

# Software Development Breaking New Ground

## Configuration Replaces Programming

Cost for hardware components in machine vision is ever decreasing, but the expenses for software development remain at a very high level. Consequently there is a demand for new concepts in software development: reducing time and effort by abandoning programming in high-level languages like C++ and using more convenient, configuration-type development tools for configuring vision applications instead. Matrox Design Assistant is an answer for this request: It relieves programmers of a significant workload and enables even non-programmers to realize machine vision applications.

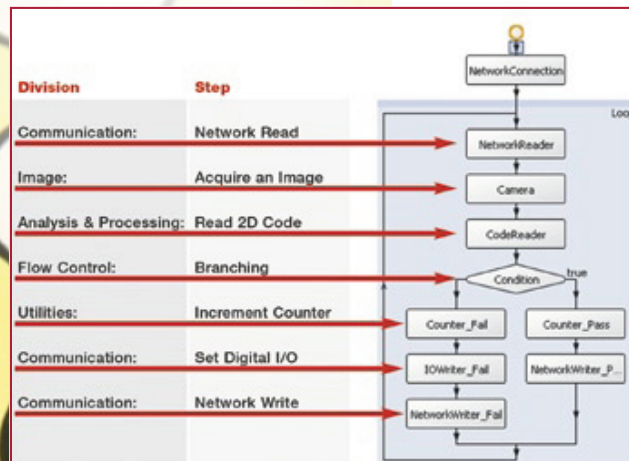


Fig. 1: The flowchart is build from single steps

Up to now, software solutions for graphical or visual application software development are facing three main challenges: to keep the workflow clear and simple even for complex tasks, to allow the generation of a customized user interface and to have the option for a fast and easy extension of the existing functionality. The Matrox Design Assistant overcomes these limitations and opens up new ways of software development. This development tool allows easy and quick configuration and deployment of machine vision applications including a user interface for the end customer. This is done with two main tools: the sequence control is implemented by a graphical flowchart and the user-interface is designed with an integrated HTML editor.

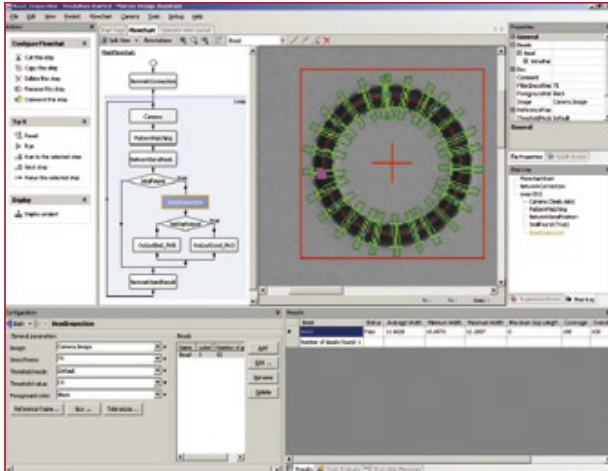
### Flowchart Steps

The flowchart is used to describe the logical workflow of a vision application. It is build by adding and interconnecting single steps, which will be executed sequentially during runtime. The Design Assistant offers a large selection of configurable steps, which are classified into five function groups: Image, Analysis & Processing, Flow Control, Communication and Utilities.

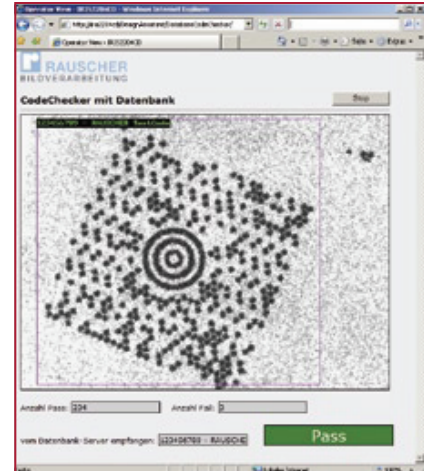
The “Image” group offers image acquisition and restoring images from disk. “Analysis & Processing” comprehends modules for a range of different vision tasks (see box). All algorithms originate from the field-proven Matrox Imaging Library (MIL) and guarantee performance, robustness and accuracy. “Flow Control” provides the steps that are necessary to control the execution of the flowchart. Decision making and branching, entering and leaving sub-flowcharts as well as pause, resume and halt the execution of the main flowchart. “Communication” summarizes functions for communicating with external hardware in the field and to evaluate external events in the flowchart: digital User-I/O, serial RS-232, Ethernet, mouse- and keyboard inputs and industry protocols like Ethernet/IP and Modbus. The “Utilities” group finally holds counters and variables for storing data of any kind. These steps can also be used to generate log files and to create and evaluate timestamps.

