MRC Motion Tracking

User Manual

Version 1.1

Table of contents

1. Introduction

Tracking the movement of patients is becoming increasingly common in the medical field. The MRC motion tracking software offers a simple solution for marker-based optical tracking. It makes use of one or two video cameras. Once the system is calibrated, it reliably and precisely provides the subject's position and movements.

The software offers two different tracking algorithms, the mono and the stereo algorithm. The mono algorithm stands out with easy handling and fast tracking and the stereo algorithm with its high precision and reliability.

2. System requirements

Since the tracking algorithms benefit from a high calculation speed, we recommend using the computer or laptop solely for the motion tracking. The software is designed for the cameras of MRC Systems. It can be used with two platforms, analog cameras with frame grabbers or digital cameras based on the GigE Vision standard, also known as GigE cameras.

In case of analog cameras the images are read via a frame grabber and each camera is bound to such a frame grabber. The software was tested with the frame grabbers DFG/USB2Pro, DFG/SV1/PCI and DFG/SV1/PCIe of TheImagingSource. As the name indicates, you need a USB, a PCI or a PCIe input at your computer or laptop. For stereo tracking you can either use two of these frame grabbers or the DFG/MC4/PCIe with 4 input channels.

GigE cameras receive the images of the HighResolution cameras via the GigE vision protocol. You need a Gigabit Ethernet network card, preferably with an Intel chip set and jumbo frame support (9 kByte), in your computer or laptop. A dedicated network card must be used for each camera in use.

To ensure smooth tracking, the following minimum computer requirements must be met:

Camera connection	• analog: framegrabbe for each camera • digital: dedicated GigE network card for each camera
Processor	Intel I7 or equivalent
Working memory	>8 GB
Graphic card	dedicated graphic card, e.g. NVidia 730 Gforce or better
Disc memory	HDD or SSD with free space of 1 GB
Operating system	• Windows 10, 64bit • Linux Ubuntu

3. Installation

Admin rights are required for the installation. Please make sure that an authorized person is present during the installation. On Windows you can start the installation by running the Microsoft Installer *.msi* file. On Linux you can unpack the archive in the console with *tar -xzf mrc-motion-trackig-1.0-Bxxxx.tar.gz* and install the Debian package with *dpkg -i mrc-motion-tracking -1.0-bxxxx-X64.deb*. (The variable xxxx depends on your software version.)

If you use HighResolution cameras, please check the settings of the firewall after installation. The installed motion tracking software must be able to communicate with the cameras.

To install the frame grabber driver for the analog cameras you have to open the directory *usbconvcxn_x.x.x_tis* (USB2Pro) or *FrameGrabberDriver\bt878cap_xxx.tis* (PCI/PCIe), execute the file *drvInstaller.exe* and follow the instructions. The installer will ask for administrator rights to install the driver. Again, the xxx depend on the driver version.

4. Setup

When the program starts, the setup menu opens (see figure 5). Here you can set the desired setup and load existing setups. Depending on the tracking algorithm, either one or two cameras can be selected. Make sure that each camera is only selected once. As soon as you click on a camera its frame rate and resolution are displayed.

You can open the system path by the **Open** button at each setup item. Then you can load a file with saved values. The **Clear** button removes the path. If no path is selected, the system's default data will be loaded. If you want to load a saved calibration, please select the folder in which the calibration files are saved.

If **Save as default** is checked, the settings and selected files are automatically loaded when the program starts. If this fails, the setup menu is displayed. If you want to change the setup later, you can reach it via the ⚙ menu with the button **Setup**.

Figure 1: Settings of Setup (example)

After you have confirmed the settings with the **Apply** button, the main software interface opens. Before you can start the tracking you have to calibrate the camera. You will find the option under ⚙. A more detailed explanation of the calibration can be found in chapter 6. Before you perform the calibration, make sure that the lens of the camera is adjusted correctly, because after a change of the focussing of the lens a calibration is necessary again.

5. Cameras

To adjust the camera settings you have to open the ▣ menu. Depending on the selected camera, you can set different parameters. If you want to save the settings afterwards to get them loaded directly at the software start, you can use the button *Save camera setup*. If your settings do not lead to the desired performance, you can also load an existing setup file with the button *Load camera setup*. If you want to save the settings of an experimental setup you can do this in a file which you can save on your computer.

