

**A Milestone of Machine Vision Software**

**HALCON**

**Installation Guide**

**7.0**

All about installing and licensing HALCON, Version 7.0.4

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the publisher.

Edition 1	December 2003	(HALCON 7.0)
Edition 1a	July 2004	(HALCON 7.0.1)
Edition 1b	April 2005	(HALCON 7.0.2)

Copyright © 2003-2006 by MVTec Software GmbH, München, Germany



Microsoft, Windows, Windows NT, Windows 2000, Windows XP, Visual Basic, Visual C++, and Microsoft .NET are either trademarks or registered trademarks of Microsoft Corporation.  
Borland and Delphi are either trademarks or registered trademarks of Borland Software Corporation.  
Linux is a trademark of Linus Torvalds.  
RedHat is a trademark of RedHat, Inc.  
SuSE is a trademark of SuSE Linux AG.  
Silicon Graphics, SGI, and IRIX are either trademarks or registered trademarks of Silicon Graphics, Inc.  
Sun and Solaris are either trademarks or registered trademarks of Sun Microsystems.  
Tru64 and Alpha Server are either trademarks or registered trademarks of Compaq Computer Corporation.  
AMD and AMD Athlon are either trademarks or registered trademarks of Advanced Micro Devices, Inc.  
Intel, Pentium, and Itanium are either trademarks or registered trademarks of Intel Corporation.  
Philips is a trademark of Philips Electronics NV.  
TriMedia is a trademark of TriMedia Technologies, Inc.  
FLEXlm is a trademark of Macrovision.

All other nationally and internationally recognized trademarks and tradenames are hereby recognized.


More information about HALCON can be found at:

<http://www.mvtec.com/halcon/>

# About This Manual

The manual provides the necessary information to install HALCON and setup the licensing mechanism successfully. It is divided into the following chapters:

- **Introduction**  
A short overview of the different HALCON versions, available licensing schemes, and the system requirements.
- **Installing HALCON**  
How to install HALCON, either for the first time or in form of an update or upgrade.
- **All About HALCON Licenses**  
Detailed information about the different types of licenses and how to obtain and install them.
- **Troubleshooting**  
Possible problems and how to solve them.
- **Appendix**  
Details like the installed file structure, the relevant environment variables, and information for users of ActivisionTools.

For **further information** about HALCON, please consult the [Quick Guide](#), which gives an overview of the available documentation. 

## Notation

Except for Linux/UNIX-specific sections, file paths and environment variables are printed in the Windows convention, e.g.,

```
%HALCONROOT%\examples\extension_package\halconuser
```

to denote the subdirectory `halconuser` containing an example package within the HALCON base directory referenced by the environment variable `HALCONROOT` (see [section A.2](#) on page 46 for more information on environment variables). The same expression in Linux/UNIX convention would look like

```
$HALCONROOT/examples/extension_package/halconuser
```



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	HALCON Configurations . . . . .	1
1.2	Releases and HALCON Versions . . . . .	2
1.3	Licensing . . . . .	2
1.4	System Requirements . . . . .	4
<b>2</b>	<b>Installing HALCON</b>	<b>5</b>
2.1	How to Obtain HALCON . . . . .	5
2.2	Installing HALCON for the First Time . . . . .	6
2.2.1	CD (Windows NT/2000/XP) . . . . .	6
2.2.2	CD (UNIX) . . . . .	8
2.2.3	WWW . . . . .	10
2.3	Installing Additional Parts of HALCON . . . . .	11
2.4	Updating HALCON . . . . .	11
2.5	Upgrading HALCON . . . . .	11
2.6	Uninstalling HALCON . . . . .	12
2.6.1	Windows NT/2000/XP . . . . .	12
2.6.2	UNIX . . . . .	12
2.7	Managing Multiple HALCON Versions . . . . .	13
2.8	Installing HALCON Frame Grabber Interfaces . . . . .	14
2.9	Installing HALCON Extension Packages . . . . .	14
2.9.1	Using an Extension Package Within HDevelop . . . . .	14
2.9.2	Using an Extension Package in a Stand-Alone Application . . . . .	15
<b>3</b>	<b>All About HALCON Licenses</b>	<b>17</b>
3.1	What is a License? . . . . .	17
3.2	Evaluation Licenses . . . . .	19
3.3	Development Licenses . . . . .	20
3.3.1	Licenses Bound to a Network Card . . . . .	20
3.3.2	Licenses Bound to a Dongle . . . . .	21
3.3.3	Floating Licenses . . . . .	21
3.4	Runtime Licenses . . . . .	25
3.5	How to Upgrade a License . . . . .	27
<b>4</b>	<b>Troubleshooting</b>	<b>29</b>

4.1	Problems During Installation (Windows NT/2000/XP)	29
4.2	Problems During Uninstallation (Windows NT/2000/XP)	30
4.3	Problems Concerning Licenses	33
4.3.1	Extracting Host IDs	33
4.3.2	Dongle Drivers	34
4.3.3	The License Manager Daemon	37
4.4	Troubleshooting in HDevelop or HALCON Applications	40
4.5	Miscellaneous Problems	41
<b>A</b>	<b>More on the Installation</b>	<b>43</b>
A.1	The Installed File Structure	44
A.2	HALCON's Environment Variables	46
A.3	Information for Users of ActivVisionTools	49
	<b>Index</b>	<b>53</b>

# Chapter 1

## Introduction

To use HALCON on a computer, you must

1. install HALCON on this computer and
2. obtain a license (except for the demo version).

Before looking into the details of these two steps in [chapter 2](#) on page 5 and [chapter 3](#) on page 17, this chapter gives an overview of the different HALCON versions and licensing methods. Finally, it describes the system requirements for running HALCON.

### 1.1 HALCON Configurations

You can use HALCON in three configurations:

1. **Development version**

The development version (sometimes also denoted as full version) includes the full spectrum of HALCON, i.e., language interfaces to C, C++, and COM, interfaces to about 50 image acquisition devices, the Extension Package Interface, which allows you to integrate your own HALCON operators, and, of course, the interactive development tools HDevelop and HMatchIt. You need this version whenever you want to develop applications based on HALCON.

2. **Runtime version**

If you have finished developing an application based on HALCON, you only need a runtime version of HALCON for each computer where the application is to be run. Since the runtime version is not determined for developing applications it does not include the development tools HDevelop and HMatchIt. Furthermore, you can obtain runtime versions that include only parts of the functionality (so-called *modules*); please contact your local distributor for more information.

3. **Demo version**

The demo version is basically a special version of HDevelop with the full image processing functionality but some limitations, e.g., without interfaces to image acquisition devices or programming languages. Using the demo version, you can test all HALCON operators within the HDevelop environment for an unlimited amount of time.

## 1.2 Releases and HALCON Versions

The term *version* has a second meaning: It denotes the major HALCON releases, e.g., HALCON 6.1 or HALCON 7.0, to differentiate them from so-called *maintenance releases* like HALCON 6.1.2. The main differences between these two types of releases are:

- **Functionality**

A new HALCON *version* always represents a major step in the functionality. This means that it contains a significant number of new operators, but possibly also new auxiliary tools, be it in HDevelop or stand-alone tools like HMatchIt. Furthermore, the functionality of individual operators may be extended or operators have been sped up. Of course, all currently known bugs in the preceding release will have been fixed.

In contrast, the main intention of a *maintenance release* is to fix all currently known bugs. Nevertheless, such a release typically also brings some speed-ups and minor functional extensions of existing operators.

- **Compatibility**

A new HALCON *version* is not downward compatible, with the following implications: First, you must upgrade your HALCON license (see [section 3.5](#) on page 27). Second, if you want to run applications created with an older release under the new version, you must regenerate the applications, as the new HALCON library is not binary compatible to the old one. The term 'applications' includes also frame grabber interfaces and extension packages you created yourself based on an older release. Note that a new version may also be source-code incompatible in some parts, e.g., the signature of an operator or a class method may have been changed. These changes are indicated in the release notes of the HALCON version. In such a case, you must adapt the source code of your application before regenerating it. Finally, an ActivVisionTools release based on an older HALCON release cannot be directly used with a new HALCON version. For some combinations of ActivVisionTools and HALCON releases this problem is solved during the HALCON installation; please refer to [section A.3](#) on page 49 for more information.

In contrast, a *maintenance release* is in most cases fully downward compatible to its corresponding version. This compatibility includes the license. Please note, however, that some maintenance releases may not be fully binary or source-code compatible because of technical reasons. In such cases, the release notes will contain corresponding warnings and describe how to proceed.

## 1.3 Licensing

To run HALCON on a computer, you need a license. The only exception to this rule is the HALCON demo version, i.e., the demo version of the development tool HDevelop, which can be run without a license. The license itself is stored in a file named `license.dat` and resides in the subdirectory `license` of the folder where you installed HALCON. In fact, from HALCON 7.0.3 on the name of the license file is not restricted to "`license.dat`" anymore. Now, it has to begin with "license" and end with ".dat", but in between other information can be included. Thus, different licenses for different versions (e.g., `license70.dat` for version 7.0) can be stored in the same directory.

Licenses are always issued for a certain HALCON version (i.e., major release, see [section 1.2](#)), e.g., for HALCON 7.0. However, a license is not exclusively bound to this version: It is *upward compatible*

*within the version number*, i.e., licenses for HALCON 7.0 are also valid for HALCON 7.0.1. Note that in contrast to previous versions, HALCON 7.0 licenses are not downward compatible, i.e., **a license for HALCON 7.0 is not valid for HALCON 6.1.**



The three possible licensing types mainly correspond to the different HALCON versions described in [section 1.1](#) on page 1. Detailed information about HALCON licenses can be found in [chapter 3](#) on page 17.

- **Evaluation license**

To evaluate the full power of HALCON, you can obtain an evaluation license from your local distributor free of charge. This type of license is not bound to any computer hardware, i.e., you can use HALCON on any computer you installed it on; however, it is only valid for a limited time, typically for a month. Note that you may not use this license to develop commercial applications.

- **Development license**

To develop HALCON applications, whether in HDevelop or via a programming language, you need a development license. In contrast to the evaluation license, this license is permanent. Furthermore, this license is bound to a certain hardware component (network card or dongle, see [section 3.1](#) on page 17).

If you want to use HALCON on multiple computers simultaneously, you need a license for each of them. You can either obtain multiple *node-locked licenses*, or, if the computers are connected via a network, you can also use a *floating license*. The main point of using floating licenses is that you do not need to specify on which computers you want to run HALCON, but only on how many of them simultaneously. Floating licenses are described in detail in [section 3.3.3](#) on page 21.

- **Runtime license**

If you finished developing your application based on HALCON and now want to install and run it on a customer's computer, you only need a runtime license. Like development licenses, runtime licenses are permanent and bound to a certain hardware component (network card or dongle); note, however, that there are no floating runtime licenses.

As already noted, you can obtain runtime licenses that cover only parts of the functionality (so-called *modules*). Please contact your local distributor for more information.

Operating System	Processor	Compiler / Environment
Windows NT/2000/XP	Pentium or compatible	Microsoft Visual Studio Microsoft Visual Studio .NET Borland Delphi
Linux 2.2/2.4	Pentium or compatible	gcc 2.95 gcc 3.2/3.3
Linux 2.4	Itanium 2	gcc 3.3
Solaris 7 or higher	SPARC	CC
IRIX 6.5	MIPS	CC
Tru64 UNIX 5.1	Alpha	cxx

Table 1.1: Platforms supported by HALCON .

## 1.4 System Requirements

[Table 1.1](#) shows the requirements for running HALCON on the currently supported operating systems. It should run on newer versions of an operating system; however, we cannot guarantee this.

Note that under Linux `libc6` is required, which is available on most current Linux distributions like RedHat or SuSE. Please check your distribution's documentation.

HALCON can also be used on the Philips TriMedia DSP, with the host computer running Windows NT/2000/XP. Please refer to the Programmer's Guide, [chapter 14](#) on page [105](#), for more information.

## Chapter 2

# Installing HALCON

In this chapter, we show how to

- install HALCON for the first time ([section 2.2](#)),
- install additional parts ([section 2.3](#) on page 11),
- update HALCON to a newer maintenance release ([section 2.4](#) on page 11), and
- upgrade HALCON to a newer version ([section 2.5](#) on page 11).

Furthermore, this chapter explains how to

- uninstall HALCON ([section 2.6](#) on page 12),
- manage multiple versions in parallel ([section 2.7](#) on page 13),
- install updates of frame grabber interfaces ([section 2.8](#) on page 14), and
- install extension packages ([section 2.9](#) on page 14).

## 2.1 How to Obtain HALCON

You can obtain HALCON in form of a CD from your local distributor, or by downloading it from <http://www.mvtec.com/halcon/download>. The CD includes all HALCON configurations (see [section 1.1](#)) while the download provides separate files for the different configurations. Note that maintenance releases are typically not available on CD.

## 2.2 Installing HALCON for the First Time

In the following sections, we explain how to install HALCON for the first time

- from CD on Windows systems: [section 2.2.1](#)
- from CD on Linux/UNIX systems: [section 2.2.2](#) on page 8
- via WWW: [section 2.2.3](#) on page 10



Please note, that **you need administrator privileges** to install HALCON under Windows NT/2000/XP.

### 2.2.1 Installing HALCON from CD under Windows NT/2000/XP



To install HALCON on Windows NT/2000/XP systems, simply insert the CD. This should automatically start the setup program. If the **setup program doesn't start automatically**, execute the program Setup.exe located in the directory nt-x86 of the CD.

#### Step 1: Internal checks, license agreement

First, the setup program checks whether your system meets the requirements for running HALCON (see [section 1.4](#) on page 4). Besides, it checks whether you already installed HALCON on this computer. If not, a license agreement is displayed; its text is stored in the file eula.txt, which will be installed in the folder you select for the installation.

