Vicon CaraPost Reference Guide

Contents

Vicon CaraPost Reference Guide — Revision 1.1 January 2015
For use with Vicon CaraPost Version 1.1

Vicon Motion Systems is an OMG plc company.
Email: *support@vicon.com* Web: *http://www.vicon.com*

About the Vicon CaraPost Reference Guide

This guide provides information about Vicon CaraPost including customization and other advanced functions, algorithms, detailed descriptions of the user interface, and a troubleshooting guide.

For a basic introduction to using Vicon CaraPost, see the *Vicon CaraPost User Guide*.

About Vicon Cara documentation

The documentation (online help and PDFs) included in the current Vicon Cara release consists of the following:

Document	Description
Vicon Cara User Guide	Provides system description, installation instructions (in PDF only) and licensing instructions, and a basic step-by-step workflow. PDF and online help installed with CaraLive.
Vicon Cara Reference	Contains details of advanced functionality, information about the configuration server, descriptions of settings and options, and supported file formats. Hardware details and a glossary are also included. PDF and online help installed with CaraLive.
Vicon CaraPost User Guide	Provides system description, installation instructions (in PDF only) and licensing instructions, and a basic step-by-step workflow. PDF and online help installed with CaraPost.
Vicon CaraPost Reference	Contains details of advanced functionality, descriptions of settings and options, and supported file formats. A glossary is also included. PDF and online help installed with CaraPost.

The PDFs (for Adobe Reader version 8.0 or later) are installed as part of your Vicon CaraPost software installation.

The documents available depend upon your Vicon software license options.

You can also obtain these PDFs, and other Vicon documents, from the Downloads page of the *Vicon Support website*.

Advanced functions

This chapter covers the following topics:

Scripting with Python

Vicon CaraPost functionality is available in the Python scripting language via the PyCara Python package.

The package contains sub-packages that enable you to create custom export formats by reading the data contained in *.cara* and *.pico* files; and to create custom batch processes using some of the core CaraPost blob detection and tracking functionality.

For information about Python, see:

- *https://docs.python.org/2/*
- *https://docs.python.org/2/tutorial*

For more information about PyCara, see:

- *Install PyCara*
- *Get started with PyCara*
- *Example Python scripts*

Install PyCara

Currently PyCara is delivered as a Windows executable installer built for Python 2.7 64-bit. It is available on the *Vicon Support website*, in the same location as the CaraPost installer.

Install it on a system that has either Vicon CaraLive or Vicon CaraPost installed, as this is necessary for licensing.

Get started with PyCara

The PyCara package contains the following sub-packages:

- PyCaraExport Enables you to read .*cara* files, and access the data contained within in order to export the data into custom file formats and enable reading into third-party packages.
- PyCaraProcess Enables you to create .*cara* files from .*pico* files and perform Vicon CaraPost processing options (eg, blob detection and point tracking).
- PyDebugServices Enables you to show or hide log information.
- PyPico Enables you to read .*pico* file metadata and extract individual frames.

After the package has been installed, you can use Python's online help system to obtain detailed information on the functionality in each package.

For example, the following documentation shows that there are two version of the `detect blobs` member function, one that takes no arguments, and one that takes a list of sensors, a start frame index and an end frame index:

```
>>> from PyCara.PyCaraProcess import CaraProcess
>>> help(CaraProcess.detect_blobs)
Help on method detect_blobs:

detect_blobs(...) unbound
PyCara.PyCaraProcess.PyCaraProcess.CaraProcess method
    detect_blobs( (CaraProcess)self) -> bool :
        Detect blobs.

    detect_blobs( (CaraProcess)self, (list)list of sensors,
(int)start frame index, (int)end frame index) -> bool :
        Detect blobs.
```

Example Python scripts

To enable you to perform basic processing operations, the following example scripts are provided in a zip file that is available on the *Vicon Support website*, in the same location as the CaraPost installer.

- *process_detect_blobs.py*
- *process_track.py*
- *process_batch_detect_blobs.py*
- *cara_fbx_export.py*
- *process_setup_ROM.py*
- *process_properties.py*
- *export_3d_simple.py*
- *export_brim.py*
- *export_general.py*
- *export_timecode.py*

You can run the example scripts from a command prompt. If the script takes arguments, to get a description of the arguments:

- On the command line, run the script with the `--help` flag or `-h` flag.

process_detect_blobs.py

This script takes a *.pico* file as input, loads all the *.pico* files for that take, and then performs blob detection on them, and writes the output to *<take_name>*`_b.cara`.

The following example shows how to call the script:

```
PS1> .\process_detect_blobs.py Example_0.pico
```

process_track.py

This script either takes a *.cara* file as input and tracks from the first frame onwards, or takes a *.pico* file as input, imports all the *.pico* files for that take, initializes using the supplied ROM and then tracks.

The following example shows how to call the script with a *.cara* file:

```
PS1> .\process_track.py Example.cara
```

The following example shows how to call the script with a *.pico* file and a ROM:

```
PS1> .\process_track.py Example_0.pico –rom ROM.cara
```

process_batch_detect_blobs.py

This script is an example of how to use PyCara and Python to construct batch processing scripts. The script is given a directory or filename (that may include wildcards). The script then attempts to find all Cara takes that match the input and, if there is not already a *.cara* file with blob detections, it creates one.

The following example shows how to call the script with a directory:

```
PS1> .\process_batch_detect_blobs.py C:/Data/Jobs
```

The following example shows how to call the script using wildcards;

```
PS1> .\process_batch_detect_blobs.py C:/Data/ROM*.pico
```

cara_fbx_export.py

This script takes a solved *.cara* file as input, and then generates an FBX output file. It produces the same output as the built-in CaraPost FBX export but can be customized to meet your requirements, for example, to add face points to different subnodes.

The following example shows how to call the script:

```
PS1> .\cara_fbx_export.py Example.cara –o c:\tmp\Example.fbx
```

To run the **cara_fbx_export.py** script, you must download and install the Autodesk Python SDK.