Depending on the experimental setup, it might be necessary to rotate the the camera images. This is necessary if a camera had to be mounted in a rotated orientation e.g. due to limited space. You can select the *flip angle* in the camera settings (see figures 2 and 3).

5.1. 12M camera

The 12M camera has three parameters which you can control, the ***Brightness, Contrast*** and ***Saturation***. With these parameters you can optimize the image to obtain a better tracking result. All parameters can be changed between 0 and 100 %.
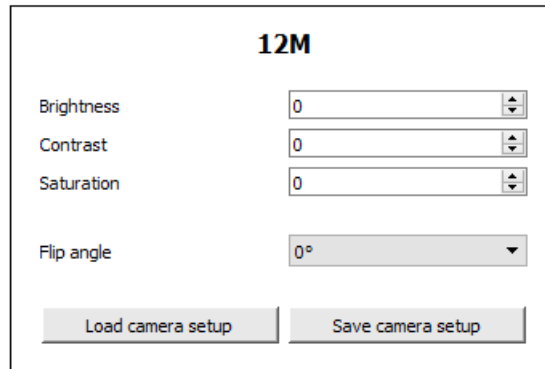

Figure 2: Settings of a 12M camera (example)

5.2. HighResolution camera

The HighResolution camera has two parameters which can be optimized for tracking, the ***Gain*** and the ***Exposure time***.

The ***Gain*** can be adjusted between 0 and 3, where 0 is the lowest and 3 is the highest gain. The effect of any changes can be immediately seen in the live view. The values for the ***Exposure time*** are given in microseconds (µs). The ExposureTime must not exceed the number of FPS. You can also see its changes immediately.
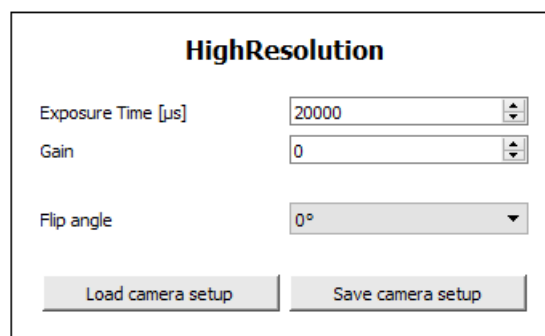
Figure 3: Settings of a GigE camera (example)

In case of connection problems with the HighResolution camera, an additional parameter is available in the setup file of the camera. You can change it when you open the file. The parameter ***packet delay*** sets the waiting time which the GigE interface waits for a new image. This parameter is in µs and can be increased up to the frame rate.

If you want to change the FPS number of the camera, please use the external program Sphinx GEV Viewer from MRC. The program is included in the installation medium of the motion tracking software.

6. Calibration

Before you can use the tracking you have to do a calibration which determines the cameras' characteristics which is needed to calculate a correct result. The section for the calibration can be found in the ⚙ menu.

The calibration is based on the recording of a checkerboard. You can print out a checkerboard by yourself. Its geometry can be adjusted with the parameters [number of] *total squares in x direction,* [number of] *total squares in y direction* and *square size [mm]*.

Before you start the calibration you have to capture input images for the calculation. For this purpose you have to take several images of the checkerboard. If you use two cameras, two images are recorded simultaneously. We recommend to take more than six images per camera.

Please use the button *Capture calibration image* to capture the images. It opens the system path and asks for a name for the captured image. All images of one camera have to be saved in the same folder. If you calibrate two cameras, each camera must have a separate folder. Once you have captured the images, you can open the folders using the *Open* button in the *Calibration* section.

With the *Show* check box you can then view the images and use the arrow keys to browse through them. In this way you can prove that in case of two cameras the checkerboard is visible in both images.

Make sure that all images from one camera are in the same folder.

If there are several cameras, the images must be in different folders.

Good calibration images are characterized by the following specs:
1) The checkerboard must be completely visible in all images.
2) The checkerboard must not be covered by other objects such as the fingers holding the board.
3) The position of the checkerboard must differ clearly between the images.
4) The checkerboard must not be overshadowed.
5) The surface of the checkerboard pattern must be uniformly illuminated.