If you are using ActivVisonTools, the setup program checks whether the installed version is compatible to the current HALCON version. Please refer to [section A.3](#) on page 49 for more information.

#### Step 2: Select installation destination

After these checks you can choose a folder where HALCON is to be installed, e.g., C:\Program Files\MVTec\HALCON.

#### Step 3: Select installation type

Then, you are asked to select between the following installation types which correspond to the HALCON configurations described in [section 1.1](#) on page 1. Note that you can install additional parts of HALCON at a later time by starting the setup program again and selecting the parts you need (see [section 2.3](#) on page 11).

#### Full

This installs the complete development version of HALCON, i.e., the full set of libraries and executables including interfaces to image acquisition devices and programming languages. Furthermore, it installs the full documentation including the [Quick Guide](#), the [Application Guide](#), the Reference Manuals for the three programming language interfaces C, C++, and COM, and the full set of example programs (see [section A.1](#) on page 44 for an overview), including all necessary images and image sequences.

### Compact

This installs a minimal development version of HALCON, i.e., besides the full set of libraries and executables including interfaces to image acquisition devices and programming languages the installation encompasses only the User's Manuals and the Reference Manual in HDevelop syntax. Note that neither examples nor images are installed.

### Custom

Like the Full and the Compact installation, this type installs the development version of HALCON. As its name suggests, it lets you select which parts of the documentation, examples, and images you want to be installed. Note that some HALCON experience is required to handle this type of installation.

### Demo

This installs the demo version of HALCON, i.e., the demo version of HDevelop without interfaces to image acquisition devices or programming languages. The installed documentation encompasses the User's Manuals, the [Quick Guide](#), and the Reference Manual in HDevelop syntax. Furthermore, the HDevelop example programs and the images and image sequences used by these examples are installed. Please note, however, that some of the large images sequences are not installed; this means that example programs based on such an image sequence cannot be run. You can install these sequences in a separate step as described in [section 2.3](#) on page 11.

### Runtime

This installs the runtime version of HALCON, i.e., only the set of libraries including interfaces to image acquisition devices and programming languages that are necessary to run a HALCON application. Naturally, this type of installation encompasses neither documentation, examples nor images.

## Step 4: Selections concerning licensing

After this selection, the setup program asks whether to install the driver programs that are necessary if you want to use a dongle-bound license. [Section 3.3.2](#) on page 21 provides detailed information about these dongle drivers, including how to install them manually if you do not let the setup program install them.

The next dialog asks whether the computer is to act as the so-called *floating license server*. In this case, the setup program installs the so-called *license manager daemon*, a system service that manages your floating licenses. Note that when using floating licenses, you need to install the license manager daemon only on one computer: the license server. Please refer to [section 3.3.3](#) on page 21 for more information about floating licenses, e.g., how to install the license manager daemon manually.

## Step 5: The actual installation

After some further questions the actual installation starts, which includes copying files from the CD to the destination folder (see [section A.1](#) on page 44 for an overview of HALCON's fileset), entering information in the Windows registry, and setting environment variables (see also [section A.2](#) on page 46). Depending on the installation type and content, it may be necessary to reboot the computer after the installation is finished.

Apart from that, no further action on your part is required. You can immediately use HALCON, e.g., experiment with the demo version of HDevelop if you have not obtained a license yet. A good starting

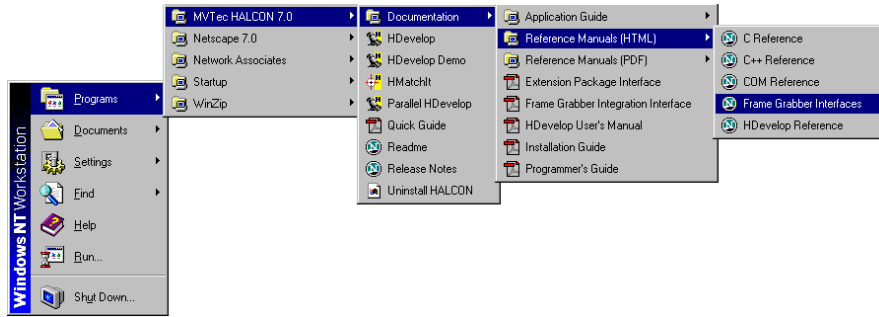


Figure 2.1: Accessing HALCON via the Windows start menu.

point is the Windows start menu, via which you can read the manuals or the release notes, or start HDevelop and HMatchIt (see [figure 2.1](#)). In case that you have already obtained a license from your local distributor, you can now proceed with the installation of the license file `license.dat` which has to be placed in the directory `%HALCONROOT%\license`.

Note that in order to use Parallel HALCON, you must initialize it once on each computer on which it is to be used. This is described in the Programmer's Guide, [section 1.1.1](#) on page 4.

## 2.2.2 Installing HALCON from CD under Linux/UNIX

To install HALCON on Linux or UNIX systems, mount the CD, e.g., on the directory `/cdrom`. On some systems, you need `root` rights to do so. Note that on some systems, most notably Linux, the default mount entry, in `/etc/fstab`, does not allow users to execute programs on a CD. In such cases, you have to mount the CD explicitly with a command similar to the following:

```
mount /dev/cdrom /cdrom
```

To start the installation, execute the shell script

```
install-unix
```

which is located in the top-level directory of the CD.

### Step 1: Internal checks, license agreement

First, the setup program checks whether your system meets the requirements for running HALCON (see [section 1.4](#) on page 4). Then, a license agreement is displayed; its text is stored in the file `eu1a.txt`, which will be installed in the directory you select for the installation.

### Step 2: Select installation destination

The script will ask you in which directory you want to install HALCON; the environment variable `HALCONROOT` must be set to this directory later; see below. If the directory does not exist yet, the script offers to create it. On the other hand, if HALCON was already installed in the selected directory, the script asks whether to remove the old installation completely or replace existing files with the new ones.

```
#!/bin/sh
# Sample shell script for HALCON environment settings
# (sh syntax)

export ARCHITECTURE=i586-linux2.2
export HALCONROOT=/usr/local/halcon
export HALCONIMAGES=${HALCONROOT}/images
export PATH=${PATH}:${HALCONROOT}/bin/${ARCHITECTURE}

if [ $LD_LIBRARY_PATH ] ; then
    export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HALCONROOT}/lib/${ARCHITECTURE}
else
    export LD_LIBRARY_PATH=${HALCONROOT}/lib/${ARCHITECTURE}
fi
```

Figure 2.2: Example for a shell script with environment variables in `sh` syntax, generated when installing HALCON into the directory `/usr/local/halcon` on a Linux system.

### Step 3: Select installation type

Then, you are asked to select between different installation types, which correspond to those described in [section 2.2.1](#) on page 6. Note that you can install additional parts of HALCON at any time by mounting the CD again and copying the corresponding directories and files to the directory where you installed HALCON (see [section 2.3](#) on page 11). You can also execute the shell script `install-unix` once again to install additional parts of HALCON.

### Step 4: The actual installation

Now, the actual installation starts, i.e., the requested parts are copied from the CD to the destination directory.

### Step 5: Set environment variables

After the installation, the following environment variables must be set or modified in order for HALCON to work (see [section A.2](#) on page 46 for more information about these and other environment variables):

- `HALCONROOT`: directory you installed HALCON in
- `HALCONIMAGES`: path(es) in which HALCON searches for images, usually `HALCONROOT/images`
- `ARCHITECTURE`: select value corresponding to the used platform from [table A.1](#) on page 47 (necessary for running of some of the example programs)
- `PATH`: this system variable should include `HALCONROOT/bin/ARCHITECTURE`
- `LD_LIBRARY_PATH`: this system variable should include `HALCONROOT/lib/ARCHITECTURE`

We recommend to set the environment variables in a login script or a shell resource script, e.g., `.cshrc` or `.profile`. Depending on the chosen value of `HALCONROOT` **the installation script automatically**



creates two example shell scripts `.cshrc_halcon` and `.profile_halcon` which contain the necessary settings in `csh` and `sh` syntax, see [figure 2.2](#). The suitable shell script can be used to be included in your login script. Note that different shells offer different commands to set environment variables, e.g., `setenv <variable> <value>` or `export <variable>=<value>`. Please consult your shell's documentation for further information. If a value consists of multiple items, those items must be separated by *colons*.

## Step 6: Further configuration

If you are using floating licenses, you must install and start the license manager daemon as described in [section 4.3.3](#) on page 37.

If you want to use a dongle-bound license (Linux only), see [section 4.3.2.3](#) on page 35 for information how to install the USB dongle driver.

In order to use Parallel HALCON, you must initialize it once on each computer it is to be used on. This is described in the Programmer's Guide, [section 1.1.1](#) on page 4.

Apart from that, no further action on your part is required. You can immediately use HALCON, e.g., experiment with the demo version of HDevelop if you have not obtained a license yet. In case that you have already obtained a license from your local distributor, you can now proceed with the installation of the license file `license.dat` which has to be placed in the directory `$HALCONROOT/license`.

## 2.2.3 Installing HALCON via WWW

All HALCON releases can be downloaded from <http://www.mvtec.com/halcon/download>. Note that you must first register yourself before downloading software. Note that the registration process differentiates between the demo and the full version. How to download and install HALCON via WWW is described in detail on the web pages themselves. Therefore, in the following we only note some characteristics of the WWW installation, which mostly result from trying to keep the downloaded files as small as possible.

- You can directly download either the demo version or the full version which does not include the demo version of HDevelop.
- To decrease the size of the download files **neither the demo nor the full version include any images and documentation**. These files can be downloaded separately in form of ZIP files and unpacked in the directory where you installed the main part of HALCON.
- The Windows version comes as an executable file that contains both the setup program and the HALCON files. We recommend to save the executable to disk and then to start the installation by executing it, e.g., by double-clicking it in the Windows Explorer. The installation is performed as described in [section 2.2.1](#) on page 6.
- The Linux/UNIX version comes as a zipped archive without installation script. Thus, you will automatically install the full version of HALCON (without images and documentation, as noted above).



## 2.3 Installing Additional Parts of HALCON

### Windows NT/2000/XP systems

On Windows NT/2000/XP systems, you can install additional parts of HALCON at any time by starting the setup program another time, e.g., by inserting the CD. The setup program then lists the already installed parts and lets you choose additional parts.

If you downloaded the full version of HALCON via WWW as described in [section 2.2.3](#) on page 10, you get a similar behavior by starting the downloaded executable again.

### Linux/UNIX systems

On Linux/UNIX systems, you can install additional parts of HALCON at any time by mounting the CD again and copying the corresponding directories and files to the directory where you installed HALCON. To copy directories, open a shell, change into the root directory on the CD, and type


```
tar -cf - <directory_to_be_copied> | ( cd $HALCONROOT; tar -xf - )
```

## 2.4 Updating HALCON

With HALCON, the term “update” means to install a newer maintenance release over a release based on the same HALCON version, e.g., HALCON 6.1.2 over 6.1.1 or 6.1. As described in [section 1.2](#) on page 2, you can update HALCON without needing a new license.

Typically, maintenance releases are not available as a CD, i.e., you must install them via WWW as described in [section 2.2.3](#) on page 10. When updating an existing HALCON installation under Windows the setup program will ask you whether you want to replace the already installed components with the updated ones.

## 2.5 Upgrading HALCON

With HALCON, the term “upgrade” means to install a newer version over an older one, e.g., HALCON 7.0 over 6.1 or 6.1.2. As described in [section 1.2](#) on page 2, new versions are not compatible to older ones. Therefore, we recommend to **uninstall the older HALCON release before installing the new one**. On Windows systems, this is especially important in order to clean up the registry. 

The easiest way to uninstall the older version is to start the installation of the new version, because the setup programs both on Windows and Linux/UNIX systems offer to perform the uninstallation for you if they detect an older version. More information regarding uninstalling HALCON can be found in [section 2.6](#). If you want to keep the older version, please refer to [section 2.7](#) on page 13.

The installation process itself is the same as the first-time installation described in [section 2.2](#) on page 6. Please note that after installing the new version you must upgrade the license as described in [section 3.5](#) on page 27.

## 2.6 Uninstalling HALCON

### 2.6.1 Uninstalling HALCON under Windows NT/2000/XP

There are multiple ways to uninstall HALCON; please note that not all are available for older versions of HALCON:

1. Select Start ▷ Programs ▷ MVTec HALCON ▷ Uninstall HALCON in the Windows start menu.
2. Choose Add/Remove Programs in the system control panel.
3. When installing HALCON, the setup program checks whether there is an old version of HALCON and allows you to uninstall it. Note that during updating an existing HALCON installation under Windows the setup program will ask you whether you want to replace the already installed components with the updated ones.

Typically, this is all you have to do. If you encounter any problems, please refer to [section 4.2](#) on page 30.



Note that **the uninstallation removes exactly those files that were installed**. This has two implications: If you added files after the installation manually, e.g., new frame grabber interfaces, extension packages, images, or manuals, these files and the corresponding directories will “survive” the uninstallation. On the other hand, if you only modified a file, e.g., an example, without changing its name the uninstallation will remove it nevertheless. Therefore you might want to copy such files to another directory before starting the uninstallation.

The uninstallation warns if it encounters DLLs or executables that might be or are actually used by another application. The warnings regarding `halconx.dll` and `HFGDirectShow.dll` can safely be ignored, i.e., the files can be removed. In contrast, warnings regarding `mvtecd.exe` and `lmgrd.exe` are serious; they indicate that the license manager daemon was installed manually but not uninstalled. In this case, we recommend to abort the uninstallation process, uninstall the license manager daemon manually as described in [section 4.3.3](#) on page 37, and then restart the uninstallation of HALCON.