By default, after installation the Filmbox Python files are located in the following folder:

C:\Program Files\Autodesk\FBX\FBX Python SDK\2015.1\lib\Python27_x64

Add this to your `PYTHONPATH` environment variable.

process_setup_ROM.py

This is a simple script demonstrating how to generate a *.cara* file for a Cara ROM take, ready to be processed in CaraPost.

process_properties.py

This is a simple script showing how to obtain a list of all properties for each property group. This enables you to see all available blob detection and tracking parameters.

export_3d_simple.py

This is a simple script showing how to output 3D face data from a *.cara* file to the console window.

export_brim.py

This is a simple script showing how to output 3D data for the calibration brim from a *.cara* file to the console window.

export_general.py

This is a simple script showing how to query a *.cara* file for point data, and plot 2D track data.

Note

Requires PyLab (*http://wiki.scipy.org/PyLab*) to be installed for data plotting.

export_timecode.py

This is a simple script showing how to convert between the various time representations used by CaraPost, namely zero-based frame indices, burnt-in frame numbers, and timecode.

3D initialization for alternative marker sets

The *Vicon CaraPost User Guide* describes how to refine camera positions and obtain marker reconstructions using a fairly dense marker set and selecting a minimum of four or five seed points. Note that adding more seed points generally improves the results of the initialization process.

The following topics describe how to apply selection criteria and select appropriate seed points for two alternative marker sets. 3D initialization settings are suggested for two marker sets representing a very dense marker-set consisting of 156 markers and a sparse marker set consisting of 35 markers.

The aim of 3D Initialization is to correctly initialize the positions of all the markers on the face that you want to track. You must also ensure that each marker point is attached to a detection in each camera view in which it is visible.

To achieve good tracking results for the take, it is very important to perform the Initialization step as accurately and completely as possible.

- *3D initialization of a dense 156-marker set*
- *3D initialization of a sparse 35-marker set*
- *Handle mismatched or unconnected points*

3D initialization of a dense 156-marker set

This section covers the following topics:

Criteria for selecting seed points for a 156-marker set

Seed points must meet the following criteria:

- Reasonably spaced out across the face
- Preferably visible in all four camera views, and at an absolute minimum, visible in the two views of a camera caddy (ie, either channels 0 and 1, or channels 2 and 3).

Set the Initialize 3D Parameters properties for a 156-marker set

In the **Initialize 3D Parameters** section of the **Properties** pane, specify:

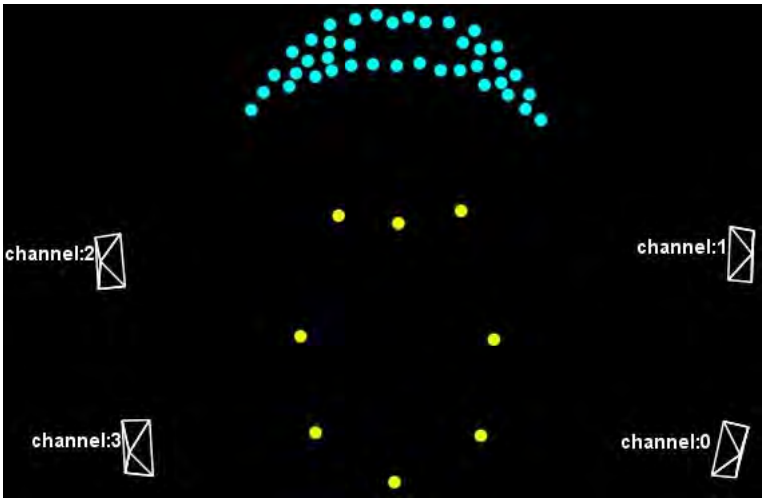
- Max. 3D Distance to Merge (mm): 5
- Max. Allowed Depth (mm): 16
- Max. Allowed Symmetric Epipolar Distance (pixels): 5

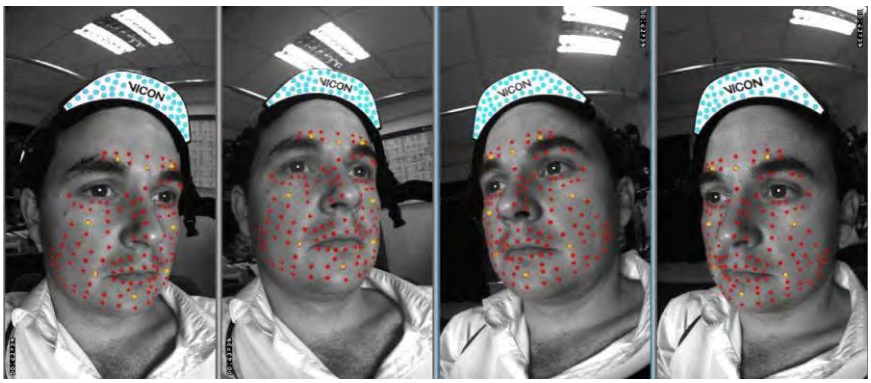
Note

For a dense marker set, the **Max. Allowed Depth** parameter is lower than the default value of 35mm, because the markers are closer together, and the higher default value may increase the chance of marker mismatches.

Select the seed points for a 156-marker set

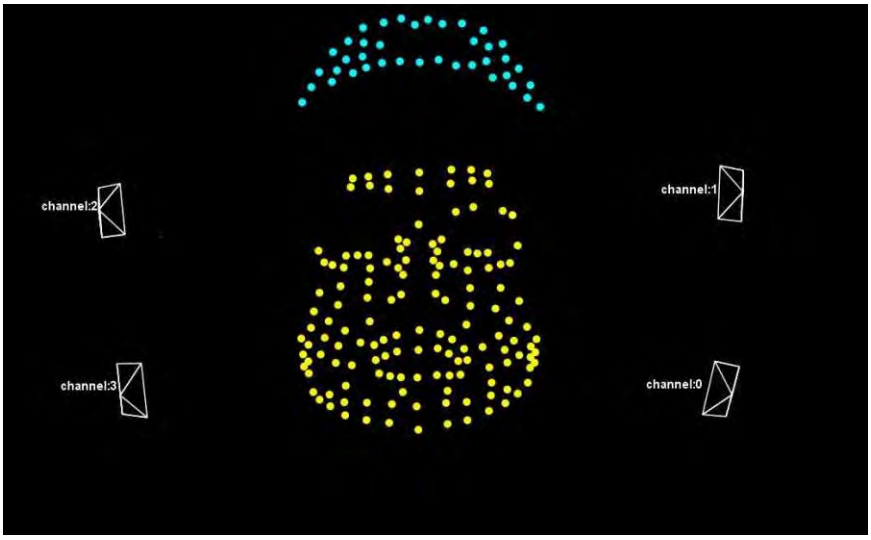
The following illustration shows eight example seed points selected on the actor's face in 3D and camera views for a 156-marker set:





