user is responsible for providing examples which are highly representative of a target class during the training phase, target models acquired during the recognition phase of the system may be subject to a much higher level of noise. Thus, the model matching component can slowly adapt these models as more targets of this type are recognized and the system determines a better value for many of the attributes which characterize that target. Changes in attribute values made by the model matching component will be very gradual compared with the changes which result from activating the model refinement process during the learning process of the TRIPLE system.

3.6 Segmentation and Feature Extraction

The general purpose segmentation and feature extraction component of the TRIPLE system extracts the necessary low level features and relationships from the image data which are then used by the knowledge-based model matching component in the target recognition process. The feature extraction process is designed to locate the most prominent features in the image data and to determine the significant relationships between these features. The features of interest in the image include regions, edges, corners, arcs, and ellipses.

Once the primitive features have been located, the feature extraction process then considers the spatial relationships between various features. If a predefined orientation of a set of features is located, the feature extraction process hypothesizes the presence of a high level symbolic feature. For example, in the domain of automobile recognition, if a pair of concentric ellipses is found in the image, the feature extraction component will hypothesize the presence of a tire and wheel. This symbolic feature can then be used in conjunction with other symbolic features in the model matching process described in the previous section.

In addition to providing initial feature and relational
information, which is a data driven process, the feature extraction component can also be used in a model driven manner by the model matching component. During the parsing of the classification tree, the model matching procedure may encounter the presence of a symbolic feature in the tree which is not present in the initial symbolic feature list provided by the feature extraction component. The model matching process can request the feature extraction procedure to reanalyze a specific portion of the image in search of that particular feature. The parameter sets and thresholds used in this instance can be more relaxed than during the initial image processing since only a portion of the image is being processed and the presence of a specific feature is being sought. If after several attempts to find the feature, the extraction process is not successful, the feature extraction component will abandon the search and the model matching process will be informed of the failure. Otherwise, the presence and location of the desired feature will be returned to the model matching component.

3.7 Recognition-Learning Process

The vision-learning cycle of the TRIPLE system consists of three main steps: model creation/refinement, target aggregation, and target recognition. A brief review of each of these stages as well as how they interact in the target recognition process is provided in the rest of this section.

The user provides an initial collection of background knowledge and a set of training examples which represent the targets needed for recognition purposes. The domain background knowledge is stored for use by all components of the TRIPLE system. The sets of target examples are given to the EBL system for characterization. During the training phase, the EBL component must ensure that the target models which are created are unique. If they fail to meet this requirement, the user should be notified and must then provide additional background knowledge or more detailed target examples. The model schemata are stored in the knowledge base for subsequent use. In addition, a copy of each schema is then sent to SCC for inclusion in the new system classification tree. Once SCC has received all the models from the training set, it creates the initial classification tree. Finally, the tree is sent to the model matching component for recognition of unknown targets.

After the initial training has been completed, the recognition component uses the classification tree to recognize each target processed by the TRIPLE system. If a target has already been modeled, it will be recognized, the confidence will be computed, and this information will be output to the user. The types of recognition produced by the TRIPLE system include complete recognition, incomplete recognition, and recognition in occluded scenes.

If the model exists, but needs to be refined, the recognition component will send the image data to the EBL system which will identify the incorrect data, update the model, and send this information to the SCC system. SCC will then make the necessary modifications to the classification tree and return the updated tree back to the recognition element, where this process begins again.

If the model does not exist, the image data is sent to the EBL system for construction of a new model. This procedure is subject to verification of the usefulness of the observed image data. Once the EBL system has built the new model schema, it is passed to the SCC component. In this case, the entire classification tree is rebuilt, in case the new model contains attributes which may lead to a more efficient tree representation. As before, the tree is then sent to the model matching component for use in subsequent identification activities.

4. EXAMPLE - VEHICLE RECOGNITION

In order to fully illustrate the interactions among the components of the TRIPLE system, this section provides a simple example using military vehicles as objects. The vehicles which have been chosen are examples of Soviet vehicles which would likely be encountered by an autonomous vehicle during a reconnaissance or surveillance mission. This group includes armoured personnel carriers, reconnaissance vehicles, and cargo transport vehicles. Complex objects such as tanks and self-propelled guns have been avoided to simplify this example. In addition, the vehicles which have been selected will be treated as 2D objects in terms of the features which are used to classify them. However, in practice, the TRIPLE system is being implemented using a 3D object modeling system which incorporates salient object features and relationships between those features.

