ERDAS IMAGINE®
Expert Classifier
Overview

ERDAS®, Inc.
Atlanta, Georgia
# Table Of Contents

Table Of Contents .......................................................... iii  
Product Overview .............................................................. 1  
Knowledge Engineer .......................................................... 2  
  Knowledge Base Editor Menus ....................................... 2  
  File ............................................................................. 2  
  Edit ............................................................................. 2  
  Evaluate ................................................................. 3  
  Help ................................................................. 3  
  Knowledge Base Component List ................................ 5  
  Decision Tree .......................................................... 5  
    Color Scheme ......................................................... 6  
  Tools ........................................................................ 6  
oters Boxes .................................................................... 7  
  Hypothesis Properties .................................................... 7  
  Rule Properties ............................................................ 7  
  Logic of Nodes ............................................................. 8  
  true ........................................................................... 8  
  False ........................................................................... 9  
  Confidences .................................................................. 9  
Variable Properties ............................................................ 11  
  Work Files ............................................................... 11  
  Prompts ..................................................................... 11  
  Optional Variables ......................................................... 11  
  Raster Variables .......................................................... 11  
    Raster Variables - Imagery ........................................ 12  
    Raster Variables - Feature .......................................... 12  
    Raster Variables - Graphic Model ............................... 12  
    Raster Variables - Program ......................................... 14  
  Scalar Variables .......................................................... 14  
    Scalar Variables - Value ............................................. 14  
    Scalar Variables - Graphic Model ............................... 15  
    Scalar Variables - Program ......................................... 15  
  Summary of Variables ..................................................... 16  
Evaluation ...................................................................... 17  
  Test Classification ......................................................... 17  
  Classification Pathway Cursor ........................................ 17  
Knowledge Classifier User Interface ............................... 18  
  Knowledge Base .......................................................... 18  
  Knowledge Classifier User Interface - Classes .............. 18  
  Inputs ........................................................................ 19  
  Missing Files ............................................................. 20  
  Outputs .................................................................... 20  
    Output Files ............................................................ 21  
    Output Area ............................................................ 21  
    Output Cell Size ....................................................... 21  
    Output Map Projection .............................................. 22  
Using Expert Classifier ...................................................... iii
Product Overview

This document is provided as an overview of the functionality available in the ERDAS IMAGINE Expert Classifier. For detailed information, please refer to the ERDAS Field Guide and the ERDAS IMAGINE Tour Guides.

The expert classification software provides a rules-based approach to multispectral image classification, post-classification refinement, and GIS modeling. In essence, an expert classification system is a hierarchy of rules, or a decision tree, that describes the conditions under which a set of low level constituent information gets abstracted into a set of high level informational classes. The constituent information consists of user-defined variables and includes raster imagery, vector coverages, spatial models, external programs, and simple scalars.

A rule is a conditional statement, or list of conditional statements, about the variable's data values and/or attributes that determine an informational component or hypothesis. Multiple rules and hypotheses can be linked together into a hierarchy that ultimately describes a final set of target informational classes or terminal hypotheses. Confidence values associated with each condition are also combined to provide a confidence image corresponding to the final output classified image.

The Expert Classifier is composed of two parts: the Knowledge Engineer and the Knowledge Classifier. The Knowledge Engineer provides the interface for an expert with first-hand knowledge of the data and the application to identify the variables, rules, and output classes of interest and create the hierarchical decision tree. The Knowledge Classifier provides an interface for a non-expert to apply the knowledge base and create the output classification.

The Expert Classifier is added to the top-level IMAGINE Classification menu as illustrated below.
**Knowledge Engineer**

The Knowledge Engineer is a separate application program invoked from the Classification dialog. The interface follows a document editing paradigm where the document is the decision tree. Below is the empty Knowledge Base Editor, before a Knowledge Base file (.ckb) has been opened or edits have been made on a new file.

![Knowledge Base Editor](image)

**Knowledge Base Editor**

**Menus**

**File**

The File menu provides access to knowledgebase file Open, Save, and Print functions. Here, you can also create a New file, rename an open file (Save As...), or undo all edits since your last Save (Revert to Saved).