Assess the results of 3D initialization for a 156-marker set

After you have merged the detections for the selected seed points and run **Bundle Current Frame [Unconstrained]** and **Initialize 3D**, you can view the results in both the 3D view and in each (2D) camera view. The yellow projections in the 2D views represent points that have been matched in that view, while the orange points do not have an associated detection in that channel:



To assess the results of the 3D initialization:

1. View the results of the 3D initialized facial points in a 3D view, by rotating (SHIFT+click+ drag), zooming (SHIFT+right-click+drag) and/or panning (SHIFT+click both mouse buttons+ drag) the 3D viewpoint.

   This enables you to identify any gross errors or mismatches in the 3D data, which need to be corrected.

2. Check the results for each facial point in each of the four 2D camera views. To do this, select each facial point in turn, and in each camera view:

   - For marker points which are colored orange (i.e. not attached to a detection in the current 2D view), check whether the point needs to be attached to a red, unconnected detection. If so, you need to correct this.
   - For marker points which are colored yellow (i.e. they are attached to a detection in the current 2D view), check that the connection is to a well-located, correct detection.

     Also check for cases where the initialization has matched a marker point to a detection which is visible at a very oblique angle, close to the occluding contour (for example a point on the side of the face which is more directly visible in the other camera caddy). In this case, it may be sensible to detach the detection from the 3D point.
   - Check for marker points that have not been initialized at all, ie, the detections are present (as red unconnected detections) in at least two 2D camera views, but no 3D point has been initialized.

For tips on how to deal with the unconnected points, see *Handle mismatched or unconnected points* on page 22.

3D initialization of a sparse 35-marker set

This section covers the following topics:

Criteria for selecting seed points for a 35-marker set

Seed points must meet the following criteria:

- Reasonably spaced out across the face
- Preferably visible in all four camera views, and at an absolute minimum, visible in the two views of a camera caddy (ie, either channels 0 and 1, or channels 2 and 3).

Set the Initialize 3D Parameters properties for a 35-marker set

In the **Initialize 3D Parameters** section of the **Properties** pane, specify the following values

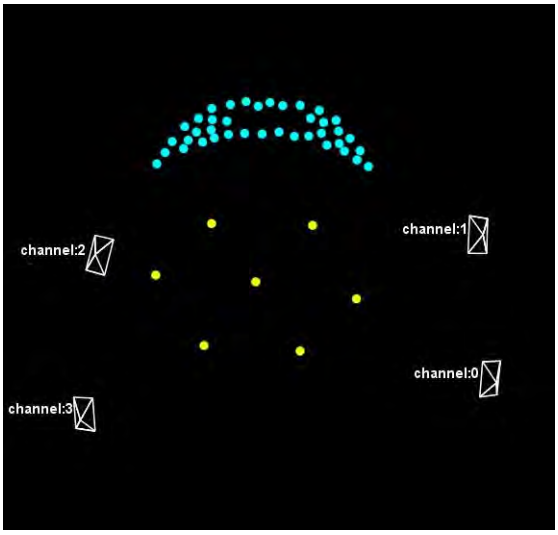
- Max. 3D Distance to Merge (mm): 5
- Max. Allowed Depth (mm): 60
- Max. Allowed Symmetric Epipolar Distance (pixels): 5

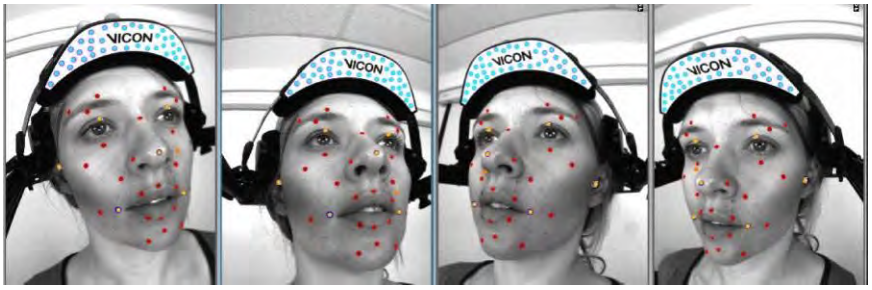
Note

For a dense marker set, the **Max. Allowed Depth** parameter is higher than the default value of 35mm, because the markers are more widely spaced, and the parameter value needs to be higher so that the initialization algorithm searches sufficiently far away from each seed point for additional matches.

Select the seed points for a 35-marker set

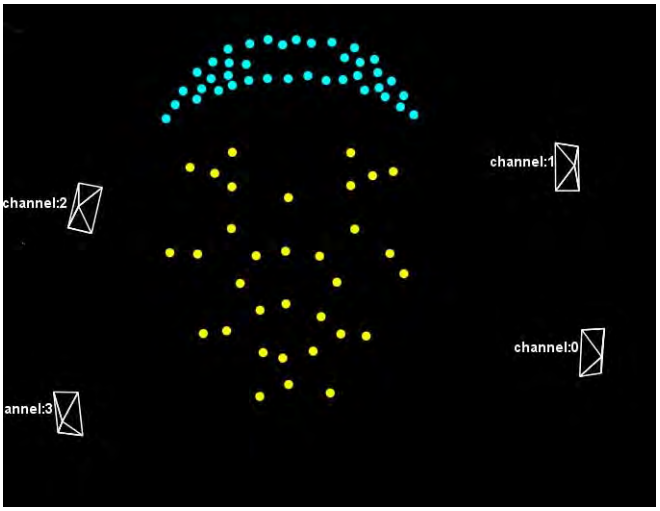
The following illustration shows seven seed points selected on the actor's face in 3D and camera views for a 35-marker set:

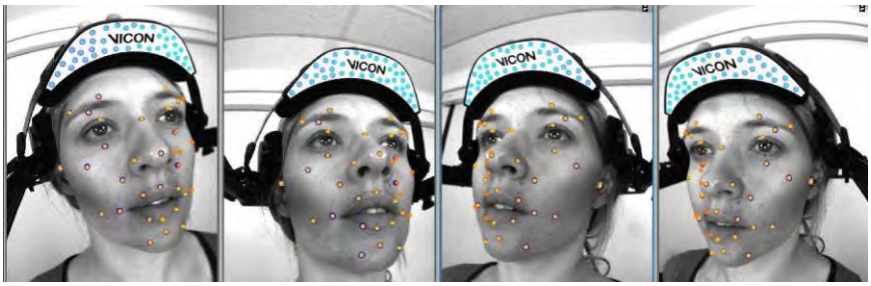




Assess the results of 3D initialization for a 35-marker set

After you have merged the detections for the selected seed points and run **Bundle Current Frame [Unconstrained]** and **Initialize 3D**, you can view the results in both the 3D view and in each (2D) camera view. The yellow projections in the 2D views represent points that have been matched in that view, while the orange points do not have an associated detection in that channel:





To assess the results of the 3D initialization:

1. View the results of the 3D initialized facial points in a 3D view, by rotating (SHIFT+click+ drag), zooming (SHIFT+right-click+drag) and/or panning (SHIFT+click both mouse buttons+ drag) the 3D viewpoint.

   This enables you to identify any gross errors or mismatches in the 3D data, which need to be corrected.

2. Check the results for each facial point in each of the four 2D camera views. To do this, select each facial point in turn, and in each camera view:

   - For marker points which are colored orange (i.e. not attached to a detection in the current 2D view), check whether the point needs to be attached to a red, unconnected detection. If so, you need to correct this.
   - For marker points which are colored yellow (i.e. they are attached to a detection in the current 2D view), check that the connection is to a well-located, correct detection.

     Also check for cases where the initialization has matched a marker point to a detection which is visible at a very oblique angle, close to the occluding contour (for example a point on the side of the face which is more directly visible in the other camera caddy). In this case, it may be sensible to detach the detection from the 3D point.
   - Check for marker points that have not been initialized at all, ie, the detections are present (as red unconnected detections) in at least two 2D camera views, but no 3D point has been initialized.

For tips on how to deal with the unconnected points, see *Handle mismatched or unconnected points* on page 22.

Handle mismatched or unconnected points

Following automatic initialization, the next step is to correct the set of initialized 3D points so that they are as accurate and complete as possible. The aim is to create a complete set of 3D points matched in as many camera views as possible as a good starting point for the tracking algorithm.

You need to consider each required point in your marker set and examine it, both in the four 2D camera views, and also in the 3D view. You may need to hand-correct the following classes of error at this stage:

- *Reconstructed point contains one or more mismatched detections across the four camera views.*
- *A required marker is entirely missing from the set of 3D points.*
- *A required marker is correctly reconstructed in 3D, but additional unconnected detections in one or more camera views would provide useful tracking data.*

Reconstructed point contains one or more mismatched detections across the four camera views.

To correct mismatched detections in reconstructed points:

1. In either a camera view or the 3D view, select the 3D point with the mismatched detection(s).
2. Do one of the following:
   - Either delete the 3D point entirely by pressing Ctrl+D; or
   - To detach unwanted detections from the point, in the **Selection** pane, select the channel(s) in which the mismatch(es) occur and then click **Edit** > **Clear Selected [Current Frame]**.
3. Before tracking the take, run **Bundle Current Frame [Unconstrained]**.

A required marker is entirely missing from the set of 3D points.

To create a new 3D point:

1. Selecting matched detections in two or more camera views.
2. Press M to merge the detections to create a new 3D point.
3. Before tracking the take, run **Bundle Current Frame [Unconstrained]**.

A required marker is correctly reconstructed in 3D, but additional unconnected detections in one or more camera views would provide useful tracking data.

To add unconnected detections to a point:

1. In either a camera view or the 3D view, select the existing 3D point (or one of its existing detections).
2. To add in the additional detections to the point, CTRL+click the missing detection(s) in other camera view(s), and press M.
3. Before tracking the take, run **Bundle Current Frame [Unconstrained]**.

Modify the calibration brim

To modify the calibration brim that is supplied with your Vicon Cara system, use the brim template that is installed to the following default location:

C:\Program Files\Vicon\CaraPost\CalibrationBrimFiles

Black and white versions of the template are supplied as:

- An SVG (Scalable Vector Graphics) file, for editing and printing:
  - *Brim_Calibration_Label-BLACK-CUSTOM_TEMPLATE.svg*
  - *Brim_Calibration_Label-WHITE-CUSTOM_TEMPLATE.svg*
- A PDF file, for viewing only:
  - *Brim_Calibration_Label-BLACK-CUSTOM_TEMPLATE.pdf*
  - *Brim_Calibration_Label-WHITE-CUSTOM_TEMPLATE.pdf*

Both the supplied SVG file and the corresponding PDF include guidelines on how to edit the SVG file to produce a custom calibration brim.

To open the file and edit the required layers, use the Inkscape SVG Editor software, a free download, available from:

http://inkscape.org/

In a normal Vicon CaraPost workflow (see the *Vicon CaraPost User Guide*), after you have imported the capture files and performed blob detection, you use the **Initialize Default Calibration Brim** command on the **Processing** menu to enable Vicon CaraPost to calculate the positions of the cameras. However, if you are using a modified calibration brim, instead of **Initialize Default Calibration Brim**, go to the **File** menu and click **Import Calibration Brim [SVG]**.