The uninstallation process will not remove any *user-specific settings*. This means that registry entries concerning, e.g., the layout of HDevelop or its file history, will be left in the category `HKEY_CURRENT_USER ▷ Software ▷ MVTec ▷ HALCON`. You may remove these entries manually without risk. Furthermore, it does not remove the dongle drivers as they might be needed by another application; [section 3.3.2](#) on page 21 describes how to remove them manually.

### 2.6.2 Uninstalling HALCON under Linux/UNIX

HALCON provides no uninstallation script for Linux/UNIX systems, therefore you must perform the uninstallation manually. In case that you are using a floating license you have to uninstall the floating license daemon (see [section 3.3.3](#) on page 21). The actual uninstallation consists of simply removing the content of the HALCON base directory `$HALCONROOT` and all its subdirectories, e.g., by executing

```
rm -rf $HALCONROOT
```

Furthermore, remove the subdirectory `.hdevelop` of the directory referenced by the environment variable `HOME` (see [section A.2](#) on page 46); `HDevelop` creates this directory to save options, window positions, and the file history.

Finally, delete references to HALCON from the environment variables (see [section 2.2.2](#) on page 8).

To remove the dongle driver, refer to [section 4.3.2.3](#) on page 35.

## 2.7 Managing Multiple HALCON Versions

### Linux/UNIX systems

On Linux/UNIX systems, you can switch between different HALCON versions by setting the environment variable `HALCONROOT` accordingly. Note that in order for this method to work, paths based on `HALCONROOT` in other environment variables like `PATH` and `LD_LIBRARY_PATH` must use the variable and not its content. See [figure 2.2](#) on page 9 for an example.

### Windows NT/2000/XP systems

Under Windows NT/2000/XP, you must adapt those environment variables that are set during the installation, i.e., `HALCONROOT`, `PATH`, and `HALCONIMAGES`, and those you set yourself (e.g., `HALCONEXTENSIONS`). Please refer to [section A.2](#) on page 46 for more information about setting environment variables.

If you are using the HALCON/COM interface, you must also re-register the corresponding DLL `halconx.dll`, for example as follows: Open a Windows Command Prompt and change into the subdirectory `bin\i586-nt4` of the root directory of your “old” HALCON installation. Unregister `halconx.dll` by typing

```
regsvr32 /u halconx.dll
```

Then change into the corresponding subdirectory of your other HALCON installation and register its `halconx.dll` by typing

```
regsvr32 halconx.dll
```

As an alternative to the Command Prompt, you can unregister and register `halconx.dll` via the dialog `Start > Run` together with the Windows Explorer: In the latter, “open” the directory `bin\i586-nt4` of the root directory of your old HALCON installation. Now, type `regsvr32 /u` in the dialog `Run` and then drag and drop `halconx.dll` from the Explorer into the dialog, where it automatically appears with the full path. To execute the command, click `OK`. Then, open the directory `bin\i586-nt4` of the root directory of the other HALCON installation in the Explorer, type `regsvr32` in the dialog `Run`, drag and drop `halconx.dll` from the Explorer into the dialog, and again click `OK`.

Please note that if you are using `ActivVisionTools`, you may have to register another `ActivVisionTools OCX` (see [section A.3](#) on page 49).

## 2.8 Installing HALCON Frame Grabber Interfaces

With every HALCON installation (except the demo version), you automatically obtain the latest release of the currently available frame grabber interfaces (see also [section A.1](#) on page 44). In between HALCON releases, however, frame grabber interfaces might be updated by MVTec or a frame grabber manufacturer. Such updates are indicated on MVTec's WWW server, to which you can connect by selecting HDevelop's menu entry `Help > HALCON News (WWW)`. You can then download the interface together with its documentation and HDevelop example programs, and install it as described on the corresponding web pages.

## 2.9 Installing HALCON Extension Packages

The HALCON Extension Package Interface enables you to integrate newly developed image processing algorithms into HALCON in the form of so-called *extension packages*. The same mechanism is used by MVTec to extend the current HALCON release with additional functionality. Which extensions packages are currently available can be checked by selecting HDevelop's menu entry `Help > HALCON News (WWW)`, which connects to MVTec's WWW server.

This section describes how to integrate a (downloaded) package named `newextpkg` in order to use it within your HALCON system.

First, move the package to the directory `%HALCONROOT%` and unpack it there. Then, add the *complete* path of the package, e.g.,

```
%HALCONROOT%\packages\newextpkg
```

to the environment variable `HALCONEXTENSIONS`. Note, that the delimiter between paths in an environment variable is a semicolon on Windows systems and a colon on Linux/UNIX systems.



**Never change the name of an extension package or the corresponding names of the libraries or DLLs contained in it.** These names are encoded *within* the libraries/DLLs. If you change the names this information will no longer match. Thus, the loader of the operating system will fail to open the dynamic libraries.

If the package contains images used, e.g., within example programs we recommend to include the (complete) path to the corresponding directory `images` within the package in the environment variable `HALCONIMAGES` (see [section A.2](#) on page 46) to access those images without specifying a complete path.

### 2.9.1 Using an Extension Package Within HDevelop

In order to use a new package within HDevelop under Windows NT/2000/XP, Solaris, IRIX, or Tru64 UNIX, you just have to restart the program. HDevelop automatically integrates all extension packages specified in `HALCONEXTENSIONS`, i.e., the operators contained in a package can be accessed and used like any other HALCON operator.

Under Linux/UNIX, you have to include the package library subdirectory (e.g., `lib/i586-linux2.2` or `lib/i586-linux2.2-gcc32`) in the environment variable `LD_LIBRARY_PATH` before starting HDevelop the first time. Note that under IRIX you have to adapt the environment variable `LD_LIBRARYN32_PATH` instead of `LD_LIBRARY_PATH`.

## 2.9.2 Using an Extension Package in a Stand-Alone Application

If you want to generate a stand-alone application that uses an extension package, you have to link the package libraries (DLLs under Windows, shared libraries under Linux/UNIX) to the application code, in addition to the standard HALCON library.

### 2.9.2.1 Using an Extension Package Under Windows NT/2000/XP

In order to create new application programs (written in C or C++) you have to link `packagec.lib` or `packagecpp.lib` to your objects. Furthermore, you will need `halconclib.lib` or `halconcpp.lib` (as for any HALCON application).

To be able to link the package DLL to your application program, the *complete* DLL file path of the new package, e.g.,

```
%HALCONROOT%\packages\newextpkg\bin\i586-nt4
```

must be added to the environment variable PATH.

**Do not copy a package DLL into the Windows system directories**, as it would be loaded twice in this case!



### 2.9.2.2 Using an Extension Package Under Linux/UNIX

In order to create new application programs (written in C or C++) you have to link `libnewextpkg.so` and `libnewextpkgc.so` or `libnewextpkgccpp.so` to your objects (besides `libhalcon.so` and `libhalconclib.so` or `libhalconcpp.so` as for any HALCON application).

Furthermore, you have to add the path to the package library subdirectory `lib/$ARCHITECTURE` to the environment variable `LD_LIBRARY_PATH`, otherwise the loader will fail to access the libraries. [Table A.1](#) on page 47 lists the content of `ARCHITECTURE` for the different operating systems. Note that under IRIX you have to adapt the environment variable `LD_LIBRARYN32_PATH` instead of `LD_LIBRARY_PATH`.



## Chapter 3

# All About HALCON Licenses

[Section 1.3](#) on page 2 already contained an overview of the possible licensing schemes. In this chapter, you will find detailed information about how to obtain and install

- evaluation licenses ([section 3.2](#) on page 19),
- development licenses ([section 3.3](#) on page 20), and
- runtime licenses ([section 3.4](#) on page 25),

Finally, [section 3.5](#) on page 27 shows how to upgrade a license.

### 3.1 What is a License?

HALCON's licensing mechanisms are based on the license manager software *FLEXlm* from Macrovision. The licenses themselves are stored in so-called *license files*; examples files are depicted in the following sections. The content of these files specifies

- what is licensed (e.g., HALCON 7.0 development version, runtime version, etc.)
- whether the license is temporary (evaluation license) or permanent
- the hardware to which the license is bound (see below)
- additional information for floating licenses.

License files are named `license.dat` (or `license70.dat` or similar, see [section 1.3](#) on page 2) and reside in the subdirectory `license` of the folder where you installed HALCON. **Note that HALCON will not run if you modify the license keys within the license file manually!**



## Licenses Bound to Hardware Components

As noted in [section 1.3](#) on page 2, development and runtime licenses are bound to a certain hardware component, i.e., you can use HALCON only on the computer that is identified by this hardware component. This is either the *network card* (see [section 3.3.1](#) on page 20) or, on Windows and Linux systems, a *dongle* (see [section 3.3.2](#) on page 21).

The license manager software FLEXlm identifies these components by a so-called *host ID*. HDevelop, both in the full and in the demo version, automatically checks whether any network cards or dongles are present and displays their IDs in the menu item `Help > About`. For an example see [figure 3.1](#), which was generated on a computer equipped with a network card and a dongle.

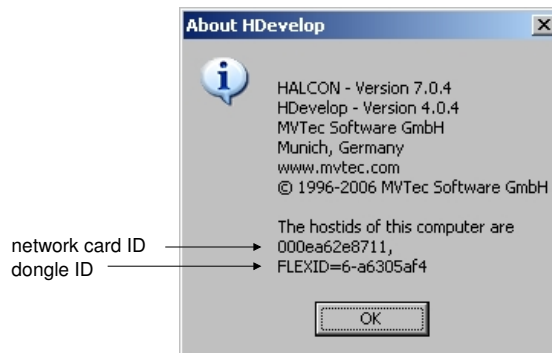


Figure 3.1: Identifying information in the HDevelop window `About`.

## Using HALCON on Different Computers

Depending on the hardware component chosen for licensing, the license is bound to a certain computer. Dongle-bound licenses allow to use HALCON on different computers by moving the dongle. Network cards can be switched between computers, but in practice they can be regarded as being bound to a computer.

Thus, if you want to develop applications with HALCON on more than one computer at different times, the easiest solution is to obtain a dongle-bound license on Windows and Linux systems. As an alternative, you can use a *floating license* (see below).

## Using HALCON in a Computer Network

If you are developing applications based on HALCON in a team on different computers in a network, you can use a so-called *floating license*. With this type of license, you only specify *how many* HALCON applications are allowed to run simultaneously; on which computer they are run does not matter. For details about floating licenses see [section 3.3.3](#) on page 21.

As remarked above, floating licenses also are an option if a single person wants to use HALCON on different computers at different times.

## 3.2 Evaluation Licenses

As already noted in [section 1.3](#) on page 2, with an evaluation license you can evaluate the full functionality of HALCON free of charge on any computer. The only restriction is that evaluation licenses are valid only for a limited time (typically a month).

### Step 1: Obtain the license

You can obtain an evaluation license from your local distributor. The distributor will send you a *license file* that looks similar to the one depicted in [figure 3.2](#): The lines starting with # are comments; the date 01/2004 indicates that the evaluation license is valid until the end of January 2004.

```
#####  
# Evaluation License for 01/2004 (ID: DEMO) #  
#####  
FEATURE MVTec_Halcon mvtecd 7.0 01-feb-2004 0 1C8F924A9B18283244B1 \  
    VENDOR_STRING=511 HOSTID=DEMO ck=7  
FEATURE MVTec_HDevelop mvtecd 7.0 01-feb-2004 0 AC9F020ABED28DA325BA \  
    VENDOR_STRING=511 HOSTID=DEMO ck=113
```

Figure 3.2: Example evaluation license for January 2004.

The lines starting with FEATURE contain the actual license data; they are called *license keys*. Evaluation licenses contain two license keys: MVTec\_Halcon includes those HALCON parts that are necessary for running HALCON, while MVTec\_HDevelop includes additional parts for developing, e.g., HDevelop.

As you can see, the license keys contain the licensed version number of HALCON ([figure 3.2](#): 7.0). As noted in [section 1.3](#) on page 2, the license is upward compatible within the version number, i.e., licenses for HALCON 7.0 are also valid for all maintenance releases of HALCON 7.0, e.g., HALCON 7.0.1.

The entry following the HALCON version specifies when the evaluation license key expires ([figure 3.2](#): 01-feb-2004).

### Step 2: Install the license

“Installing” the license simply means placing the license file into the subdirectory `license` of the folder where you installed HALCON. If necessary, rename the file to `license.dat` (or `license70.dat` or similar, see [section 1.3](#) on page 2).

Note that you can evaluate HALCON on any computer where you installed HALCON just by copying the evaluation license file into the corresponding subdirectory `license`. You can also evaluate HALCON under different operating systems.

## 3.3 Development Licenses

Like the evaluation license, a development license allows you to use the full functionality of HALCON including the development tools like HDevelop (see also [section 1.3](#) on page 2). But in contrast to the evaluation license, a development license is permanent, i.e., there is no temporal restriction. However, it must be bound to a certain hardware component (see also [section 3.1](#) on page 17). The following section describes how to proceed to obtain and install a license bound to a

- network card ([section 3.3.1](#))
- dongle ([section 3.3.2](#))

### 3.3.1 Licenses Bound to a Network Card

#### Step 1: Extract the host ID

Start HDevelop (full or demo version) and select the menu item Help ▸ About. This dialog displays the host ID of the network card, e.g., the string "00e02958e36a" in [figure 3.1](#) on page 18.

If HDevelop fails to detect a host ID although your computer does have a network card, please try to extract the host ID manually as described in [section 4.3.1](#) on page 33. This section also describes what to do if IDs like "ffffffff" or "0", or multiple IDs are displayed.