If the calibration images are okay you can start the calibration with the button *Calibrate*. After a short time the result of the calibration will be displayed in a text field. Now you have several possibilities. You can dismiss the calibration with the button *Cancel* if the result isn't good. Good values are RMS values below 0.5. If you are satisfied with the calibration you can save it in the file path. It is set as the default in the setup with the button *Save*.

If you want to use the calibration for tracking you can, press *Apply*. Once you have pressed the button, the text field disappears and the calibration is completed. Now you can start tracking.

Whenever the camera lens was moved, the calibration is no longer correct.
If two cameras are used and their relative position to each other was changed, the calibration is no longer correct.
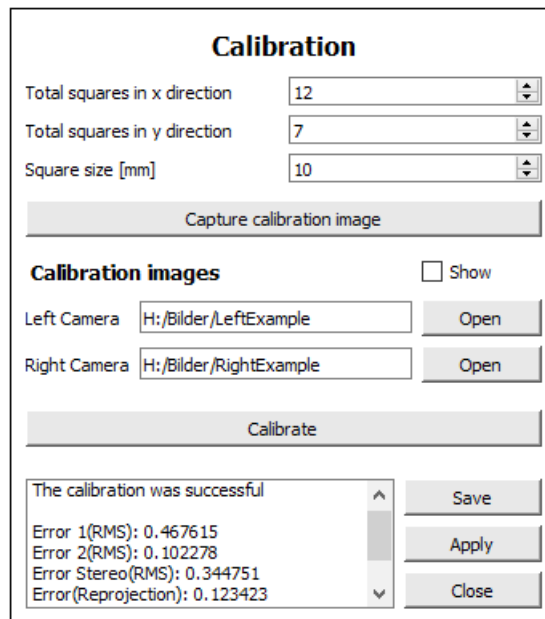
Figure 4: Settings of Calibration (example)

7. Tracking

Once you have a calibration you can activate the tracking in ⌖ with the switch. Now the selected algorithm starts searching for its markers. You have several parameters to optimize the stability and accuracy as well as the display (see figures 5 and 6).

(1) The most important one for detection is the **threshold**. This parameter specifies the **threshold** at which the images are filtered. You can view the filtered images by checking the **Show** checkbox. Adjust the parameter so that the marker is clearly visible with its properties.

(2) If a marker is lost during tracking, the algorithm waits for a few frames until it looks for a new marker. This time is defined by the parameter **Redetection rate**. If you have regular losses, you should set the **Redetection rate** to *1*.

(3) Both tracking algorithms, mono and stereo, allow to define search windows. With the option **Region of interest** you can use the mouse to drag the window into the camera images. If you want to remove it, you can click on **Remove**.

(4) During saving you have the option of **smoothing**. You can activate this option with the checkbox **Activate**. With the parameter **Amount of smoothed values** you can specify how many tracking results of the past are used to perform the smoothing.

(5) If you want to load or save your tracking settings you can do this by clicking the buttons **Load tracking setup** and **Save tracking setup**. You can save it in the same file as you have saved your camera setup or a separate file, whatever you prefer.

7.1. Mono tracking

The search of markers in the mono tracking algorithm uses the marker properties. A more detailed description of the structure of markers can be found in the document "Markers – Description". During the tracking make sure that the **threshold** is adapted to the environment, especially the light conditions.

You can set the marker properties in the section **Detection**. The **Marker Size** defines the length of the edge of the black base area. Press the **Open** button in **Marker Configuration** to open a separate window where you can change the marker properties.

Figure 5: Settings of Mono Tracking (example)

7.2. Stereo Tracking

The stereo tracking algorithm is based on the principle of triangulation of the individual blobs of the markers. The spatial position of each marker is determined from the positions of the markers in both images. The position of the blobs of the markers must be known and is read from the *Marker.yml* file in the document path when the program starts. The exact description of the file and the generation of a marker can be found in document "Markers – Description".

While the tracking algorithm is running you can vary three specific parameters for the stereo algorithm.

(1) The marker precision, which indicates in mm how far the triangulated blobs may deviate from the values specified in the marker.yml file. As soon as a marker is detected, this precision is displayed in the left camera image.

(2) The correlation tolerance, which specifies the spatial tolerance at which a possible correlation of two blobs is present.