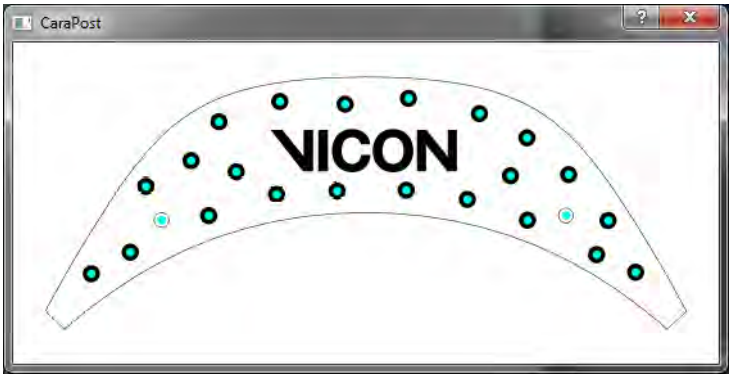
Important

Note that although this functionality in CaraPost enables you to use a customized (planar) calibration brim, end-user support for this functionality is not provided. You are responsible both for creating a suitable calibration brim geometry, and also for printing and producing the final physical brim.

To view the customized calibration brim:

1. On the **View** menu, click **Show Calibration Brim SVG**.

The following illustration shows the V2.0 calibration brim with black blobs on a white background, as displayed for viewing.



Note

A known issue in Vicon CaraPost causes certain editing operations performed on the SVG file in the Inkscape SVG editor to result in an SVG file that displays the image of the brim upside-down in CaraPost. This issue affects only the display of the brim and does not affect its functionality.

Target tracking

The Target Tracker is a manual tracking tool which enables you add and track additional points where no detections exist, or fill in gaps for existing tracked points where the automatic tracker has failed.

In normal use of Vicon CaraPost, you should not need to use the Target Tracker very often, but it can occasionally be useful for tracking marker points which cannot be tracked any other way. A good example is for tracking marker points which go in and out of occlusion during a take, for example, eyelid markers.

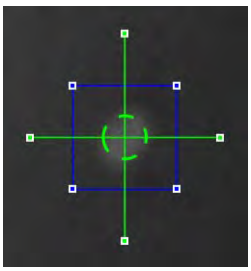
The following target tracking workflows are covered:

- *Create and track a new target-tracked point*
- *Fill gaps or re-track existing points with Target Tracker*
- *Example of tracking a partially occluded marker*

For a description of how the target tracker can be used to create missing detections as part of the main automatic tracking workflow within CaraPost, see *Optimize a take* in the *Vicon CaraPost User Guide.*

About target tracks

When you add a target track, if you zoom in (keeping the target track selected), you can see that it is displayed as a blue box containing a green circle around a cross.



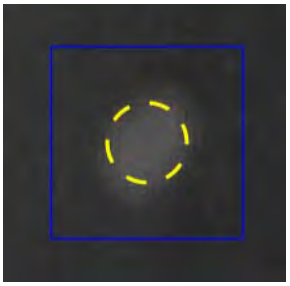
The blue box contains the pattern that you are tracking (the marker). Ensure it comfortably contains the marker. The green cross represents the comparison region, which is the area in adjacent frames that will be searched to find the pattern that is being matched (ie the marker). The green circle represents the middle of the pattern.

To adjust the comparison region, drag its white handles. Make the comparison region slightly larger than the pattern to be matched (usually a circular marker). If you make the comparison region too large, the accuracy of the pattern matching is reduced. If you make the region too small, the outline of the circle is hard to distinguish against the background.

Tip

Increase the size of the comparison region (the green cross) for fast-moving shots and reduce it for slow-moving ones.

When a target track is not selected, it is rendered as a blue box representing the pattern area, and a broken yellow circle representing the center of the pattern area.



A target track that is not on a keyframe is rendered as a broken blue circle representing the center of the pattern area.

Create and track a new target-tracked point

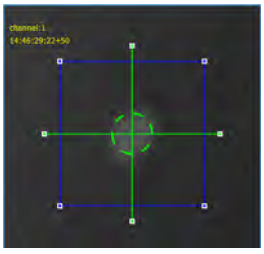
You can use the Target Tracker to create and track new marker points in a take. You might want to use this functionality in cases where the blob detector has failed to create detection(s) for a marker, or if an automatically created detection is incorrectly located, and you want to replace it with a correctly located manual detection.

To add detections manually:

1. Scrub to the first frame to which you want to add a detection (that is, a frame where you can see a marker that is lacking a corresponding detection).
2. In the camera view where the detection is missing, zoom in (SHIFT+right-click and drag) to the relevant area of the image.
3. On the **Edit** menu, click **Add Target Track** (or press SHIFT+A).

   The words **Add Target Track** are displayed beneath the Channel labels in the camera views.
4. In the camera view where the detection is missing, click where you want to add the new detection.

   A new target track keyframe detection, which is attached to a corresponding new point (which appears in the Continuity chart), is created in the camera view.



5. Adjust the keyframe, using the following tips:

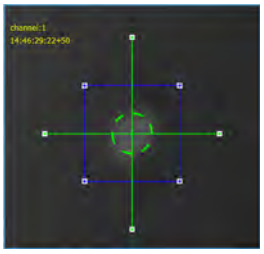
   - The blue box represents the area of the pattern (ie the marker) you want to track. Adjust this using the handles and aim for the box width to be approximately two to three times the diameter of the marker.
   - The green cross represents the search area, that is, the area in adjacent frames that will be searched to find the marker. The blob template contained within the blue box is compared to all locations within the limits of this area. A smaller search area results in faster tracking and

reduces the chances of the track swapping onto a different marker. However, making the search area smaller also increases the likelihood that the marker may be lost during tracking when there is rapid movement. Again, you can adjust these limits using the handles.

For more information, see *About target tracks* on page 26.

- You can use the nudge controls (CTRL+SHIFT+arrow) to center the selected target track on the blob.
- If the keyframe detection becomes deselected, to reselect it, click in its center.

After you have finished adjusting the keyframe detection, it should look similar to the following example:



6. You can now add a target track keyframe for your new point in a second camera view for the same marker. To do this, with your new point selected, on the **Edit** menu, click **Edit Target Track** (or press SHIFT+E).

   The text **Edit Target Track** is displayed below the channel labels.

7. In the second camera view, click on the marker.

   A green cross with a circle around appears. As you now have two corresponding detections in two camera views, CaraPost can now reconstruct the 3D position of this marker. Adjust this keyframe as you did in step 5 for the first keyframe.

8. You can now track the new point to the end of the take. To do this, ensure you have selected only the required point and in the Tracker window, click the Track forward (>) button.