#### Step 2: Obtain the license

Send the host ID of the network card to your local distributor. The distributor then sends you a *license file* that looks similar to the one depicted in [figure 3.3](#).

```
#####
# MVTec Software GmbH (ID: 00e02958e3b6) #
#####
FEATURE MVTec_Halcon mvtecd 7.0 01-jan-0000 0 9C5F4246EF7E9331AE2A \
  VENDOR_STRING=511 HOSTID=00e02958e3b6 ck=95
FEATURE MVTec_HDevelop mvtecd 7.0 01-jan-0000 0 7C6FE22615F23052B293 \
  VENDOR_STRING=511 HOSTID=00e02958e3b6 ck=237
```

Figure 3.3: Example development license, bound to network card.

Like the evaluation license depicted in [figure 3.2](#) on page 19, it contains two *license keys* (lines starting with FEATURE), one for the runtime parts of HALCON (MVTec\_Halcon) and one for the development parts (MVTec\_HDevelop). The differences to the evaluation license show up as follows:

- The expiration date is set to 01-jan-0000, which means that the license is permanent (alternatively, the entry may contain the string permanent).
- Both license keys are bound to the HOSTID you extracted in the first step (in the example: 00e02958e36a).

If the FEATURE line contains the additional keyword TS\_OK, then this license allows also the checkout from a terminal server.

### Step 3: Install the license

Place the license file into the subdirectory `license` of the folder where you installed HALCON. If necessary, rename the file to `license.dat` (or `license70.dat` or similar, see [section 1.3](#) on page 2).

## 3.3.2 Licenses Bound to a Dongle

Note that for Windows systems we assume that you let the setup program install the driver programs necessary for using dongles as described in [section 2.2.1](#) on page 6. If you did not install the drivers, please refer to [section 4.3.2](#) on page 34. On Linux systems, the dongle driver has to be installed manually, which is described in [section 4.3.2.3](#) on page 35.

### Step 1: Obtain dongle and license

Please note that **you cannot use any dongle but only those supplied by MVTec via your local distributor**. Currently, HALCON supports dongles connected to the parallel port and USB dongles.



The distributor will send you the dongle together with a corresponding *license file*, which looks similar to the one depicted in [figure 3.4](#). The only difference to the network card license depicted in [figure 3.3](#) on page 20 is that the entry `HOSTID` now contains the ID of the dongle (in the example: `FLEXID=6-a6305af4`). This ID is also printed on the back of the dongle. Dongles for the parallel port have IDs starting with `'FLEXID=6-'` or `'FLEXID=7-'`, while USB dongles have IDs starting with `'FLEXID=9-'`.

If the `FEATURE` line contains the additional keyword `TS_OK`, then this license allows also the checkout from a terminal server.

```
#####
# MVTec Software GmbH (ID: FLEXID=6-a6305af4) #
#####
FEATURE MVTec_Halcon mvtecd 7.0 01-jan-0000 0 9CAF22FFEF0BA4884FBC \
  VENDOR_STRING=511 HOSTID=FLEXID=6-a6305af4 ck=99
FEATURE MVTec_HDevelop mvtecd 7.0 01-jan-0000 0 6CEFC24F38BABA7A5A88 \
  VENDOR_STRING=511 HOSTID=FLEXID=6-a6305af4 ck=170
```

Figure 3.4: Example development license, bound to dongle.

### Step 2: Install the license

Place the license file into the subdirectory `license` of the folder where you installed HALCON. If necessary, rename the file to `license.dat` (or `license70.dat` or similar, see [section 1.3](#) on page 2).

If you want to use HALCON on more than one computer by switching the dongle between them, repeat this step for every computer.

## 3.3.3 Floating Licenses

[Section 3.1](#) on page 17 already briefly described the basic concept of floating licenses; now, we take a closer look:

- With a floating license, you can use HALCON on multiple computers in a network without having to identify each of these computers.
- When ordering a floating license via your local distributor you have to specify *how many* HALCON applications are allowed to run simultaneously; on which computer they are run does not matter.
- Only one computer must be identified: the so-called *license server*. On this computer, the so-called *license manager daemon* is installed, a program that keeps track of the HALCON applications currently being run (see [section 4.3.3](#) on page 37 for details).  
As already described for non-floating licenses, computers can be identified via network card or dongle.



- Note that **only development licenses are available as floating licenses**.

Floating licenses are well-suited especially in the following scenarios:

- A single person wants to develop HALCON applications on different computers, and does not want to use a dongle-bound license.
- A group of persons wants to develop HALCON applications simultaneously in a computer network.

### Step 1: Choose the computer acting as the license server

In principle, any computer can be chosen as the license server; it need not to be a “server” in the sense that it must provide special functionality or have a special kind of operating system, e.g., Windows 2000 Server, installed. The only requirement is that the computer must be accessible whenever HALCON is to be used in the network.

You can even use different architectures for the license server and for the HALCON applications, respectively, e.g., a Linux workstation for the license server and Windows for developing HALCON applications. The license server can also be used for developing HALCON applications; but even if not, HALCON must be installed on it.

### Step 2: Extract the host ID of the license server

Like all development licenses, floating licenses must be bound to a hardware component. Here, it is the computer acting as the license server that must be identified, be it via a network card or a dongle. Please refer to [section 3.3.1](#) on page 20 (network card). If you choose a dongle-bound license, no further action is required as you get the dongle together with the license (see [section 3.3.2](#) on page 21).

### Step 3: Obtain the license

Send the desired number of licenses (i.e., the maximum number of HALCON applications that should run simultaneously), the hostname of the computer which acts as the license server, and – except in case of a dongle-bound license – the extracted host ID of the license server to your local distributor.

The distributor then sends you a license file, which looks similar to the one depicted in [figure 3.5](#). If you requested a dongle-bound license, you will also receive the dongle.

## Step 4: Adapt the license

In contrast to the license types described in the previous sections, you may need to adapt parts of the floating license file. Therefore, we take a closer look at the example floating license depicted in [figure 3.5](#). It is bound to the same network card as the example node-locked license depicted in [figure 3.3](#) on page 20.

```
#####
# MVTec Software GmbH (ID: 00e02958e3b6) #
#####
SERVER myservername 00e02958e3b6 27000
DAEMON mvtecd C:\Progra~1\MVTec\Halcon\FLEXlm\i586-nt4\mvtecd
FEATURE MVTec_Halcon mvtecd 7.0 01-jan-0000 7 4C8FF29FED464E201826 \
  VENDOR_STRING=511 ck=63
FEATURE MVTec_HDevelop mvtecd 7.0 01-jan-0000 7 BC1F521FFBD8753F9EB5 \
  VENDOR_STRING=511 ck=126
```

Figure 3.5: Floating license with 7 licenses, bound to network card.

The license consists of the following parts:

- **Description of the license server**

The line starting with `SERVER` describes the computer acting as the *license server* by stating its hostname (in the example: `myservername`), its host ID (`00e02958e3b6`), and the number of the TCP/IP port (`27000`), over which the HALCON applications connect to the license manager daemon running on the license server.

Please note that the **hostname must be specified correctly**. The reason is that the HALCON applications need the name to connect to the license server (even if they are started on the license server itself).

This means that if you did not send the name of the computer acting as the license server to your distributor when requesting the license, you must adapt this entry. The same holds if you decide to switch the dongle to another computer.

You may also need to adapt the port number, e.g., if the default port number `27000` is already used by another software in your computer network. Note, that on many systems all ports  $< 1024$  are privileged and can only be used by privileged accounts!

- **Path to the license manager daemon**

The line starting with `DAEMON` contains the path of the daemon `mvtecd` (see also [section 4.3.3](#) on page 37). This program resides in the subdirectory `FLEXlm\%ARCHITECTURE%` of the folder where you installed HALCON on the license server (see [section A.2](#) on page 46 for possible values of the environment variable `ARCHITECTURE`). In most cases, you must adapt this path. Unfortunately, you cannot use environment variables in the license file.

Note that paths including blanks may not be handled properly under Windows NT/2000/XP: Thus, if you have installed HALCON to a directory like `C:\Program Files\MVTec\HALCON` you will have to use the short path `C:\Progra~1\MVTec\HALCON` instead. You can query short path names as follows: Open a Windows Command Prompt, change into the directory that contains the directory or file whose short name is searched for (here: `C:`), and then type



```
dir /x
```

Note that the daemon `mvtecd` opens a second port. By default, this port is selected by the operating system, and thus its number can change. Especially when using a firewall, you might need to specify the port number explicitly. For this, append the string `port=number` to the line starting with `DAEMON`, for example as follows:

```
DAEMON mvtecd C:\Progra~1\MVTec\HALCON\FLEXlm\i586-nt4\mvtecd port=28000
```

- **The license keys**

Like the node-locked network card license depicted in [figure 3.3](#) on page 20, the floating license contains two *license keys* (lines starting with `FEATURE`), one for the runtime parts of HALCON (`MVTec_Halcon`) and one for the development parts (`MVTec_HDevelop`). In contrast to the node-locked version, the floating license keys do not contain the entry `HOSTID`, of course, because with a floating license HALCON applications can be started on any computer in the network. Instead, the keys specify how many applications can run simultaneously in the entry after the expiration date (in the example: 7). **This part of the license file must not be modified.**



### Step 5: Install the license

As in the previous sections, installing the license file means to rename the file to `license.dat` (or `license70.dat` or similar, see [section 1.3](#) on page 2), if necessary, and then to place it into the subdirectory `license` of the folder where you installed HALCON. Note that you must place a copy of the license file on all computers you installed HALCON on, i.e., on the license server and on all those computers you want to use HALCON on.

### Step 6: (Re-)start the license manager daemon

Finally, you must start the license manager daemon, or restart it if it is already running. Please refer to [section 4.3.3](#) on page 37 for further information.

Note that whenever you get a new floating license you must copy it to all computers and then restart the license manager daemon. The same is true if you modify parts of the license file while the license manager daemon is running. If you use different HALCON versions at the same time, you have to start the different license manager daemons on different servers.

## 3.4 Runtime Licenses

In contrast to a development license, a runtime license only allows to run HALCON applications. Like a development license, a runtime license is permanent, but must be *node-locked*, i.e., bound to a certain hardware component (see also [section 3.1](#) on page 17).

### Step 1: Extract the required modules

To extract the modules that are used by an application proceed as follows:

1. If the application is running in HDevelop, select the menu item File > Modules which will open a dialog. [Figure 3.6](#) shows the result for the HDevelop example program `examples\hdevelop\Applications\FA\ic.dev`, which uses color transformation, blob analysis, and morphology to extract ICs, resistors, and capacitors on a board. As you can see, you only need the module 'Foundation' to run this program.

If you click Save, the required modules are saved in a file with the extension `.txt` in the currently used directory.

2. If the application is written in a programming language (C, C++, COM), insert the operator `get_modules` (see the corresponding entry in the HALCON Reference Manuals for more information) at the end of the program. Note that **the operator `get_modules` will only return the correct modules if all HALCON operators used in the application are executed at least once.**

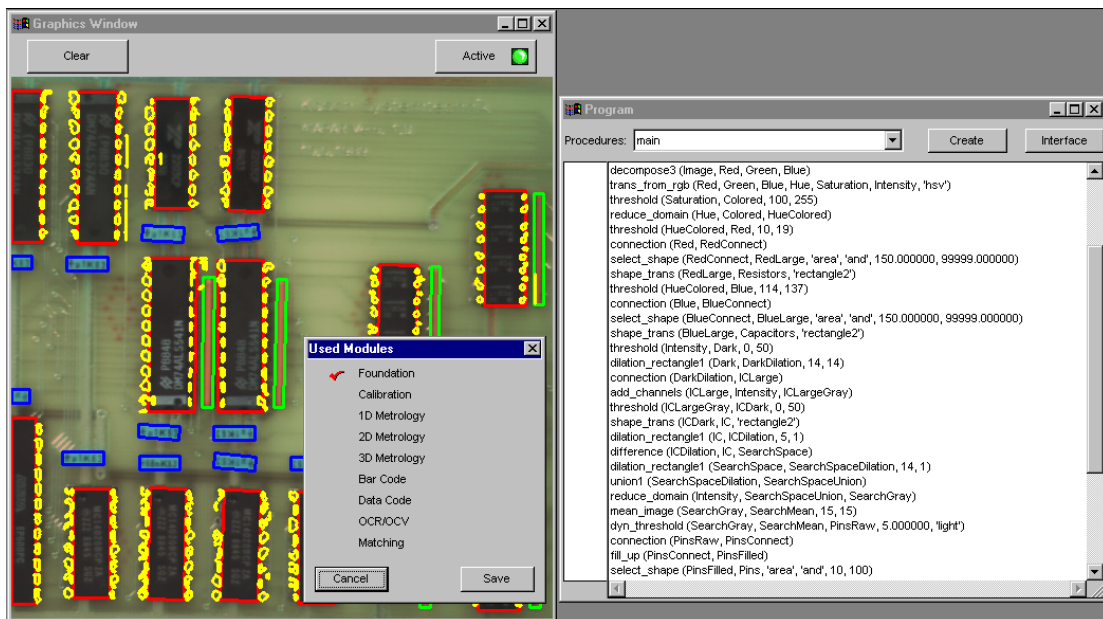


Figure 3.6: Used modules in the example program `examples\hdevelop\Applications\FA\ic.dev`.



### Step 2: Extract the host ID

Please refer to [section 3.3.1](#) on page 20 (network card) for information about how to extract the host ID. If you choose a dongle-bound license, no further action is required as you get the dongle together with the license (see [section 3.3.2](#) on page 21).

### Step 3: Obtain the license

Send the determined module names and – except in case of a dongle-bound license – the extracted host ID to your local distributor.