![File Menu](image)

**Edit**

The Edit menu provides additional access to functions more readily available from the toolbar. However, it provides a couple of powerful globally acting functions.

The **Delete All Disabled** option allows you to remove all currently disabled branches from the knowledgebase. The **Clear All Work Files...** option allows you to remove all work files associated with the current session (See “Work Files” on page 11.).
After a knowledgebase is opened or created, a graphical representation of the decision tree is presented in the editing window as shown below.

Evaluate
The Evaluate menu provides options to evaluate the effectiveness of your knowledge base. Details are provided under Product Overview on page 17.

Help
The help menu provides access to on-line help.
In the upper left corner of the editing window is an overview of the entire decision tree with a green box indicating the position within the knowledgebase of the currently displayed portion of the decision tree. This box can be dragged to change the view of the decision tree graphic in the display window on the right. The branch containing the currently selected hypothesis, rule, or condition is highlighted in the overview.

Below the overview is the “Product Overview”. Each list contains all of the elements of the selected type within the knowledgebase. To view the properties of a selected element, click on the Properties tool. To view the next occurrence of a hypothesis or a rule, click on the Find Next tool.

The decision tree occupies the right two-thirds of the editing window. Each branch of the decision tree consists of nodes depicted as boxes connected by lines indicating the logical relationship of the decision hierarchy. On the left side of the decision tree are the hypotheses representing the final output classes. Moving to the right from the output classes are the rules used to define the hypotheses. Each individual rule is composed of a list of conditions all of which must be met for the rule to be true. An individual rule is satisfied by ANDing its list of conditions. When a logical OR is needed one simply sets two or more rules to share a hypothesis. In this way the hypothesis is set to true if any of the rules connected to it are true. A condition may contain variables, scalars, imagery, coverages, output from external programs, etc.
Knowledge Base Component List

The Knowledge Base Component List serves as the organization center for the knowledge base. All hypotheses, rules and variables can be viewed, accessed, and edited in the Knowledge Base Component List, shown below, and new ones can be created. There are folder tabs for the three types of components, each with a scrolling list of its existing Hypotheses, Rules or Variables. A component selected by clicking on it in a scrolling list is highlighted in the graphic decision tree and brought into view within the decision tree display window. In cases where a component has been used multiple times in the Knowledge Base, the additional instances of use can be located with the Find Next tool.

Additional tools are provided for the Knowledge Base Component List. They are used to create new components, display the properties of the selected component, delete a component, or find the next occurrence of the selected component.

Decision Tree

The decision tree grows in depth when the hypothesis of one rule is referred to by a condition of another rule. The terminal hypotheses of the decision tree represent the final classes of interest. Intermediate hypotheses may also be flagged as being a class of interest. This may occur when there is an association between classes.

The example below represents a single branch of a decision tree depicting a hypothesis, its rule, and conditions.

In this example the rule, Gentle Southern Slope, determines the hypothesis, Good Location. The rule has four conditions depicted on the right side, all of which must be satisfied for the rule to be true.
The rule may be split however if either Southern or Gentle slope will define the Good Location hypothesis. While both conditions must still be true to fire a rule, only one rule must be true to satisfy the hypothesis.

Color Scheme

The colors of the various components of the decision tree have the following meaning.

- Green - a hypothesis. Components that represents an output class have a color patch showing the color assigned to the class. Components that represent an intermediate hypothesis do not have the color patch. These hypotheses may be used multiple times within a knowledge base but they do not represent an output class.
- Yellow - a rule. One or more rules define a hypothesis. Any satisfied rule makes it’s hypothesis true.
- Cyan - a condition. One or more conditions define a rule. All of a rule’s conditions must be true for a rule to be satisfied.

Tools

To facilitate placement of Knowledge Engineer components in the decision tree diagram, the following tools have been provided. Select, Cut, Copy, Paste, Lock, and Help are exactly like those found in other IMAGINE applications. Hypothesis, Rule, Activate, and Deactivate are specific to Knowledge Engineer.