   Tip

   If you can't see the **Tracker** window, right-click on the menu bar of one of the visible windows and click **Tracker**.

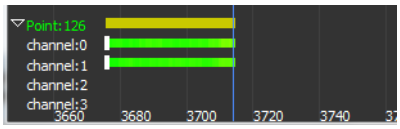
A progress bar is displayed below the play controls and the take plays through.

Tip

While the take plays, the Track Forward button changes to a Pause button. If you click the Pause button, Vicon CaraPost stops tracking. You may need to stop tracking if the target track starts to drift off the marker when you are working with markers that are difficult to track.

9. With the new point still selected, open a **Selected Point** pane by right-clicking on a menu bar and clicking on **Selected Point**.

   In the **Selected Point** pane, three bars should be displayed, similar to those in the following illustration.

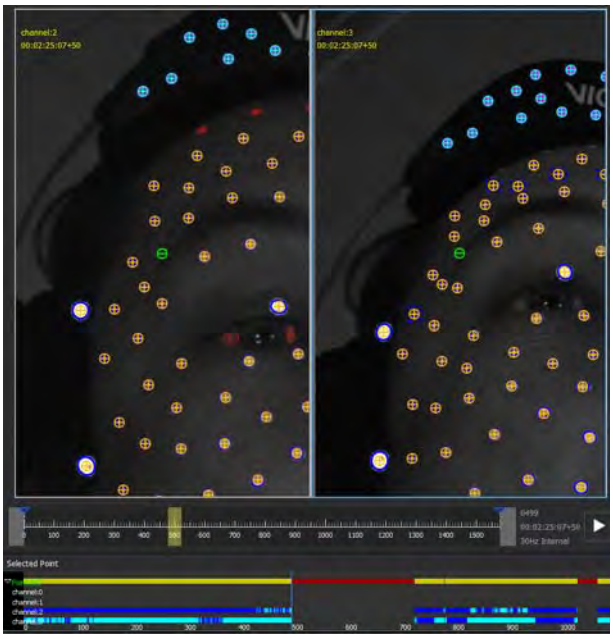


   The yellow top bar indicates that there is a 3D reconstruction for this point, present in two camera views, and the colors of the two bars beneath it (between green and red) indicate the health of the 2D target tracks. Green indicates that the target tracking results in that camera is good; red indicates that the target tracking results are poor.

Fill gaps or re-track existing points with Target Tracker

The main use of the Target Tracker is to fill in gaps or to re-track over poor existing results (or interpolated results) for a point.

The following workflow demonstrates re-tracking over an existing interpolated point result for a point where the automatic tracker has failed. The illustration below shows an eyebrow marker point where the automatic tracker has generated a purely interpolated result (shown as a red bar in the **Selected Point** pane).



To re-track over the interpolated data, you need to add keyframes and track the blob using the Target Tracker.

To re-track over the interpolated data:

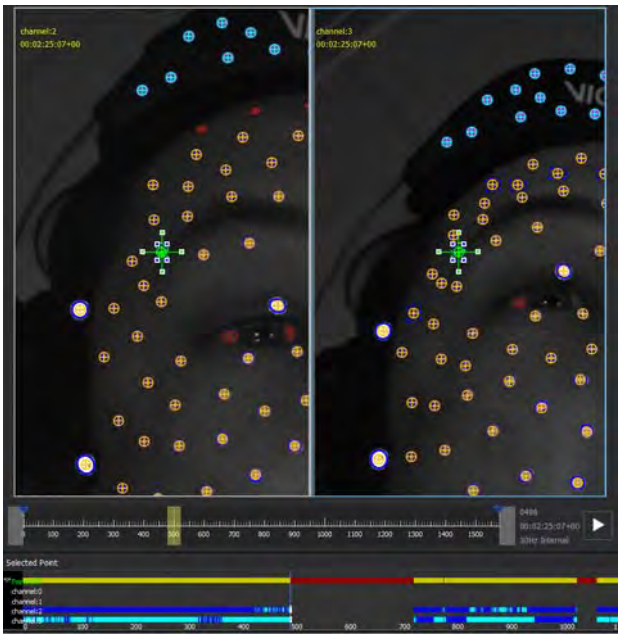
1. In a 2D or 3D view, select the point that contains the interpolated results (in this case, point 567) by clicking on it.
2. Go to (Ctrl+G) a frame just before the start of the gap, which in this case is frame 498. Check that the tracking result for the point looks OK in this frame.

3. To add a keyframe, in the **Tracker** pane, click **Add Keyframe To Current Frame**.

   Tip

   If you can't see the **Tracker** pane, right-click on any menu bar and then click **Tracker**.

   A white tracking box appears around the selected marker and a white bar is displayed in the **Selected Point** pane and the **Continuity** chart:

   In this case, the marker point is attached to detections in channels 2 and 3 in the frame before the gap, so keyframe detections are added in channels 2 and 3 only.

   Note

   If required, you can add additional detections in other channel(s) by clicking **Edit** > **Edit Target Track** (for more information, see *Create and track a new target-tracked point* on page 28.

4. Go to the frame after the gap (in this case, frame 728) and add another keyframe.

5. In the **Tracker** pane, ensure that:

   - **Break Threshold** is set at an appropriate level. 0.8 is usually a reasonable value to use.

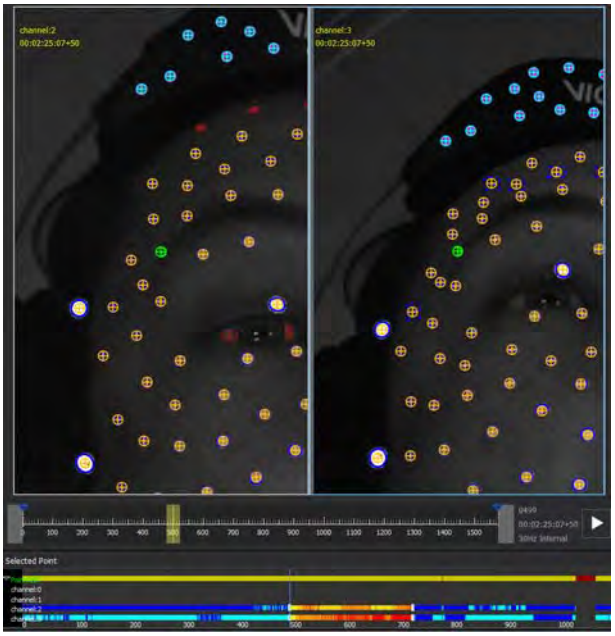
     This parameter defines how well the image patch being tracked has to match the corresponding patch in the adjacent frame (with 1 being a perfect match) for tracking to continue. If the patch falls below this score then tracking stops.