(3) The pixel precision which is the tolerance in mm how close two blobs may be to each other to still be detected separately.

Figure 6: Settings of Stereo Tracking (example)

7.3. Display

In the Display section you can control, other display options (see figure 7). The **Display images to be skipped** indicates how many images are left out for display. It doesn't effect the tracking, but can be used if the computer is working on its limit.

Figure 7: Settings of Display (example)

With the checkbox **Coordinate system** you can insert a coordinate system in the images. It is calculated from the result and is projected into the images of the first camera. It is drawn at the origin of your marker and defines the x, y and z axis.

The **Tracking in color** checkbox shows the detected blobs of the marker and the coordinate system in colors. It does not convert the original images into color images.

 Be aware that this option requires high processing power and can lead to frame drops.

8. External Control

To provide the tracking data to other programs, the motion tracking software offers a TCP/IP interface. The settings can be found in the menu ⚙. In the **Connection** section you can open an interface at different ports. The motion tracking software acts as a TCP server that waits for an incoming connection. You can specify a desired **port** in the mask in the lower section. You can also enter a **password** that the other side has to send at the beginning of the connection. The **password** is optional. As soon as you press *Create*, the port opens and the external program can connect. It will show the opened port with the IP address of the motion tracking software in a small section. As soon as you press *Close* in the section, the connection and port will be closed. The protocol that can be transmitted after a connection is explained in a separate document.
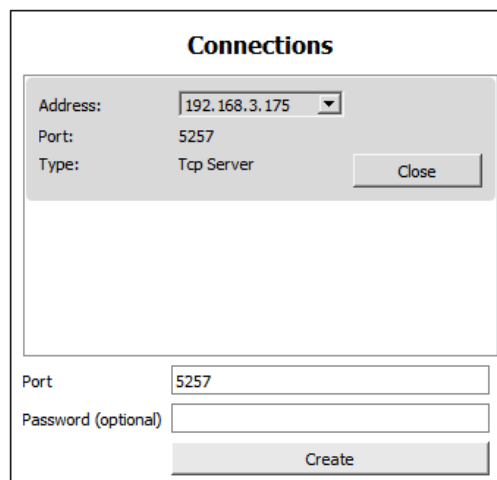

Figure 8: Connection setup

9. Live Graphs

Once the tracking has started, live data can be displayed in the 〰 menu. It shows the current values with the timestamp. The position is displayed in mm from the *x*, *y* and *z* axis. For the angular orientation the Euler angles *α*, *β* and *γ* are calculated in °.

You can use the drop-down menu at the top of the page to select a marker. Note that this option needs a lot of calculation time and may affect the performance of the tracking. If there are gaps in the graphs, they are from frame drops. You can prevent them by switching off the **Tracking in Color** display properties. You can also set a smaller **Region of interest** to reduce the required processing power.
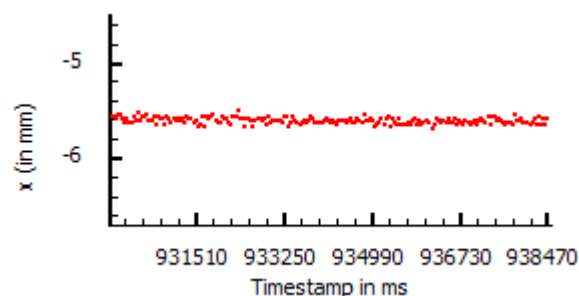

Figure 9: Example of the x axis live graph

10. Recording & Saving

If you want to save the tracking results or record a video, you can do so in the menu ⊢. This menu also offers the possibility to load and replay videos.

10.1. Tracking file

The tracking data are saved in the *.trk* file. The first line of the *.trk* file contains the time and date when the saving of data started. The second line is a head line which explains the following data. Below this line there is a calculation result in each line, which can be assigned uniquely by the marker identification number (id) and the timestamp. If several markers are tracked at the same time, the markers are saved with rising id per timestamp.

There are two options to save the raw data of a result. The first one is in quaternion combined with the translation vector. For quaternions, Q_0 is named first and then Q_1, Q_2 and Q_3. The second option is in the euler format, where you get the transformation as *x*, *y* and *z* and then the rotation as *α*, *β* and *γ*.