With the Hypothesis tool, you can place a new hypothesis in the decision tree window.
The Rule tool is used to add a new rule. A new rule may be attached to an existing hypothesis by dragging and dropping the rule on the hypothesis.

The Disable tool is used to disable sections of the decision tree during testing or classification. It is particularly useful in assessing the effects of removing a portion or portions of the knowledge base without actually deleting the section. The tool deactivates the section of the tree beginning with the component selected and continues to include all those to the right of that location on the decision tree. Structures to the left of that point that are invalidated by deactivation of one or more of its constituent components are also deactivated. Deactivated portions of the decision tree appear "grayed out".

The Enable tool reactivates sections of the decision tree that have been deactivated with the Deactivate tool. The tool activates the entire contiguous deactivated section of the decision tree that is clicked on.

Properties Boxes
There are three properties boxes from which you may edit the properties of the hypotheses, rules, or variables of your knowledgebase. There are several ways in which a property box can be opened.

- Knowledge Engineer edit window - double-click on a hypothesis or a rule (variables are not accessible from the edit window)
- Knowledge Base Components List
  - double-click any hypothesis, rule, or variable
  - select a hypothesis, rule, or variable and then click the Properties tool
  - click the New tool

Hypothesis Properties
The Name of the hypothesis is displayed or defined there and a checkbox allows the hypothesis to be designated a final output class. If a hypothesis is designated as a final output class, a Color for the class or hypothesis can be specified or Grayscale may be selected. Selecting Grayscale gives it a gray value as part of a level slice based on the number of hypotheses designated as Grayscale. The larger the number of hypotheses, the greater the number of slices.

Rule Properties
The Rule Properties dialog allows each of the variables, or conditions of the rule to be defined. The rule is identified by a name string.
A method of determining Confidence for the rule’s conditions must be selected. If the **Compute from Conditions** option is selected, the confidence values for each rule are computed by the Knowledge Classifier from the confidence of each condition associated with the rule. If the **Specify** option is selected, you must enter the confidence value to be assigned to the rule.

The list of conditions is presented in a cell array where each condition is defined by a variable, a relation, a value, and a confidence.

A variable, relation, or value in the list of conditions can be changed by selecting its cell in the Cellarray and choosing from the popup list. The confidence entries can be edited by clicking in the appropriate cell in the Cellarray.

**Logic of Nodes**

The logic of a node depends upon how its components are evaluated. A node is an object that contains properties for a hypothesis, a rule, or a condition. Nodes are displayed as rectangles in the knowledgebase window. Logical connections between nodes are displayed as lines between the rectangles. Nodes are evaluated as True or False.

A condition compares a pixel to a real value. A rule assigns a meaning to one or more conditions. A hypothesis forms a classification based on the truth of one or more rules.

**True**

In the following example, In order for the Hypothesis to be evaluated as true, either Condition 1 and Condition 2 must be true or Condition 3 and Condition 4 must be true.
False

In the following example, In order for the Hypothesis to be evaluated as false, either Condition 1 or Condition 2 must be false and Condition 3 or Condition 4 must be false.

Confidences

A confidence value is a probability that the node is significant. Confidence values for rules may be explicitly assigned by the knowledge engineer or they may be computed by the Expert Classifier from the confidence values of the conditions. Confidence values for conditions are always assigned by the knowledge engineer. Computed confidences do not override the knowledge engineer-assigned confidences. The equations below shows how confidences are computed. In the case of multiple ORs, the highest confidence is taken.