   - **Stop On Next Keyframe** is selected.

   - **Overwrite Existing Tracks** is selected. Note that if you had a pure gap, rather than an interpolated result, you could leave this option cleared.

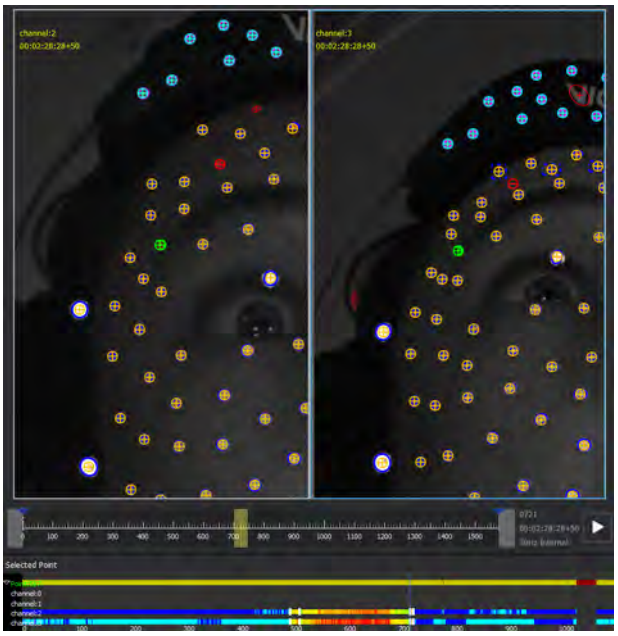
6. Ensure you are still at the frame after the gap (frame 728) and click the Track backward button (<).

   The Target Tracker tracks backwards across the gap, displaying a colored bar between the two keyframes, which indicates the quality of the Target Tracking result.



7. Check the quality of the tracking results, both in the 2D views and in 3D. If you are satisfied with the quality of the result, you can stop at this step.

8. However, if the Target Tracker has not produced good results across the whole gap, you can usually improve the results by adding keyframes at a few additional locations using the **Add Keyframe To Current Frame** button. For each added keyframe, ensure that the keyframe detection in each camera view is located at exactly the right position before re-tracking over the poor quality results using the Target Tracker. The following illustration shows how two additional keyframes have been added to improve the tracking results in this case:

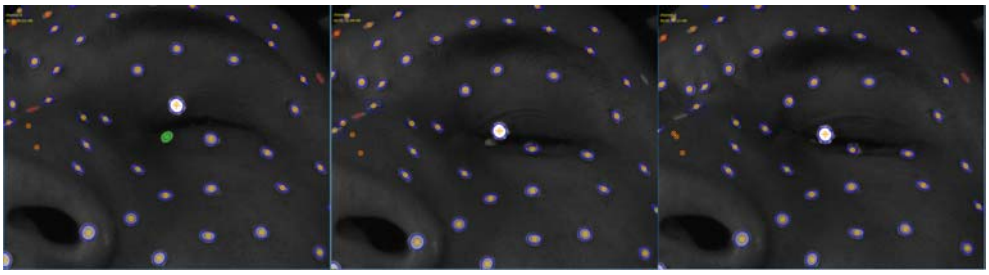


Example of tracking a partially occluded marker

One of the most common uses of the Target Tracker is to track markers which have become either partially or completely occluded during the take. The most common examples of where this happens are points on the eye or eyelid, and sometimes points around the lips for cases of extreme lip curl.

The following workflow demonstrates re-tracking over the top of interpolated results for a point in the corner of the eye that becomes occluded, where the automatic tracker has failed. This represents one of the most difficult tracking problems you may encounter using Vicon CaraPost.

The following illustration shows three examples of an eye corner point (selected in green at the frame before the gap starts), showing how the corresponding detection becomes occluded as the actor tightly closes his eyes:



By the third example frame shown, the eye corner detection is almost completely occluded.

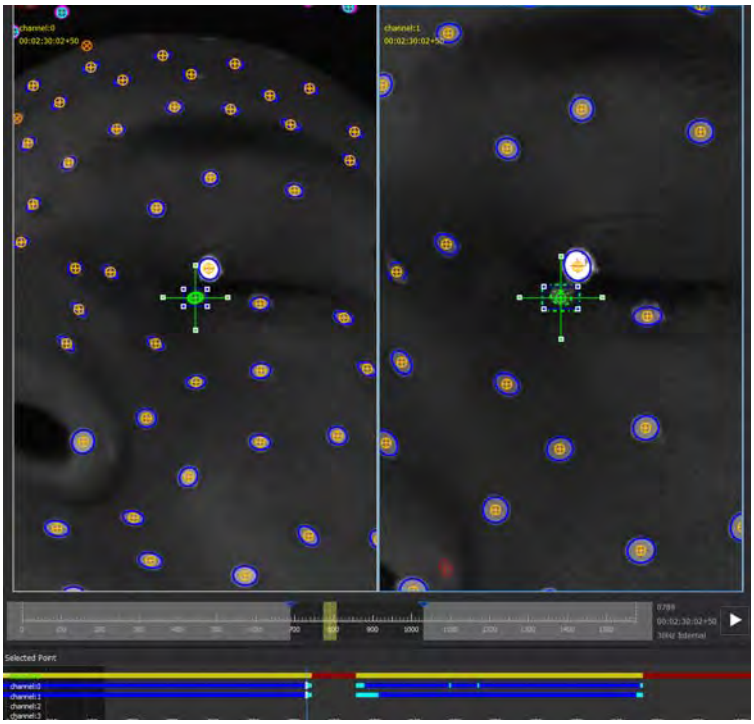
It may not be possible to produce a perfect result for this marker point, because it becomes almost completely occluded in some frames, but the Target Tracker enables us to produce reasonable results, by manually tracking the small parts of the marker that are visible, or the closest point to it if the marker becomes fully occluded. The workflow is similar to the standard use of the Target Tracker (see *Fill gaps or re-track existing points with Target Tracker* on page 31), but at points where the marker is badly occluded, you may need to manually add a large number of keyframes to achieve good tracking results.