The distributor then sends you a license file, which looks similar to the one depicted in [figure 3.7](#). If you requested a dongle-bound license, you will also receive the dongle.

```
#####
# MVTec Software GmbH (ID: 00e02958e3b6) #
#####
FEATURE MVTec_Halcon mvtecd 7.0 01-jan-0000 0 7CDF033B01C491200715 \
  VENDOR_STRING=1 HOSTID=00e02958e3b6 ck=45
```

Figure 3.7: Runtime license for the module 'Foundation', bound to network card.

If you compare the depicted license with the corresponding development license in [figure 3.3](#) on page 20, you will note two differences: First, the runtime license contains only one *license key* for the runtime parts of HALCON (FEATURE MVTec\_Halcon). Secondly, the entry VENDOR\_STRING contains a different number (1 instead of 511 for the development license). In this entry the licensed modules are stored; in the example, only 'Foundation' is licensed.

### Step 3: Install the license

Place the license file into the subdirectory `license` of the folder where you installed HALCON. If necessary, rename the file to `license.dat` (or `license70.dat` or similar, see [section 1.3](#) on page 2). In case of a dongle-bound please make sure that the dongle driver is installed (see [section 4.3.2](#) on page 34).

## 3.5 How to Upgrade a License

If you upgrade a HALCON node-locked license to a newer version, e.g., from HALCON 6.1.x to HALCON 7.0, your distributor provides you with a new license file which contains new license keys. This new license file should replace the old one in %HALCONROOT%\license\license.dat . Note that the comments at the beginning of the license file reflect the history of the license, see [figure 3.8](#) for an example upgrade from HALCON 6.0 to HALCON 7.0.

```
#####
# MVTec Software GmbH #
# License history: #
# 13-dec-2001: new license, HALCON 6.0, ID: 00e0293dc732 #
# 01-jun-2002: free upgrade, HALCON 6.1, ID: 00e0293dc732 #
# 07-apr-2004: upgrade, HALCON 7.0, ID: 00e0293dc732 #
#####
FEATURE MVTec_Halcon mvtecd 7.0 01-jan-0000 0 1C9F947AA4AE2B37E1A9 \
  VENDOR_STRING=65535 HOSTID=00e0293dc732 ck=103
FEATURE MVTec_HDevelop mvtecd 7.0 01-jan-0000 0 AC5F042AA75C8EAEABC7 \
  VENDOR_STRING=65535 HOSTID=00e0293dc732 ck=153
```

Figure 3.8: Upgrading a node-locked license from HALCON 6.0 to HALCON 7.0.

In case of upgrading a floating license your distributor will provide you with a so-called *upgrade license file*, which contains upgrade license keys (lines start with UPGRADE. Append the contents of this file to the end of the old license file %HALCONROOT%\license\license.dat , as depicted in [figure 3.9](#) for an example upgrade from HALCON 6.1 to HALCON 7.0. Please note that some editors insert special characters that cannot be interpreted by FLEXlm. Under Windows systems, we recommend to use WordPad.

```
#####
# MVTec Software GmbH (ID: 00e02958e3b6) #
#####
SERVER myservername 00e02958e3b6 27000
DAEMON mvtecd C:\Progra~1\MVTec\Halcon\FLEXlm\i586-nt4\mvtecd
FEATURE MVTec_Halcon mvtecd 6.1 01-jan-0000 5 9C6D4634F08A932DBC2D \
  VENDOR_STRING=65535 ck=82
FEATURE MVTec_HDevelop mvtecd 6.1 01-jan-0000 5 7CADE63412EF3451B596 \
  VENDOR_STRING=65535 ck=34
UPGRADE MVTec_Halcon mvtecd 6.1 7.0 01-jan-0000 5 3COD76C4ADF2BFE51017 \
  VENDOR_STRING=511 ck=54
UPGRADE MVTec_HDevelop mvtecd 6.1 7.0 01-jan-0000 5 1CCDC6246E6E497BB0B4 \
  VENDOR_STRING=511 ck=48
```

Figure 3.9: Upgrading a floating license from HALCON 6.1 to HALCON 7.0.



# Chapter 4

## Troubleshooting

This chapter offers help for problems encountered during installing or uninstalling HALCON on Windows NT/2000/XP systems (section 4.1 and section 4.2, respectively), with the licensing mechanism (section 4.3 on page 33), or when starting HDevelop or your own HALCON applications (section 4.4 on page 40). Section 4.5 on page 41 discusses miscellaneous problems.

### 4.1 Problems During Installation (Windows NT/2000/XP)

- **Registration of `halconx.dll` failed**

On some systems you might get a warning message that the HALCON/COM interface library `halconx.dll` failed to self-register. A possible cause for this may be that the Microsoft library `atl.dll` was not registered properly. This library resides in the directory `%SystemRoot%\system32`, e.g., `C:\WINNT\system32`. To register the library, open a Windows Command Prompt, change into the directory `misc\i586-nt4` on the CD, and execute the supplied program `reg_halconx` twice as follows:

```
reg_halconx atl.dll
reg_halconx
```

The first call of `reg_halconx` registers `atl.dll`, the second one registers `halconx.dll`.

As an alternative to the Command Prompt, you can use the dialog `Start > Run`; here, you can select the program via the button `Browse` or drag and drop `reg_halconx` from the Windows Explorer, followed by the library name if necessary. Note that by default the Windows Explorer does not show DLL files, unless you explicitly tell it to do so.

## 4.2 Problems During Uninstallation (Windows NT/2000/XP)

- **Unregistration of halconx.dll**

If you had to register the HALCON/COM interface library `halconx.dll` manually as described in [section 4.1](#) on page 29, you must unregister it manually before you can uninstall HALCON. To do so, insert a HALCON CD, then open a Windows Command Prompt, change into the directory `misc\i586-nt4` on the CD, and execute the supplied program `reg_halconx` as follows:

```
reg_halconx /u
```

As an alternative to the Command Prompt, open the dialog Start ▷ Run, select the program `reg_halconx` via the button Browse, append the option `/u`, and then click OK.

- **“Internal Error”**

If the uninstall process terminates with a message like

**“Internal Error, unable to load or call external DLL. Please contact your distributor for more information.”**

the most likely reason is that a new HALCON version was installed over an existing one without completely removing the old files first.

Other possible reasons might be that the whole HALCON directory was moved to another position on the hard disk, or the environment variable `HALCONROOT` was changed manually. You have to follow the following steps to recover from the error:

1. Check whether the environment variable `HALCONROOT` matches the location of your HALCON installation. You can check this via the system control panel System (look for Environment) or in a Windows Command Prompt via `echo %HALCONROOT%`. If the variable contains nothing or the files are in some other location, you have to set `HALCONROOT` manually via the System control panel.
2. Check whether the following files / directories are present:
  - In the directory `%HALCONROOT%\FLEX1m\i586-nt4: HalconUninst.dll`  
Note that by default the Windows Explorer does not show DLL files, unless you explicitly tell it to do so.  
`HalconUninst.dll` is a HALCON specific DLL for the uninstallation process. If this file has been removed, you can get a copy of the file from the directory `FLEX1mi586-nt4` on the HALCON CD.
  - **Only for HALCON 7.0**  
In the directory `C:\Program Files\InstallShield Installation Information: {47F424B4-1077-11D8-A0D3-00E01883F42C}` with a set of files. This directory has been created during the installation. If it has been removed, an automatic uninstallation is not possible.
  - **Only for HALCON 6.1**  
In the directory `C:\Program Files\InstallShield Installation Information: {BBEC9F40-4A36-11D6-A14C-00E0296C2846}` with a set of files. This directory has been created during the installation. If it has been removed, an automatic uninstallation is not possible.

- **Only for HALCON releases < 6.1**

In your %HALCONROOT% directory: `Uninst.isu`

This file has been created during the installation. If it has been removed, an automatic uninstallation is not possible.

3. Check whether the registry entry for the uninstaller is set properly: Start `regedit.exe` in a Windows Command Prompt or in the dialog Start ▸ Run.

- **Only for HALCON 7.0**

Go to

```
HKEY_LOCAL_MACHINE -> SOFTWARE -> Microsoft -> Windows ->
CurrentVersion -> Uninstall ->
{47F424B4-1077-11D8-A0D3-00E01883F42C}
```

There, you should find a key named `UninstallString`. The value of this string should be something like this:

```
RunDll32 C:\PROGRA~1\COMMON~1\INSTAL~1\engine\6\INTEL3~1\Ctor.dll,
LaunchSetup "C:\Program Files\InstallShield Installation
Information\{47F424B4-1077-11D8-A0D3-00E01883F42C}\
setup.exe" -!0x9 UninstallHALCON
```

- **Only for HALCON 6.1**

Go to

```
HKEY_LOCAL_MACHINE -> SOFTWARE -> Microsoft -> Windows ->
CurrentVersion -> Uninstall ->
{BBEC9F40-4A36-11D6-A14C-00E0296C2846}
```

There, you should find a key named `UninstallString`. The value of this string should be something like this:

```
RunDll32 C:\PROGRA~1\COMMON~1\INSTAL~1\engine\6\INTEL3~1\Ctor.dll,
LaunchSetup "C:\Program Files\InstallShield Installation
Information\{BBEC9F40-4A36-11D6-A14C-00E0296C2846}\
setup.exe" -!0x9 UninstallHALCON
```

- **Only for HALCON releases < 6.1**

Go to

```
HKEY_LOCAL_MACHINE -> SOFTWARE -> Microsoft -> Windows ->
CurrentVersion -> Uninstall -> HALCON <Version-Number>
```

with `<Version-Number>` being the version number of the HALCON version that you want to uninstall. There, you should find a key named `UninstallString`. The value of this string should be something like this (the option `-c...` is only included if you have chosen an installation with floating licenses):

```
C:\WINNT\IsUninst.exe
-f"C:\Program Files\MVTec\HALCON\Uninst.isu"
-c"C:\Program Files\MVTec\HALCON\FLEX1m\i586-nt4\
HalconUninst.dll"
```



Make sure that the path C:\Program Files\MVTec\HALCON (or equivalent) in the above example points to the folder where you installed HALCON. Note that **file names that contain blanks must be quoted** as in the above example. If you encounter unquoted path names containing blanks, please insert the quotation marks yourself.

4. Close the registry editor and try to run the uninstaller again.

- **Uninstallation failed**

If the automatic uninstallation fails for another reason, proceed as follows:

- For floating licenses only: Uninstall the license manager daemon as described in [section 4.3.3.1](#) on page 37.
- Start `regedit` and delete the key

```
HKEY_LOCAL_MACHINE -> SOFTWARE -> MVTEC -> HALCON
```

- **Only for HALCON 7.0**

Also delete the key

```
HKEY_LOCAL_MACHINE -> SOFTWARE -> Microsoft -> Windows ->
CurrentVersion -> Uninstall ->
{47F424B4-1077-11D8-A0D3-00E01883F42C}
```

- **Only for HALCON 6.1**

Also delete the key

```
HKEY_LOCAL_MACHINE -> SOFTWARE -> Microsoft -> Windows ->
CurrentVersion -> Uninstall ->
{BBEC9F40-4A36-11D6-A14C-00E0296C2846}
```

- **Only for HALCON releases < 6.1**

Also delete the key

```
HKEY_LOCAL_MACHINE -> SOFTWARE -> Microsoft -> Windows ->
CurrentVersion -> Uninstall -> HALCON
```

- Using, e.g., the Windows Explorer, delete the directory

```
%ProfileRoot%\All Users\Start Menu\Programs\MVTec HALCON
```

with `%ProfileRoot%` depending on the used Windows version: C:\WINNT\Profiles (Windows NT) or set to C:\Documents and Settings (Windows 2000 and Windows XP).

- Delete the environment variables `HALCONROOT` and `HALCONIMAGES` in Start ▸ Settings ▸ Control Panel.

- Finally, delete the contents of the HALCON directory and all its subdirectories.

## 4.3 Problems Concerning Licenses

If you encounter problems with your HALCON license even though your license file exists and is located in the correct directory, a first step is always to check if the information identifying your network card or dongle matches the entries in the license file (see the corresponding sections in [chapter 3](#) on page 17). If the two do not match, please send the new identifying information to your distributor. See [section 4.3.1](#) if you encounter problems with extracting the identifying information.

[Section 4.3.2](#) contains information all around dongle drivers, e.g., how to check whether they are installed correctly.

If you have problems with a floating license, the first step is to check whether the entries in the license file that can be customized contain the correct information, especially the port number for the license manager daemon (see [section 3.3.3](#) on page 21). If the license file is correct, please refer to [section 4.3.3](#) on page 37, which explains how to check whether the floating license manager daemon was installed successfully and how to install it manually.

The FLEXlm End User's Guide can be obtained from <http://www.macrovision.com>.

### 4.3.1 Extracting Host IDs

- **The dialog Help ▸ About HDevelop does not show any host IDs**

If HDevelop fails to detect any host IDs although your computer does have a network board, a Pentium III, or a dongle, please try to extract the host IDs manually using the program `lmhostid` shipped together with the license manager FLEXlm. Under Windows NT/2000/XP, open a Windows Command Prompt<sup>1</sup>. Under Linux/UNIX, open a shell, change into the directory `$HALCONROOT/FLEXlm` and then into the subdirectory corresponding to your operating system (e.g., `i586-linux2.2` or `sparc-sun-solaris7`).

- To identify a computer by its **network card**, type `lmhostid -ether` (or just `lmhostid`). The output might look like this on a Windows system:

```
> lmhostid -ether
lmhostid - Copyright (C) 1989-2003 Macrovision Corporation
The FLEXlm host ID of this machine is "00e02958e36a"
```

If `lmhostid` returns `"ffffffff"` or `"0"` please see below.