Along branch

\[ \prod_{i=0}^{n} c_{i} \]

Across Branch (multiple ANDs)

\[ 1 - \left( \prod_{i=0}^{n} (1 - c_{i}) \right) \]

Where:

- \( c_{i} \) = the confidence value of node (i)
- \( \prod c_{i} \) = the product of the confidence values
- \( 1 - c_{i} \) = the error probability
- \( n \) = the number of nodes in a branch

The error probability is used in the calculation of across-branch confidences to avoid diminishing the cumulative confidence since all conditions must be true for the rule to be true.
This figure defines the across branch and along branch terms and shows how confidences are calculated from given condition confidences. In this example, there are no explicitly assigned rule confidences; all are calculated using the equations above.
Variable Properties

The Variable Properties dialog provides for the definition of the variable objects to be used in the rules' conditions. It is invoked from the Knowledge Engineer menus or toolbar, or by double-clicking the Variable name in the Rule Properties dialog. Double-clicking in an empty cell in Rule Properties opens a Variable Properties dialog with a blank name field. All variables are defined by a variable name.

There are two types of supported variables, raster and scalar, one of which is selected from the Variable Type popup list. Variables also have an associated data type: integer, float, or boolean, which is set in the Data Type popup list.

Work Files

Program and Graphic Model variables create semi-permanent Work Files. If a work files exist for a variable at the time the model or program is run, it will be used. Keeping existing work files makes subsequent executions of Knowledge Classifier run faster. However, if you have changed the knowledgebase, you may have invalidated the work files and therefore you may want to clear the work files and recompute.

If you want temporary output files to be regenerated, you must clear the work file. There are two ways to clear a work file. To clear only the work files associated with one variable, click the Clear Work File button on the properties dialog for that variable. If you wish to clear all work files, use the Edit | Clear all Work Files... option on the main menu bar.

If a window is used by a program or graphic model, the work file is automatically deleted if the window is modified.

Prompts

A note space is provided for variables that are left undefined. In this space, the knowledge engineer may leave detailed information regarding a variable that must be defined by the end user. At run time, the Knowledge Classifier will display this information to the user.

Optional Variables

Any variable may be designated as optional by clicking the Optional checkbox. If such a variable exists at run time, it will be used. If it does not exist, the classification will proceed without it. If a variable has not been marked as optional, then the classifier will not work in its absence.

Raster Variables

Raster variables may be defined by imagery, feature layers (including vector layers), graphic spatial models, or by running other programs. Selecting the type of raster variable is accomplished by depressing one of the raster variable radio buttons. The contents of the area below the radio buttons change depending on the radio button selection. The following paragraphs and figures show examples of the Variable Properties dialog for raster variables, one for each of the four types of raster variables.
**Raster Variables - Imagery**

Imagery variables consist of any raster data format supported by IMAGINE. The data may be single or multi-layer, continuous or thematic and may be resampled or calibrated to different map projections. The imagery variable definition consists of the filename, the layer name, and the attribute to use. In addition to the layer attributes, the designation, "cell value", may be set to use the actual pixel value in the condition as shown below.

![Variable Properties](image.png)

Raster imagery variables also support an undefined status indicated by the **Leave Undefined (Prompt Analyst)** checkbox. If this checkbox is selected, the other parts are disabled and you will be prompted for the image filename in the Knowledge Classifier user interface. See Product Overview on page 19.

**Raster Variables - Feature**

Variables defined as feature layers may be annotation layers, shapefiles, SDE vectors, or ArcInfo coverages. In either case, the variable is defined by its filename. The feature variable dialog is shown below. The Knowledge Engineer restricts the use of feature variables as inputs to models, i.e. all vector rasterization occurs in the Modeler and not the Knowledge Classifier. Feature variables also support the **Leave Undefined** option to prompt the user of the Knowledge Classifier for the filename.

**Raster Variables - Graphic Model**

Variables may be defined as models. The Knowledge Engineer requires the models to be graphical model (.gmd) files. It is assumed that these graphical models are also designed and created by the Knowledge Engineer user. Support for model variables extends the capability of the Expert Classifier dramatically.
Each graphic model is designed to have one or more inputs and one output. These inputs and output may be other variables defined in the knowledge base. The model definition consists of the model name, the output variable name together with the layer name and attribute to use, and the input variables to the model. The Graphic Model may be created with the "prompt user" flag set for the inputs and outputs so that they can be filled in by the Knowledge Engineer with other variables. See Product Overview on page 19.