Figure 2 shows the system training set for the selected Soviet vehicles used in this example. The set includes an armoured personnel carrier (BTR-70), a reconnaissance vehicle (BRDM-1), and a cargo truck (MAZ-502). The feature set defined for this example are simple features such as object length, height, wheelbase, number of wheels, and so on. Each of these features can be reliably extracted from an image containing that target. Note that not all features are defined on all objects; the cargo truck does not have an armament feature but does include a load height feature which specifies the height of the cargo platform.

The EBL component of the TRIPLE system takes the above training set and selects the relevant target features using the background knowledge for this particular target domain. In this example, the rules may contain information such as:

- Vehicle length, height, and wheelbase are always used in characterizing a target.

If a vehicle has armaments, they are useful in characterizing that target.

The set of relevant features for the Soviet vehicles is shown in Figure 3. Since the armoured personnel carrier and the reconnaissance vehicle have armaments, that feature has been included in the target models in addition to length, height, and wheelbase. As mentioned in section 3, the remaining target attributes are retained in the target model database in case the system needs to use them in the future.

Once the EBL characterization for each target has been obtained, the SCC component creates the classification tree which will be used by the knowledge-based matching component during the subsequent recognition procedure. SCC segregates the targets based on the conceptual simplicity of the target classes. The classification tree for this example is indicated in Figure 4. Because the cargo truck contains no armaments, it is conceptually different from the other two vehicles and is placed in a separate branch of the classification tree. In the right branch of the tree, the length feature of those targets is determined to be the most useful in distinguishing between the recon vehicle and the personnel carrier. The resulting classification tree is sent to the matching component for use in recognizing these three targets.

The matching component continues to use the classification tree until it encounters a target which cannot
Figure 2: System training set for Soviet vehicles

Figure 3: EBL-selected relevant target features

Figure 4: SCC-generated target classification tree
be processed by the tree information. This situation was fully described in section 3.5. When this failure occurs, the unknown target is sent to the EBL component for complete characterization. This event signals the beginning of the learning cycle. Figure 5 shows a target which cannot be processed by the classification tree due to differences in the each of the three target features. The vehicle is assumed to be a new target and the feature list is sent to the EBL system. Figure 6 displays the results of the relevant feature selection by the EBL component. In the process of selecting relevant features, EBL has determined that the number of wheels is a useful feature when distinguishing the new vehicle from the MAZ-520 cargo truck which is also present. This feature is then marked in the MAZ-520 feature list as relevant. The resulting classification tree produced by the SCC component is shown in Figure 7. The number-of-wheels feature is now used in the left branch of the tree to distinguish between the MAZ-520 cargo truck and the new unknown vehicle.

This simple example has shown how the TRIPLE system can achieve target model acquisition and model refinement by explaining feature significance and using those features to efficiently recognize future instances of a target.

![Unknown Vehicle](image)

**Figure 5:** Unknown vehicle which triggers the learning cycle

| Unknown Vehicle |
|-----------------|-----------------|
| Length          | 7.35 m          |
| Cab Height      | 2.68 m          |
| Wheelbase       | 3.5 m           |
| Number of Wheels| 6               |

**Figure 6:** EBL-selected relevant target features

![Classification Tree](image)

**Figure 7:** New SCC-generated target classification tree
5. CONCLUSIONS AND FUTURE WORK

We have shown the usefulness of integrating machine learning with target recognition to create a system which adapts its representation of the target domain in order to operate effectively in an unconstrained, outdoor environment. Present target recognition systems do not have this capability. Using the characterization ability of explanation-based learning and the efficient classification techniques of conceptual clustering, the TRIPLE system provides automatic knowledge-base acquisition and refinement. The system is robust in several respects: it can acquire a target model using a single example of that target; it uses only the most relevant features during subsequent recognition stages; and target model refinements are facilitated through the use of the EBL component.

Presently, we are working on an implementation which recognizes complex, 3D targets such as cars, trucks, vans, etc. This system uses 3D object models which consist of object features and relationships among the features. In addition, we will evaluate the performance of the TRIPLE system when subjected to problems such as occlusion, image noise, etc.

REFERENCES

Image Understanding Workshop

Proceedings of a Workshop
Held at
Cambridge, Massachusetts

April 6-8, 1988

Volume II

Sponsored by:
Defense Advanced Research Projects Agency
Information Science and Technology Office

This document contains copies of reports prepared for the DARPA Image Understanding Workshop. Included are results from both the basic and strategic computing programs within DARPA/ISTO sponsored projects.

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the United States Government.