- To check the **dongle ID**, type `lmhostid -flexid`. The output now might look like this:

```
> lmhostid -flexid
lmhostid - Copyright (C) 1989-2003 Macrovision Corporation
The FLEXlm host ID of this machine is "FLEXID=6-a6305af4"
```

The host ID must be identical to the one printed on the back of the dongle. If this is not the case, please see below.

<sup>1</sup>Do not start the program from the Windows Explorer. You *must* use a Command Prompt.

- `lmhostid -ether` returns "ffffffff" or "0"

If `lmhostid` returns "ffffffff" or "0", this usually indicates that you do not have a network card. If you do have one, on **Windows 2000 and Windows XP** systems check whether the client for Microsoft networks is installed in the dialog `Start > Settings > Network and Dial-up Connections > Local Area Connections > Properties`. If the client for Microsoft networks does not appear in the list install it by clicking `Install` and then selecting it from the list of clients.

On **Windows NT** systems check whether the NetBEUI and TCP/IP protocols are installed in the dialog `Start > Settings > Control Panel > Network > Protocol`. If the protocols do not appear in the dialog, please install these two protocols by adding them in the dialog.

Then, call `lmhostid` again or open HDevelop's dialog `Help > About` (see [section 3.1](#) on page 17) to check whether a correct host ID is found now.

- **Multiple IDs for only one network card**

On Windows 2000 and Windows XP systems, sometimes more than one ID is returned even if there is only a single network card. In this case, use the ID that remains when both the NetBEUI and IPX/SPX protocols are disabled. Alternatively, use the ID that does not change when you reboot your computer.

- **Licensing via network card does not work when network is not connected**

On Windows 2000 and Windows XP systems, you must disable Media Sense (DHCP) when no network is connected to the network card. Usually, it is sufficient to disable the DHCP protocol in this case. Further details can be found under <http://support.microsoft.com/support/kb/articles/Q239/9/24.asp>.

- `lmhostid -flexid` does not return the dongle ID

If `lmhostid` does not return the ID that is printed on the back of the dongle, check whether the dongle driver is installed and configured correctly as described in [section 4.3.2](#).

If the driver is installed but `lmhostid` still does not return the correct ID, please check the corresponding port (parallel or USB) of your computer, before requesting a new dongle.

- `lmhostid -ether` does not return all installed network card IDs under Linux

`lmhostid` returns only the ID of the network card that is configured as `eth0` on your Linux computer. To use another network card you must configure your computer that the desired network card is configured as `eth0`.

## 4.3.2 Dongle Drivers

### 4.3.2.1 Dongles for the Parallel Port (Windows only)


As described in [section 2.2.1](#) on page 6, you can let HALCON install the driver programs necessary for using dongles. You can check the success of this installation in the following system dialog, which should contain an entry called `Sentinel`.

- **Windows NT** : `Start > Settings > Control Panel > Devices`.

- **Windows 2000, Windows XP** : Start ▷ Programs ▷ Accessories ▷ System Tools ▷ System Info ▷ Software Environment, then select Drivers under Windows 2000 or System Drivers under Windows XP, respectively.

You can install, configure, and uninstall the dongle driver manually using an auxiliary program that is part of each HALCON installation. Open a Windows Command Prompt or the dialog Start ▷ Run and start the program

```
%HALCONROOT%\FLEX1m\i586-nt4\flexid6-7\win_nt\setupx86.exe
```

To install the driver, select the menu item Functions ▷ Install. Afterwards, reboot your computer. Then, the driver should appear in the dialog described above. **Please note that you need administrator privileges to install the driver!** If you try to install the driver without administrator privileges, the setup program might falsely state that the installation was successful. 

You might need to configure the driver if your computer has more than one parallel port.

Note that the HALCON uninstallation process does not uninstall the dongle driver as it might also be used by another application. However, the uninstallation removes the auxiliary program. Therefore, if you want to uninstall the driver, do so before uninstalling HALCON; of course, the program can still be found on the HALCON CD in the directory FLEX1m\i586-nt4\flexid6-7\win\_nt.

### 4.3.2.2 Dongles for the USB Port (Windows)

As described in [section 2.2.1](#) on page 6, you can let HALCON install the driver programs necessary for using dongles. You can check the success of this installation in the system dialog given in [section 4.3.2.1](#) on page 34, which should contain two entries called Hardlock and HASP.

Depending on your operating system, you can install, configure, and uninstall the dongle drivers manually using an auxiliary program that is part of each HALCON installation. Open a Windows Command Prompt or the dialog Start ▷ Run and execute the program %HALCONROOT%\FLEX1m\i586-nt4\flexid9\install.exe in one of the following ways:

hinstall	opens a dialog explaining how to use hinstall
hinstall -info	shows the status of the drivers
hinstall -i	installs the drivers
hinstall -remove	removes the drivers

**Please note that you need administrator privileges to install the drivers!** 

Note that the HALCON uninstallation process does not uninstall the driver as they might also be used by another application. However, the uninstallation removes the auxiliary program. Therefore, if you want to uninstall the drivers, do so before uninstalling HALCON; of course, the program can still be found on the HALCON CD in the directory FLEX1m\i586-nt4\flexid9.

### 4.3.2.3 Dongles for the USB Port (Linux)

The USB dongle daemon has been tested on the following Linux distributions: RedHat (8.x or 9.x) and SuSE (8.x or 9.x). Other distributions may work, but were not tested explicitly. The output of the following examples is specific to SuSE Linux.



**On SuSE 9.0 systems, unplugging and reconnecting USB devices can cause the hotplug daemon to hang.**

In order to install the USB dongle daemon on Linux, issue the following commands as root:

```
# cd $HALCONROOT/FLEX1m/$ARCHITECTURE/flexid9
# ./dinst bin
```

The installation script will install the dongle daemon and the corresponding start-up scripts.

```
-----
Copy AKSUSB daemon to /usr/sbin ...
Copy AKSUSB daemon startup script to /etc/init.d
Setting up to autostart AKSUSB daemon
create symbolic link 'S90aksusbd' to '/etc/init.d/aksusbd'
Trying to start AKSUSB daemon ...
Starting AKSUSB daemon                               done
Done
-----
```

To test, if the dongle daemon is running, enter the following command as root (where `/etc/init.d` is to be replaced by the directory reported when installing the dongle daemon):

```
# /etc/init.d/aksusbd status
Checking for AKSUSB daemon: OK
```

When the dongle is plugged into an USB port, it should be detected by the system automatically. To actually verify this, the output from the following command should contain a block similar to the one displayed below:

```
# cat /proc/bus/usb/devices
...
T: Bus=01 Lev=01 Prnt=01 Port=01 Cnt=01 Dev#= 3 Spd=1.5 MxCh= 0
D: Ver= 1.00 Cls=ff(vend.) Sub=00 Prot=00 MxPS= 8 #Cfgs= 1
P: Vendor=0529 ProdID=0001 Rev= 1.00
S: Manufacturer=AKS
S: Product=HASP4 USB 1.31
C:* #Ifs= 1 Cfg#= 1 Atr=80 MxPwr= 54mA
I: If#= 0 Alt= 0 #EPs= 0 Cls=ff(vend.) Sub=00 Prot=00 Driver=(none)
...
```

Look for lines containing the words AKS and HASP4 in the output. In order for the above command to work, make sure your kernel is configured to support the USB device filesystem. This should be the default setting for the major distributions. If the option is not included in your kernel configuration, a custom kernel has to be compiled with the corresponding option set. See your operating system manual for information about compiling a new kernel. In a menu-based kernel configuration, the corresponding option appears similar as in the following example:

```
Support for USB
...
--- Miscellaneous USB options
...
[x] Preliminary USB device filesystem
```

Alternatively, you can edit the file `.config` in the kernel source directory to contain the line

```
CONFIG_USB_DEVICEFS=y
```

To uninstall the dongle daemon, the start-up scripts and the daemon executable have to be removed (as root):

```
rm /usr/sbin/aksusbd
rm /etc/init.d/rc5.d/S90aksusbd
rm /etc/init.d/aksusbd
```

Again, the directory `/etc/init.d` depends on your Linux distribution. You can find the actual location of the start-up scripts with the command:

```
find /etc -name "aksusbd"
```

### 4.3.3 The License Manager Daemon

The license manager daemon (which is used in case of floating licenses) consists of two programs:

1. `lmgrd`: This is the main license manager daemon, which is provided by the licensing software FLEXlm. It handles the connections from the HALCON applications and passes them on to the second daemon.
2. `mvtecd`: This is the so-called *vendor daemon*, which is charged with keeping track of the HALCON applications currently being run.

The following sections explain how to install, (re-)start, and uninstall the license manager daemon on Windows NT/2000/XP (section 4.3.3.1) and Linux/UNIX (section 4.3.3.2 on page 39) platforms.

#### 4.3.3.1 Windows NT/2000/XP

##### How to Check Whether the License Manager Daemon is Running

To check whether the daemon was installed and started successfully, open the Windows dialog showing the state of the installed services:

- Windows NT: Open the dialog Start ▷ Settings ▷ Control Panel ▷ Services.
- Windows 2000 and Windows XP: Open the dialog Start ▷ Settings ▷ Control Panel ▷ Administrative Tools ▷ Component Services ▷ Services.

If the daemon was installed successfully by the setup program, the dialog contains the entry HALCON Licenses.

## How to Install the License Manager Daemon



If you did not install the license manager daemon via the setup as described in [section 2.2.1](#) on page 6, or if its installation failed, you can install it manually at a later time as described below. Note that **you need administrator privileges** for this procedure.

Open a Windows Command Prompt, change into the directory `%HALCONROOT%\FLEXlm\i586-nt4`, and type (*one* long command line):

```
installs -n "HALCON Licenses"  
-c "%HALCONROOT%\license\license.dat"  
-l "%HALCONROOT%\license\license.log"  
-e "%HALCONROOT%\FLEXlm\i586-nt4\lmgrd.exe"
```

The environment variable `HALCONROOT` points to the folder where you installed HALCON (see also [section A.2](#) on page 46). The quotes are necessary to handle paths that contain blanks.

To check whether the installation succeeded, (re-)open the Windows dialog showing the state of the installed services (see above), which now should contain the entry `HALCON Licenses` (or, to be more exact, an entry with the name you specified with the option `-n` or `FLEXlm license manager` if you leave out this option).

Note that if you installed the license manager daemon manually, you must also uninstall it manually before you uninstall HALCON. Otherwise it remains installed.

## How to Start the License Manager Daemon

After the installation, the daemon is not started automatically. You can start it by rebooting the computer. If you have administrator privileges, you can start it directly in the dialog which shows the state of the installed services (see above) by selecting it and then clicking the start button or icon; if this fails, please check whether the entries in the license file are correct.

## How to Restart the License Manager Daemon

Whenever you replace a floating license or modify it you must *restart* the license manager daemon. Another reason for restarting the license manager daemon is when you update to a HALCON release that uses a newer version of the FLEXlm license manager software, e.g., from HALCON 6.1.1 to 6.1.2. One possible method is to reboot the computer acting as the license server. If you have administrator privileges, you can also restart the daemon manually: Open the dialog showing the state of the installed services (see above), select the entry `HALCON Licenses` and then stop and start it again.

## How to Uninstall the License Manager Daemon



Please note that for the following procedure **you need administrator privileges**.

Before you can uninstall the license manager daemon, you must first stop it: Open the Windows dialog showing the state of the installed services (see above), select the entry `HALCON Licenses` and then click the stop button or icon. Note that under Windows 2000 and Windows XP an error message may appear, which can safely be ignored.

Now, open a Windows Command Prompt, change into the directory %HALCONROOT%\FLEXlm\i586-nt4, and type:

```
installs -r -n "HALCON Licenses"
```

To check whether the uninstallation succeeded, re-open the services dialog, which should not contain an entry HALCON Licenses anymore.

### 4.3.3.2 Linux/UNIX

#### How to Install the License Manager Daemon

On Linux/UNIX systems, the license manager daemons `lmgrd` and `mvtecd` are automatically “installed” in the subdirectory `FLEXlm/$ARCHITECTURE` of the directory you installed HALCON in (see [table A.1](#) on page 47 for the possible values of ARCHITECTURE).

#### How to Start the License Manager Daemon

The license manager daemon `lmgrd` must be started from the appropriate startup file (called, e.g., `/etc/init.d/boot.local`, `/sbin/init.d/boot.local`, `/etc/rc.boot`, `/etc/rc.local`, or `/etc/localrc`, please consult your system’s documentation). Add the following (long) line to this file, replacing the environment variables with their content (see [section A.2](#) on page 46):

```
$HALCONROOT/FLEXlm/$ARCHITECTURE/lmgrd -c $HALCONROOT/license/license.dat  
> $HALCONROOT/license/license.log 2>&1 &
```

If you copied `lmgrd` to another location, you must of course adapt the path to it accordingly. Please note, that you **should not start `lmgrd` with root privileges!**

`lmgrd` automatically starts the vendor daemon `mvtecd`.

#### How to Restart the License Manager Daemon

Whenever you replace a floating license or modify it you must restart the license manager daemon. Another reason for restarting the license manager daemon is when you update to a HALCON release that uses a newer version of the FLEXlm license manager software, e.g., from HALCON 6.1.1 to 6.1.2.

One possible method is to reboot the computer acting as the license server. If you have administrator privileges, you can also restart the daemon manually by stopping the process `lmgrd` and then restarting it with the line used in the startup file (see above).

#### How to Uninstall the License Manager Daemon

To uninstall the license manager daemon you must remove the corresponding entry in the startup file (see above) and stop the currently running `lmgrd` and `mvtecd` processes. This can be achieved by rebooting the computer that acts as the license server or by use of the `kill` command (please see your system’s documentation or ask your system administrator for advice).