For convenience, an Edit Model... button exists to start Model Maker with the specified graphical model file (.gmd). If the model has not yet been defined, Model Maker is started in new-model mode.
Integration of the Modeler is a fundamental feature providing the Expert Classifier with all the non-point operations, such as neighborhood functions, clump, and search. The Knowledge Classifier itself is a pixel processing algorithm that is unable to perform non-point operations.

**Raster Variables - Program**

Another class of variable supported by the Expert Classifier is an external program, as illustrated in the figure below. This allows for the integration of other IMAGINE command line programs, such as the classifiers. However, these external programs are not restricted to IMAGINE programs.

The Expert Classifier can also run any other program provided it can be run in a non-interactive mode, that it accepts command line arguments, and one of its command line arguments is the output filename.

To create a program variable you must provide the program name, the number of arguments, and which argument is the output filename. The cellarray must be populated with valid arguments. Like the model variables, the program variable may use other variables in the knowledge base as inputs to the program. The Var column in the cellarray indicates whether the argument is a knowledge base variable or not. The program variable assumes the variable name for referencing in the rest of the knowledge base.

**Scalar Variables**

Scalar variables may be defined with an explicit value or defined as the output from a model or external program. When a variable type of Scalar is selected, you have three radio button options: **Value**, **Graphic Model**, or **Program**. The latter two options are identical to the raster variable type described above except that the models and programs output scalar values instead of raster images.

**Scalar Variables - Value**

You may enter an explicit scalar value or leave the variable undefined, to be supplied by the user of the Knowledge Classifier.
Scalar Variables - Graphic Model

This type of variable is identical to Product Overview on page 12 except that the output is scalar not raster.

Scalar Variables - Program

This type of variable is identical to Product Overview on page 14 except that the output is scalar not raster.
Summary of Variables

The following table summarizes the types of variables and their properties.

### Variables and Properties

<table>
<thead>
<tr>
<th>Variable</th>
<th>Inputs</th>
<th>Outputs</th>
<th>Left Undefined</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scalar</td>
<td>Model, Program, Value</td>
<td>Scalar</td>
<td>Supported</td>
</tr>
<tr>
<td>Imagery</td>
<td>Raster/Layer/Attribute</td>
<td>Raster</td>
<td>Supported</td>
</tr>
<tr>
<td>Feature</td>
<td>Coverage, Annotation</td>
<td>Used in Models</td>
<td>Supported</td>
</tr>
<tr>
<td>Models</td>
<td>Variables, Modeler Objects</td>
<td>Raster, Scalar</td>
<td>No</td>
</tr>
<tr>
<td>Programs</td>
<td>Variables, Arguments</td>
<td>Raster, Scalar</td>
<td>No</td>
</tr>
</tbody>
</table>
Evaluation

The task of creating a useful, well constructed knowledge base requires numerous iterations of trial, evaluation, and refinement. To facilitate this process two options are provided under the Evaluate menu of the Knowledge Engineer.

Test Classification

Performing a test classification is an option under the Evaluate menu. Invoking this menu starts the Knowledge Classifier user interface with the current state of the knowledge base. If the Knowledge Classifier user interface is already running, then commands are passed to load the current knowledge base. Temporary files for the current knowledge base, and the output files are created and used for the test classification. When the test classification is complete, you can then evaluate the results using the Classification Pathway Cursor.

Classification Pathway Cursor

Following the classification of a test image using the knowledge base, you may wish to evaluate the results and refine the knowledge base. The Classification Pathway Cursor is started from the Evaluate menu and displays the dialog shown below. This feature behaves similar to the Viewer Inquire Cursor.

It allows you to interactively move a cross-hair cursor over a viewer containing the classified output file and read the class name, class value, and confidence value for the pixel under the cross-hair. Additionally, the path in the decision tree graphic in the Knowledge Engineer is highlighted to indicate the series of rules that were satisfied to classify the pixel. This gives you visual and quantitative feedback to evaluate the results.