The quaternions correspond to the definition $q = 1 * Q_0 + i * Q_1 + j * Q_2 + k * Q_3$

Where set of quaternions $H = \{q = 1 * Q_0 + i * Q_1 + j * Q_2 + k * Q_3 \mid qt \in R, ij = k, jk = i, ki = j, ijk = -1\}$

2020-02-14T14:32:22.159+0100 equals timestamp: 1070929

id	timestamp	translation	rotation - quaternion
808	1070929	24.759 10.1551 184.278	0.814939 -0.240853 -0.101638 0.517237
36873	1070929	-18.0297 17.0613 163.966	0.970542 -0.180585 -0.156299 -0.0317462
808	1070951	24.7832 10.1544 184.259	0.814224 -0.241185 -0.103654 0.517808
36873	1070951	-17.9958 17.0256 163.872	0.970691 -0.17781 -0.15873 -0.0307752
808	1070973	24.7545 10.1763 184.319	0.814181 -0.242899 -0.102661 0.517272
36873	1070973	-18.0477 17.0953 163.997	0.97071 -0.181097 -0.154415 -0.0328864

Table 1: Tracking file example with quaternion format and two markers

2020-02-14T14:29:36.615+0100 equals timestamp: 4990769

id	timestamp	translation	rotation - euler angle
36873	4990769	-1.03415 -22.8487 209.936	-4.38762 3.3023 0.292802
36873	4990791	-1.04104 -22.8492 209.953	-4.24589 3.05195 0.274979
36873	4990813	-1.03713 -22.8355 209.917	-4.80845 3.36424 0.29699
36873	4990835	-1.03508 -22.8376 209.899	-4.82579 3.21806 0.288675

Table 2: Tracking file example with vector format and one marker
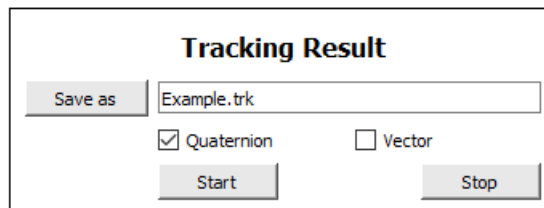

Figure 10: Settings of Saving (example)

10.2. Video file

If you want to save a video file you can do this by selecting a file path and name with the *Save as* button and then press the *Start* button. A successful recording is indicated in the camera image in the upper left corner with a red dot. You can stop the recording with the button *Stop*. The file is saved in *.mp4* format and includes the raw images without any tracking results.

If you want to play a video, you can open the desired file by clicking on the *Open* button in the file path. Tracking in this video is also possible. The *.mp4* format can lead to losses in the image quality, which affect the result. By using the drop down menu *Speed* you can vary the playing speed between *25%*, *50%*, *100%* and *200%*. To control the video you can use the three buttons *Play*, *Pause* and *Stop*. The respective state of the video is shown in the upper left image with the corresponding symbol.

11. Trouble shooting

Marker is not detected despite calibration?

First check the *Threshold* in the menu ⚙. You can display it with *Show*. Make sure that the marker is clearly visible in the image. If this is the case, check if the calibration file matches the current setup. If this does not solve the problem you can increase the parameter *Marker precision*. If the marker is now detected, look at the coordinate system. If it is not in the origin of the marker but somewhere in space, the setup has been changed and the calibration has to be repeated.

The coordinate system is displayed incorrectly?

Check the calibration and repeat it, if possible. If the coordinate system is still displayed incorrectly, check the *Marker.yml* file and measure the blobs again. You will find a detailed description in the document "Markers - Description".

Data gaps in the live graphs?

These data gaps are due to frame drops. You can try to prevent this by switching off the *Color* option in the ⚙ menu, reducing the *Region of interest* and increasing skipped images.

12. Libraries and licences

The MRC Motion Tracking software uses the following libraries with specific licences:

    a) OpenCV version 3.4 from http://opencv.org with 3-clause BSD license
    b) QT 5.9.2 with LGPLv3 licence

The information on the licences is included as data files on the installation medium and displayed in the tab **i** .

Contact

MRC Systems GmbH
Hans-Bunte-Str. 10
D-69123 Heidelberg
Germany
Tel.: 06221/13803-00
Email: info@mrc-systems.de

Subject to change.