## 4.4 Troubleshooting in HDevelop or HALCON Applications

This section explains some typical error messages when starting HDevelop or your own HALCON applications and their reasons.

- **Error using license file**

This error message might have several reasons:

- The file `%HALCONROOT%\license\license.dat` is missing and/or not readable.
- Your license is not valid on this machine.
- In case of floating licenses: There are too many applications using HALCON active, i.e., the maximum number of simultaneous HALCON applications (which is specified in the floating license) has been exceeded.

- **Lost connection to license server**

Verify that the license server is running. You may also check whether your machine is properly connected with the server. For this you may need to contact your system administrator.

- **No license for this operator**

The operator which you try to execute belongs to a HALCON module that is not licensed (see [section 3.4](#) on page 25). Obtain a new license including this module.

- **hdevelop: Command not found** (Linux/UNIX)

Check your system environment variable `PATH`. It must include the path `$HALCONROOT/bin/$ARCHITECTURE` (see [table A.1](#) on page 47 for the values of `ARCHITECTURE`).

- **lib\* : can't open file** (Linux/UNIX)

Check the system variable `LD_LIBRARY_PATH` (see [section A.2](#) on page 46).

- **No help files for package <package-name> in directory <directory>**

Possible reasons for this error message are:

- No files `%HALCONROOT%\help\*` (if the package name is “system”) or no help files in one of the user packages.
- If the package name is “system”: Wrong `HALCONROOT`.
- Check the file permissions. Probably HDevelop cannot access important files.

- **Help file for package <package-name> is corrupt**

Possible reasons for this error message are:

- If the package name is “system”: Inconsistent version of `%HALCONROOT%\help\*` or wrong `HALCONROOT`.
- If the package name is that of a user package: Inconsistent version of the help files of this package.

- **Can't open display** (Linux/UNIX)

If you see an error message like this you may have a wrong system variable `DISPLAY` and/or your program is not allowed to open a window by the specified X-server.

## 4.5 Miscellaneous Problems

- **No refresh of window content on a Linux/UNIX system**

On some Linux/UNIX systems the default behavior regarding occluded windows may be set in an inconvenient way for HALCON. The result is that if a window is temporarily occluded by another window, its content is not saved and restored anymore, i.e., windows remain “black” after uncovering. An example for this are all SuSE Linux distributions  $\geq 7.0$ . The corresponding property is called “backing-store”; you can check the current setting of this property by typing (the following example corresponds to a SuSE 8.2 Linux system):

```
xdpyinfo | grep backing-store
```

which should result in the output like

```
options:    backing-store YES, save-unders YES
```

if the window content is saved and restored. You can change this behavior by modifying the file `Xservers` residing in the directory `/usr/lib/X11/xdm` (or `/opt/kde3/share/config/kdm` in case of newer Linux versions), see your system’s documentation. Note, that you probably need root privileges to modify this file. Append the option `+bs` (i.e., “plus backing-store”) to the line that starts the local X server:

```
:0 local /usr/X11R6/bin/X :0 vt07 +bs
```

Now, stop and start the X server again (by using the appropriate commands or by rebooting your computer); the command `xdpyinfo` now should yield the output shown above.



# Appendix A

## More on the Installation

This appendix contains information about

- the installed file structure ([section A.1](#)),
- the relevant environment variables ([section A.2](#) on page 46), and
- special information for users of ActivisionTools ([section A.3](#) on page 49).

## A.1 The Installed File Structure

Let's take a look at the installed file structure in the directory %HALCONROOT%. In the following, the most important directories and files are described briefly. Please note that, depending on your installation, not all directories may be present.

- **FLEXlm:** This directory contains programs used for licensing in subdirectories corresponding to the different platforms (see [chapter 3](#) on page 17 and [section 4.3.3](#) on page 37).
- **bin:** This directory contains HALCON programs, for example HDevelop (Windows: `hdevelop.exe`; Linux/UNIX: `hdevelop`), again in subdirectories corresponding to the different platforms. For Windows, this directory also contains the DLLs of the HALCON libraries and the DLLs for the supported frame grabber interfaces.
- **calib:** This directory contains description files for the calibration plates, which you can use to calibrate your camera.
- **doc\html:** Here, you find the HTML documentation of the HALCON frame grabber interfaces (subdirectory `manuals`), the HTML version of the reference manuals (subdirectory `reference`), and a set of tutorials (subdirectory `tutorials`).
- **doc\pdf:** Here, you find the PDF version of the user's manuals (subdirectory `manuals`), the **Application Guide** (subdirectory `application_guide`), and of the reference manuals (subdirectory `reference`).
- **examples:** The subdirectories of this directory contain example programs for the different parts of the HALCON system:
  - ▷ **application\_guide:** Examples referenced in the **Application Guide**.
  - ▷ **c:** Examples for using HALCON within the programming language C (see also the Programmer's Guide, [chapter 15](#) on page 109).
  - ▷ **cpp:** Examples for using HALCON within the programming language C++ (see also the Programmer's Guide, [chapter 6](#) on page 55).
  - ▷ **c#:** Examples for using HALCON within the programming language C# (see also the Programmer's Guide, [chapter 10](#) on page 81).
  - ▷ **delphi:** Examples for using HALCON within Borland Delphi.
  - ▷ **fg\_integration:** Example programs for frame grabber interfaces (see also the [Frame Grabber Integration Programmer's Manual](#)).
  - ▷ **extension\_package:** The example user package `halconuser` (see also the [Extension Package Programmer's Manual](#)).
  - ▷ **hdevelop:** Examples for using HALCON in HDevelop:
    - ▷ **Applications:** Examples that show how to realize machine vision applications.
    - ▷ **Control ...XLD:** Examples for the HALCON operators, in subdirectories following the operator hierarchy as in the Reference Manuals or in HDevelop's menu `Operators`.
    - ▷ **Manuals\HDevelop:** Examples referenced in the [HDevelop User's Manual](#).

- ▷ **mfc**: Examples for using HALCON together with MFC (see also the sections in the [Programmer's Guide](#) mentioned above).
- ▷ **motif**: Examples for using HALCON together with Motif (see also the Programmer's Guide, [chapter 6](#) on page 55).
- ▷ **quick\_guide**: Examples referenced in the [Quick Guide](#).
- ▷ **vb**: Examples for using HALCON within Visual Basic, together with the example programs described in the Programmer's Guide, [chapter 10](#) on page 81.
- ▷ **vb.net**: Examples for using HALCON within Visual Basic .NET (see also the Programmer's Guide, [chapter 10](#) on page 81).

Please note that the examples should not be used directly for two reasons: First, on most platforms only the user who installed HALCON is allowed to save a (modified) example program. More importantly, on some operating systems not all users have the permission to create new files in the example subdirectories; this causes errors in example programs that contain operators that write data to files. To experiment with examples we therefore recommend to create a private copy in your own working directory.

- **filter**: This directory contains predefined filter masks.
- **help**: This directory contains information for the online help of HDevelop as well as the file `examples.xml` that contains the index data for the `Open example program...` dialog in HDevelop.
- **images**: This directory contains example images and, in subdirectories, image sequences. These images are used by the example programs described above.
- **include**: This directory contains the header files that are necessary to use HALCON within the programming languages C or C++.
- **lib**: This directory contains the HALCON libraries and under Linux/UNIX also the libraries for the supported frame grabber interfaces (Windows: file extension `.lib`; Linux/UNIX: file extension `.so`), again in subdirectories corresponding to the different platforms.
- **license**: The license file `license.dat` must be placed in this directory (see [section 3.1](#) on page 17).
- **lut**: This directory contains predefined look-up tables.
- **ocr**: This directory contains trained fonts.

## A.2 HALCON's Environment Variables

Most of the configuration necessary to work with HALCON amounts to setting environment variables, e.g., to tell HALCON the directories where to find images or extension packages etc. These environment variables are described below, after some information regarding the different platforms.

### Setting Environment Variables Under Windows NT/2000/XP

The installation program Setup.exe automatically sets the necessary environment variables, e.g., %HALCONROOT%, %HALCONIMAGES%, and %PATH% (see below). To take a look at these settings, open the dialog Start ▸ Settings ▸ Control Panel ▸ System and select Environment. You can add or modify a variable by entering the name of a variable and the desired value. If a value consists of multiple items, e.g., the variable %PATH%, which may contain multiple directories, those items must be separated by *semicolons*. Please note that in order to modify variables set during the HALCON installation you need administrator privileges!

### Setting Environment Variables Under Linux/UNIX

As described in [section 2.2.2](#) on page 8, you must set the necessary environment variables in a login script or a shell resource script.

### HALCON-Specific Environment Variables

- HALCONROOT  
This is the most important environment variable. It designates the directory where HALCON is installed. A typical path is, for example, C:\Program Files\MVTec\HALCON (Windows NT/2000/XP) or /usr/local/halcon (Linux/UNIX). Based on this variable, the system switches to subdirectories, which are important for running HALCON. Some of them are listed below; the HALCON file structure is described in [section A.1](#) on page 44.
  - %HALCONROOT%\help  
The operator data base is located in this directory. It is accessible by all HALCON programs to obtain information about HALCON operators.
  - %HALCONROOT%\doc\html\reference\hdevelop  
HDevelop uses this directory for online help, which can be displayed by a suitable HTML browser.
  - %HALCONROOT%\license  
This directory contains the license file necessary for using HALCON (see [chapter 3](#) on page 17).
  - %HALCONROOT%\images  
If the variable HALCONIMAGES (see below) is not set the system looks for image files in this directory.
- HALCONIMAGES  
The system uses this environment variable to search for image files specified by a relative path. As a rule it contains several directory names and possibly including the CD ROM drive, separated by semicolons (Windows NT/2000/XP) or colons (Linux/UNIX).

ARCHITECTURE	Operating System (Platform)	Compiler
i586-nt4	Windows NT/2000/XP on Pentium (or compatible)	Visual Studio
i586-linux2.2 i586-linux2.2-gcc32	Linux 2.2/2.4 on Pentium (or compatible)	gcc 2.95 gcc 3.2/3.3
ia64-linux2.4	Linux 2.4 on Itanium 2	gcc 3.3
sparc-sun-solaris7	Solaris 7 or higher on SPARC workstations	CC
mips-sgi-irix6.5	IRIX 6.5 on SGI workstations (Mips processors)	CC
alpha-compaq-osf5.1	Tru64 UNIX 5.1 on Alpha workstations	cxx

Table A.1: Values of ARCHITECTURE for the currently supported platforms.

- ARCHITECTURE

This variable designates the used platform by an abbreviation (e.g., `i586-nt4` or `i586-linux2.2`; its syntax is: *processor-hardware\_vendor-operating\_system*). ARCHITECTURE appears in several directory paths: Executable HALCON programs, e.g. `hdevelop`, and DLLs, e.g. `halcon.dll` (Windows NT/2000/XP only), reside in `%HALCONROOT%\bin\%ARCHITECTURE%`. On Windows NT/2000/XP systems, this path is therefore automatically included in the environment variable `PATH`; on a Linux/UNIX system, you must include it in your login script.

The libraries that you need for linking programs, e.g., `halcon.lib` (Windows NT/2000/XP) or `halcon.so` (Linux/UNIX) reside in the directory `%HALCONROOT%\lib\%ARCHITECTURE%`.

[Table A.1](#) gives an overview of the currently supported platforms and the corresponding values of ARCHITECTURE.

- HALCONEXTENSIONS

This is a list of directories in which user-defined extension operators (so-called extension packages) are kept. Each package consists of a number of operators linked into a shared library, plus the additional operator documentation in help files and HTML files. See [section 2.9](#) on page 14 for information on how to install an extension package, and the [Extension Package Programmer's Manual](#) for details on creating your own extension packages.

- HALCONSPY

If this environment variable is defined (regardless of the value) *before you start* a HALCON program, the HALCON debugging tool HALCON Spy is activated. This corresponds to call the HALCON operator `set_spy` with the parameters "mode", "on" *within* a HALCON program. The main difference between the two modes for activating HALCON Spy is that by defining `%HALCONSPY%` it is possible to monitor an already linked HALCON program during runtime without modifications. For further information on how to use HALCON Spy and how to parameterize it via this environment variable please refer to the Programmer's Guide, [section 2.1](#) on page 9.

## General Environment Variables

- PATH

**Windows NT/2000/XP** : During the installation, the directories `%HALCONROOT%\bin\i586-nt4` and `%HALCONROOT%\FLEX1m\i586-nt4` are automatically added to the system variable `PATH`.

**Linux/UNIX** : If you want to start HDevelop from an arbitrary directory, you must include the HALCON program path `$HALCONROOT/bin/$ARCHITECTURE` in the system variable `PATH`.

- `LD_LIBRARY_PATH` (Linux/UNIX only)

Please include the HALCON library path `$HALCONROOT/lib/$ARCHITECTURE` in the system variable `LD_LIBRARY_PATH`. This is necessary both for running HDevelop and for creating stand-alone applications.

Note that under IRIX you have to use the environment variable `LD_LIBRARYN32_PATH` instead of `LD_LIBRARY_PATH`.

- `DISPLAY` (Linux/UNIX only)

The system uses this environment variable to open windows. It is used in the same way as for other X applications.

- `HOME` (Linux/UNIX only)

This system variable points to your home directory.

## A.3 Information for Users of ActivVisionTools

ActivVisionTools are based on HALCON, to be more exact on HALCON/COM. What makes matters more complicated is that the used HALCON functionality is not compiled into ActivVisionTools; instead, ActivVisionTools use HALCON in the same way as any other HALCON application, i.e., they expect HALCON to be installed (if not already installed, the ActivVisionTools setup installs it), locate the corresponding DLL `halconx.dll` via the registry and use it.