To accomplish this graphical interaction between the knowledge base and the output classification it is important to maintain a link between the current state of the knowledge base and the classified file. If they are not forced to be synchronous then the classification pathways may become invalid. This link is maintained by version numbers which reflect a state of the knowledge base and the classification. Whenever a knowledge base is modified from its last saved state, its version number is incremented.

This version number is also written to the classified output file. When the Classification Pathway Cursor is started, it searches for a Viewer containing an expert classified file which matches the version number of the current knowledge base and links to that Viewer. If no Viewer can be found, the Knowledge Engineer looks in the list of temporary test classifications it has created to attempt to match version numbers. If a match is found a Viewer is started and the classified output file is loaded. If no match is found you are given an error message that a classification output file for the current state of the knowledge base cannot be found.

To maintain the synchronous link between knowledge base and classification it is also necessary to prevent any knowledge base modifications from being performed while in the Classification Pathway Cursor mode.
Knowledge Classifier User Interface

The Knowledge Classifier is itself composed of two parts, an application providing a user interface and a command line executable. The user interface application allows you to input a limited set of parameters to control the use of the knowledge base. The user interface is designed to lead you through pages of input parameters. On the right side of the dialog, the Next and Previous buttons move through the pages forward and backward. You are only allowed to move to the next page if parameters on the current page are valid. However, you may always back up to view or change the contents of previous pages.

Other buttons on the right side include OK and Batch buttons which become enabled only if all the input parameters are valid or optional, and a Save As button to save a new knowledge base file containing your inputs on the Knowledge Classifier.

By design, there is not a way for the user of the Knowledge Classifier to Save, and overwrite the input knowledge base file.

Knowledge Base

The Knowledge Base page is initially displayed when the program begins. You must specify the knowledge base file before you can continue. The Knowledge Base page is shown in the figure below.

Knowledge Classifier User Interface - Classes

After you select the knowledge base file, the interface automatically advances to the Classes page as shown below. This is where you may select a subset of classes defined in the knowledge base. The default selection of classes is all of the classes in the knowledge base. Available classes are listed on the left and selected classes are on the right.
Inputs

The Inputs page is the location in the interface where you must fill in all the variables which were marked as **Leave Undefined** in the Variable Properties dialog. This page, shown below, contains a scrolling list of options that are dynamically created by the application. The type of variables eligible for being prompted for in the Knowledge Classifier user interface are scalars, imagery, and features.

The option list created depends on the type of variable. For imagery, the filename, layer name, and attribute or cell value designation must be filled in by you. For feature variables, the feature filename is required. For scalar inputs, the values may be integer, float, or boolean. Defaults for these values may be specified by the Knowledge Engineer user and will be displayed in the option lists as defaults. If no undefined variables are present in the knowledge base then this page is not displayed. Once all the inputs are specified, the **Next** button becomes enabled and you may proceed.
Missing Files

The Missing Files page, shown below, is displayed only when files specified in the knowledge base are not found. When the knowledge base is opened, the existence of all the files is tested and, if a file does not exist, it is put in the Missing Files page. This page is also a scrolling list and contains the names of the missing files and supplies the interface for specifying valid replacement files. The check is performed on all images, features, models, and programs defined in the knowledge base but not on any hardcoded inputs to models or arguments to programs.

Outputs

The Output page, shown below, is the final page in the Knowledge Classifier startup sequence and must be filled in with the output information. Once all the output parameters are specified the OK and Batch buttons are enabled and you may perform the classification.
Output Files
You specify the number of best classes per pixel to produce. This determines the number of layers in the output classified image, the confidence image, and the feedback image files. The top layer contains the best (highest confidence) classification, the second layer the second best and so on. In the case of two or more classes having the same confidence, they are placed in layers in the same order they appear in the decision tree. You must also indicate whether to produce a confidence file and/or a feedback file. These files are used to evaluate the classification results. See Product Overview on page 17. If you elect to produce an optional file, then the filename must be provided.

Output Area
For the output area specification, you may select either Window or AOI from the popup list on the Outputs page. If Window is selected then the Set button brings up the Set Window dialog shown below.