### Well-arranged Design

The flowchart is set up with the described steps and is executed sequentially in the main loop (fig. 1). Already during building the flowchart every single step can be configured and tested interactively. The configuration is done via simple configuration masks and a short online help will show an explanation of the most important parameters. The configuration mask does not overlap the flowchart or the im-



◀ Fig. 2: Clear layout: flowchart, camera and configuration panes are visible on a single monitor



▶ Fig. 3: The Operator-View during runtime can be viewed via the Internet Explorer

age of the camera. So the whole design process stays clear and well structured even when dealing with larger and more complex flowcharts (fig. 2).

The Design Assistant offers 38 steps for a fast and convenient solution for all kinds of different vision applications – from simple feature presence/absence checks up to very complex measurement tasks. The algorithms from the Matrox Imaging Library, combined with the flexible communication within the flowchart, with the operator view and external hardware builds a strong platform for all applications.

### Extensions Unlimited

If a requirement comes up, which can not be solved with the existing tool-set of the Design Assistant, a Custom Step SDK can be used for building own steps and for integrating them seamlessly. These custom steps can be used – exactly like the ones that come with the Design Assistant – without programming knowl-

edge. The own steps are programmed with Microsoft C#; developers can optionally use the full functionality of the Matrox Imaging Library. Examples for customer's steps are a FTP client for posting defect images on a special FTP site, a parser for configuration files and special calculation modules.

### Operator View Strictly Separated

Another great strength of the Design Assistant lies in the strict separation between the logical description of the vision application, the flowchart, and the user-interface presented to the end customer during runtime, the operator view. While the flowchart represents the internal logic, the operator view is the graphical user interface for the end customer for visualization and parameterization of the application (fig. 3). The operator view is generated within the Design Assistant with a graphical HTML editor, where all necessary elements are added and interconnected with the flowchart's in- and

outputs. Input- and output- fields, buttons, displays for the camera images and other graphical elements can be added. An operator view can be a website which can be seen during runtime from any remote PC over the network. Alternatively the operator view can be programmed for special requirements for example in Visual Basic .NET.

### Powerful Application Development

The Design Assistant is a powerful tool for the quick application development. It allows tackling and solving different vision applications without programming knowledge. In addition a vast amount of flexibility and possibility is offered to expand its functionality – even specialized, custom algorithms can be integrated.

The combination of such an innovative software tool with the highly integrated smart camera Matrox Iris GT, which features an Intel Atom 1.6GHz CPU in an IP67 rated casing is a great offer for OEMs, system integrators and end users. Vision applications can be developed quickly and without traditional and time-consuming code writing.

### Imaging Steps in Analysis & Processing

<b>Image Processing</b>	enhancing and transforming images in preparation for subsequent analysis
<b>Intensity Checker</b>	analyze an object using image intensity
<b>Blob Analysis</b>	identify, count, locate and measure basic features and objects (blobs)
<b>Edge Locator</b>	find objects by locating straight edges
<b>Metrology</b>	2D geometric dimensioning and tolerancing
<b>Code Reader</b>	locating and reading 1D, 2D and composite identification marks
<b>String Reader</b>	reading character strings that are engraved, etched, marked, printed, punched or stamped on surfaces
<b>Pattern Matching</b>	techniques to find an object using grey-levels
<b>Model Finder</b>	patented five techniques to find an object using geometric features (e.g., contours)
<b>Bead Inspection</b>	analyze beads

▶ **Author**  
Raoul Kimmelman,  
Head of Product Marketing



▶ **Contact**  
Rauscher GmbH, Olching, Germany  
Tel.: +49 8142 44841 0  
Fax: +49 8142 44841 90  
info@rauscher.de  
www.rauscher.de