As only one instance of `halconx.dll` can be registered, this means that ActivVisionTools and HALCON applications use the same HALCON installation. This must be kept in mind when you want to use both HALCON and ActivVisionTools on the same computer. Of course, as long as you want to use the HALCON release that your ActivVisionTools release is based on, there is no problem. However, when you update or upgrade HALCON or ActivVisionTools, the issue of compatibility gets more complicated.

Therefore, the HALCON setup program checks whether there is an ActivVisionTools installation on the computer before installing HALCON on it and informs you about its compatibility. Some HALCON releases even offer to replace the ActivVisionTools OCX with a compatible one (see below).

Compatibility	AVT 1.0 - 1.3	AVT 2.0	AVT 2.1	AVT 2.2	AVT 3.0	AVT 3.1
HALCON 6.0	based on	×				
HALCON 6.0.1	×	based on				
HALCON 6.0.2	(×)	(×)				
HALCON 6.0.3	(×)	× <sup>*)</sup>				
HALCON 6.0.4	(×)	× <sup>*)</sup>				
HALCON 6.1		× <sup>*)</sup>	based on	(×)		
HALCON 6.1.1		× <sup>*)</sup>	×	based on		
HALCON 6.1.2		× <sup>*)</sup>	×	×		
HALCON 7.0				× <sup>*)</sup>		
HALCON 7.0.1				× <sup>*)</sup>	based on	
HALCON 7.0.2				× <sup>*)</sup>	×	
HALCON 7.0.3				× <sup>*)</sup>	×	

(\*) via replacement OCX

Table A.2: Compatibility between the different HALCON and ActivVisionTools releases.

Table A.2 shows which ActivVisionTools releases are compatible to which HALCON releases. In principle, the same rules as described in [section 1.2](#) on page 2 for the HALCON releases themselves apply, with some differences:

- **Same HALCON version**

If an ActivVisionTools release is based on a certain (maintenance release of a) HALCON ver-

sion, it should also be compatible to all other maintenance releases based on the same HALCON version. For example, ActivVisionTools 2.1, which is based on HALCON 6.1, is compatible to HALCON 6.1.1 and 6.1.2.

Sometimes, however, new HALCON maintenance releases are not fully downward compatible because of technical reasons. For example, HALCON 6.0.2 introduced a change that limited the compatibility of ActivVisionTools 1.0 - 2.0.

To fix such incompatibilities, HALCON releases may provide a replacement OCX for some ActivVisionTools versions. For example, HALCON 6.0.3 and 6.0.4 provided a replacement OCX for ActivVisionTools 2.0. If the HALCON setup program detects such an ActivVisionTools version, it offers to replace the OCX automatically. Typically, you will choose this option. If not, you can replace it manually as described below.

- **Higher HALCON version**

An ActivVisionTools release is by default not compatible to HALCON versions that are higher than its base. For example, ActivVisionTools 1.3, which is based on HALCON 6.0, is not compatible to HALCON 6.1.

When you install such a higher HALCON version, the setup program warns that by continuing you will disable your ActivVisionTools installation. If you still want to use your ActivVisionTools installation, you must also keep your old HALCON installation and switch back to it as described in [section 2.7](#) on page 13. Note that if you only want to evaluate the new HALCON version, you can minimize the switching effort by installing only the demo version: In this case there is no need to re-register the HALCON/COM interface library `halconx.dll`.

Again, some HALCON releases provide a replacement OCX. For example, all HALCON 6.1.x releases provided a replacement OCX for ActivVisionTools 2.0, which was based on HALCON 6.0.1; HALCON 7.0 provides a replacement OCX for ActivVisionTools 2.2.

- **Lower HALCON version**

An ActivVisionTools release is not compatible to HALCON versions that are lower than its base. For example, ActivVisionTools 2.1, which is based on HALCON 6.1, is not compatible to HALCON 6.0.x.

When you install such a lower HALCON version, the setup program warns that by continuing you will disable your ActivVisionTools installation. If you still want to use your ActivVisionTools installation, you must also keep your its HALCON installation and switch back to it as described in [section 2.7](#) on page 13.

## How to Replace the ActivVisionTools OCX Manually

Open a Windows Command Prompt and change into the subdirectory `bin\i586-nt4` of the root directory of your ActivVisionTools installation. Unregister `ActivVTools.ocx` by typing

```
regsvr32 /u ActivVTools.ocx
```

Then delete the OCX (or rename it) and replace it by the new one, which can be found on the CD in the directory `misc\i586-nt4\avt22`, or in the ActivVisionTools download area (<http://www.activ-vision-tools.com/download>). Register the new OCX by typing

```
regsvr32 ActivVTools.ocx
```

As an alternative to the Command Prompt, you can unregister and register the OCX via the dialog Start > Run together with the Windows Explorer: In the latter, “open” the directory `bin\i586-nt4` of the root directory of your ActivVisionTools installation. Now, type `regsvr32 /u` the dialog Run and then drag and drop `ActivVTools.ocx` from the Explorer into the dialog, where it automatically appears with the full path. To execute the command, click OK. Then, replace the OCX with the new open, type `regsvr32` in the dialog Run, drag and drop `ActivVTools.ocx` from the Explorer into the dialog, and again click OK.

If you decide to switch back to an older HALCON version as described in [section 2.7](#) on page 13, you must also unregister the new OCX and register the old one; if you deleted it you can recover it from the corresponding directory on the ActivVisionTools CD.



# Index

- ActivVisionTools, [2](#), [6](#), [13](#), [49](#)
- ARCHITECTURE, [9](#), [15](#), [23](#), [39](#), [40](#), [47](#)
- COM, [13](#)
- demo version (HALCON), [1](#)
- development version (HALCON), [1](#)
- DHCP, [34](#)
- directories
  - bin, [40](#), [44](#), [47](#), [48](#)
  - calib, [44](#)
  - doc, [44](#), [46](#)
  - examples, [44](#)
  - filter, [45](#)
  - FLEXlm, [33](#), [35](#), [38](#), [39](#), [44](#), [47](#)
  - help, [45](#), [46](#)
  - images, [45](#), [46](#)
  - include, [45](#)
  - lib, [45](#), [47](#), [48](#)
  - license, [2](#), [17](#), [19](#), [21](#), [24](#), [26](#), [27](#), [45](#), [46](#)
  - lut, [45](#)
  - ocr, [45](#)
- DISPLAY, [40](#), [48](#)
- dongle, [18](#), [21](#), [33](#)
  - device driver
    - installation, [7](#), [35](#)
    - uninstallation, [12](#), [35](#)
  - parallel port, [21](#), [34](#)
  - troubleshooting, [34](#)
  - USB, [21](#), [35](#)
- environment variables
  - ARCHITECTURE, [9](#), [15](#), [23](#), [39](#), [40](#), [47](#)
  - DISPLAY, [40](#), [48](#)
  - HALCONEXTENSIONS, [13](#), [14](#), [47](#)
  - HALCONIMAGES, [9](#), [13](#), [14](#), [32](#), [46](#)
  - HALCONROOT, [8](#), [9](#), [12](#), [13](#), [30](#), [32](#), [38](#), [40](#), [46](#)
  - HALCONSPY, [47](#)
  - HOME, [13](#), [48](#)
  - LD\_LIBRARY\_PATH, [9](#), [13–15](#), [40](#), [48](#), [48](#)
  - PATH, [9](#), [13](#), [15](#), [40](#), [47](#)
  - extension package (HALCON), [14](#), [47](#)
    - installation, [5](#), [14](#)
  - File menu (HDevelop)
    - Modules, [25](#)
  - frame grabber interfaces (HALCON)
    - installation, [5](#), [14](#)
  - full version (HALCON), [1](#)
  - get\_modules (GetModules), [25](#)
  - HALCON
    - auxiliary tools
      - lmgrd, [37](#), [39](#)
      - lmhostid, [33](#), [34](#)
      - mvtecd, [23](#), [37](#), [39](#)
      - reg\_halconx, [29](#), [30](#)
    - demo version, [1](#)
    - directories
      - bin, [40](#), [44](#), [47](#), [48](#)
      - calib, [44](#)
      - doc, [44](#), [46](#)
      - examples, [44](#)
      - filter, [45](#)
      - FLEXlm, [33](#), [35](#), [38](#), [39](#), [44](#), [47](#)
      - help, [45](#), [46](#)
      - images, [45](#), [46](#)
      - include, [45](#)
      - lib, [45](#), [47](#), [48](#)
      - license, [2](#), [17](#), [19](#), [21](#), [24](#), [26](#), [27](#), [45](#), [46](#)
      - lut, [45](#)
      - ocr, [45](#)
    - extension package, [14](#), [47](#)
      - installation, [5](#), [14](#)
    - file structure, [44](#)

- frame grabber interfaces
  - installation, [5, 14](#)
- full (development) version, [1](#)
- installation, [5](#)
  - add parts, [5, 11](#)
  - first time, [5, 6](#)
  - troubleshooting, [29](#)
  - update, [5, 11](#)
  - upgrade, [5, 11](#)
- language interfaces
  - HALCON/COM, [13](#)
- libraries
  - halconc.lib, [15](#)
  - halconcpp.lib, [15](#)
  - halconx.dll, [29, 30](#)
  - libhalcon.so, [15](#)
  - libhalconc.so, [15](#)
  - libhalconcpp.so, [15](#)
- maintenance release, [2, 11](#)
- modules (runtime version), [1, 3](#)
  - Foundation, [26](#)
- runtime version, [1](#)
- uninstallation, [5, 12](#)
  - troubleshooting, [30](#)
- update, [5, 11](#)
- upgrade, [5, 11](#)
- version, [2, 2, 11](#)
- HALCON Spy, [47](#)
- halconc.lib, [15](#)
- halconcpp.lib, [15](#)
- HALCONEXTENSIONS, [13, 14, 47](#)
- HALCONIMAGES, [9, 13, 14, 32, 46](#)
- HALCONROOT, [8, 9, 12, 13, 30, 32, 38, 40, 46](#)
- HALCONSPY, [47](#)
- halconx.dll, [29, 30](#)
- HDevelop
  - menu File, [25](#)
  - menu Operators, [44](#)
  - troubleshooting, [40](#)
- HOME, [13, 48](#)
- installation
  - dongle device driver, [7, 35](#)
  - HALCON, [5](#)
    - add parts, [5, 11](#)
    - first time, [5, 6](#)
    - troubleshooting, [29](#)
    - update, [5, 11](#)
    - upgrade, [5, 11](#)
  - HALCON extension package, [5, 14](#)
  - HALCON frame grabber interfaces, [5, 14](#)
  - license manager daemon, [7, 37](#)
- language interfaces
  - HALCON/COM, [13](#)
- LD\_LIBRARY\_PATH, [9, 13–15, 40, 48, 48](#)
- LD\_LIBRARYN32\_PATH, [14, 15, 48](#)
- libhalcon.so, [15](#)
- libhalconc.so, [15](#)
- libhalconcpp.so, [15](#)
- license, [7](#)
- license (HALCON)
  - bound to dongle, [18, 21, 33](#)
    - troubleshooting, [34](#)
  - bound to network card, [18, 20, 23, 26, 27, 33](#)
    - troubleshooting, [34](#)
  - compatibility, [2, 2, 19](#)
  - development license, [3, 17, 20, 21, 23, 27](#)
  - evaluation license, [3, 17, 19](#)
  - floating license, [3, 10, 18, 21, 27, 38, 39](#)
  - host ID, [18](#)
  - license file (license.dat), [17, 19–21, 23, 26, 27, 40, 46](#)
  - license key, [19, 20, 24, 26](#)
  - license manager daemon
    - installation, [7, 37](#)
    - restart, [24, 37](#)
    - uninstallation, [37](#)
  - license server, [22, 23](#)
  - node-locked license, [3, 20, 21, 25, 27](#)
  - runtime license, [3, 25](#)
  - upgrade, [17, 27, 27](#)
  - vendor daemon, [7, 22, 23](#)
- Linux, [8, 11–15, 46](#)
- lmgrd, [37, 39](#)
- lmhostid, [33, 34](#)
- maintenance release (HALCON), [2, 11](#)
- menu bar (HDevelop)
  - File, [25](#)
  - Operators, [44](#)
- modules (HALCON, runtime version), [1, 3](#)
  - Foundation, [26](#)
- mvtecd, [23, 37, 39](#)

network card, 18, 20, 23, 26, 27, 33  
troubleshooting, 34

operating systems  
Linux, 8, 11–15, 46  
UNIX, 8, 11–15, 46  
Windows, 6, 11–13, 15, 46  
NT, 32, 34, 37  
2000, 32, 34, 35, 37, 38  
XP, 32, 34, 35, 37, 38

Operators menu (HDevelop), 44

PATH, 9, 13, 15, 40, 47

Philips TriMedia, 4

reg\_halconx, 29, 30

runtime licenses, 17

runtime version (HALCON), 1  
modules, 1, 3

set\_spy (SetSpy), 47

system requirements, 4

terminal server, 20, 21

troubleshooting  
HALCON  
applications, 40  
installation, 29  
uninstallation, 30  
HDevelop, 40  
license (HALCON)  
bound to dongle, 34  
bound to network card, 34  
troubleshooting network card linux, 34

uninstallation  
dongle device driver, 12, 35  
HALCON, 5, 12  
troubleshooting, 30  
license manager daemon, 37

UNIX, 8, 11–15, 46

vendor daemon, 39

version (HALCON), 2, 2, 11

Windows, 6, 11–13, 15, 46  
NT, 32, 34, 37  
2000, 32, 34, 35, 37, 38  
XP, 32, 34, 35, 37, 38