The window may be set as the Union or Intersection of all input variables. Additionally, you may specify a window by entering the window corners or to get the window from the Inquire Box.

If the AOI (Area Of Interest) option is selected, the Set button opens a standard Set AOI dialog which is used throughout IMAGINE. The window defaults to the bounding box of the AOI.

The default Area, if you do nothing at all, is a window based on the Union of all inputs.

Output Cell Size
The output cell size may be set from the cell Size popup list. The options are Minimum, Maximum, or Specify.... Minimum is the smallest pixel size from all of the input rasters; maximum is the largest. Specify opens a dialog to set the cell size and units as shown below.
Output Map Projection

As mentioned earlier, the Knowledge Classifier supports input images of different map projections. If the knowledge base contains images of different projections then the Select Image for Map Projection popup list becomes enabled and you must select which image from the list to use to provide an output map projection. All images will then get reprojected to this base image's projection.

⚠️ Every input image must have a projection (or calibration) or every input image must have no projection. There cannot be a mix of images with and without a projection.

To simplify the interface, the assumption is made that at least one of the input images is in the desired output projection. You cannot set an arbitrary output projection using the Projection Chooser tool.
Knowledge Classifier Process

The Knowledge Classifier user interface checks every time you interact with the application to see if all the required information has been provided to run the classification. Once all the information is specified the OK and Batch buttons become enabled and the Knowledge Classifier classification process can be run.

The Knowledge Classifier Process may also be run independent of the user interface. The process has a command line interface to specify the knowledge base file, which classes to use or not use, and the definitions of the undefined variables. Additionally, an ACSII text file may be provided on the command line to specify these same arguments. The command line syntax for the Knowledge Classifier process is:

```
inference_engine (<knowtree>, <clasfile>,<conffile>,
    -use_class <"class 1", "class 2" ...>
    -ignore_class <"class 1", "class 2"...>
    -define_var <"name", value>
    -option_file <"name">)
```

This command line interface to the Knowledge Classifier process gives ERDAS IMAGINE Developer’s Toolkit users and third party developers the ability for integration and multiple image processing.

Preferences

The Expert Classifier consults the IMAGINE preference files to provide defaults and processing options. To avoid creating redundant preferences, the Expert Classifier uses preferences in the Spatial Modeler category where applicable. The following table outlines the preferences and categories used by the Expert Classifier. The As Defaults column indicates whether the option is exposed to you and could potentially be overridden.

<table>
<thead>
<tr>
<th>Preference</th>
<th>Category</th>
<th>As Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Window Rule</td>
<td>Spatial Modeler</td>
<td>X</td>
</tr>
<tr>
<td>Cell Size Rule</td>
<td>Spatial Modeler</td>
<td>X</td>
</tr>
<tr>
<td>Thematic Interpolation</td>
<td>Spatial Modeler</td>
<td></td>
</tr>
<tr>
<td>Continuous Interpolation</td>
<td>Spatial Modeler</td>
<td></td>
</tr>
<tr>
<td>Grid Sampling for Reprojection</td>
<td>Raster Processing</td>
<td></td>
</tr>
<tr>
<td>RMS Tolerance</td>
<td>Raster Processing</td>
<td></td>
</tr>
<tr>
<td>Confidence Computation Single Rule</td>
<td>Knowledge Engineer</td>
<td>X</td>
</tr>
<tr>
<td>Confidence Computation Multiple Rule</td>
<td>Knowledge Engineer</td>
<td>X</td>
</tr>
<tr>
<td>Number of Best Classes / Pixel</td>
<td>Knowledge Engineer</td>
<td>X</td>
</tr>
<tr>
<td>Create Confidence Image</td>
<td>Knowledge Engineer</td>
<td>X</td>
</tr>
<tr>
<td>Create Pathway Feedback Image</td>
<td>Knowledge Engineer</td>
<td>X</td>
</tr>
<tr>
<td>Handling Existing Output Files</td>
<td>Knowledge Engineer</td>
<td></td>
</tr>
</tbody>
</table>