To target track a partially occluded marker:

1. In a 2D or 3D view, select the point that contains the interpolated results (in this case, point 578) by clicking on it.
2. Go to (Ctrl+G) a frame just before the start of the interpolated result (or gap). In the above example, the tracking of the eye point has already started to drift off before the tracking completely fails, so you would go to the last good frame of tracking for this point, which is frame 789.

3. To add a keyframe, in the **Tracker** pane, click **Add Keyframe To Current Frame**.

   A white tracking box appears around the selected marker and a white bar is displayed in the **Selected Point** pane and the **Continuity** chart.



   In this case, the marker point is attached to detections in channels 0 and 1 in the frame before the gap, so keyframe detections are added in channels 0 and 1 only.

4. Go to the frame after the gap frame (in this case, frame 826) and add another keyframe.

5. In the **Tracker** pane, ensure that:

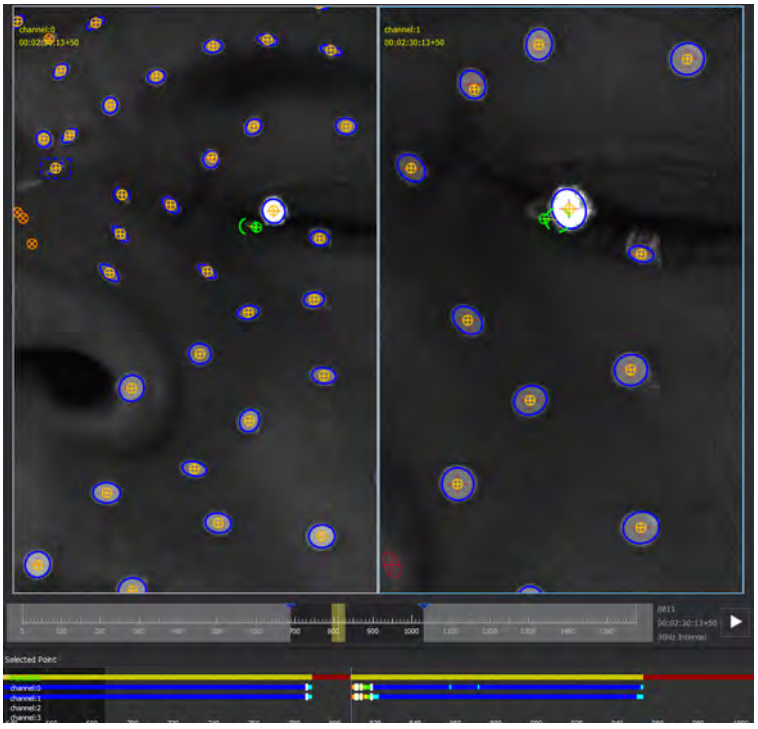
   - **Break Threshold** is set at an appropriate level. 0.8 is usually a reasonable value to start with, but may need to be slightly lower in this case, if we cannot get the Target Tracker to track.

     This parameter defines how well the image patch being tracked has to match the corresponding patch in the adjacent frame (with 1 being a perfect match) for tracking to continue. If the patch falls below this score then tracking stops. In this case, the marker appearance changes

a lot across the gap so the tracking will probably stop due to the **Break Threshold** being reached on a number of occasions.

- **Stop On Next Keyframe** is selected.
- **Overwrite Existing Tracks** is selected. Note that if you are tracking over a gap (rather than an existing interpolated result), for difficult tracking cases, you should probably select this option, as you may need to redo some of the tracking if/when the Target Tracker fails.

6. Ensure you are still at the frame after the gap (frame 826). Because this tracking task is very difficult, track backwards one frame at a time by clicking the **Track backwards one frame** button (|<), so you can immediately see if the tracking fails.
7. Keep tracking backwards frame by frame, until either the Target Tracker fails to track in both channels, or the Target Tracker makes a tracking error, as shown in the following example.

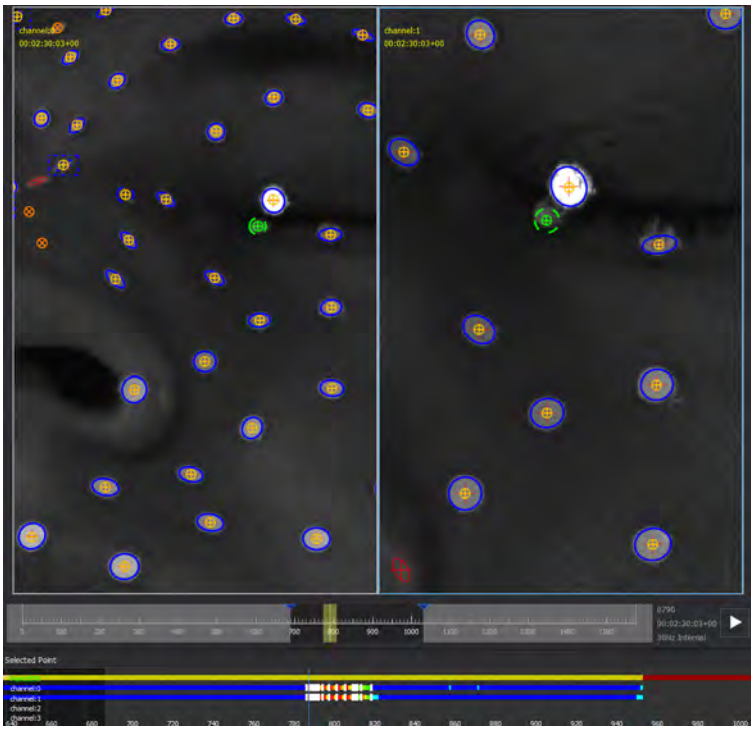


8. When you hit a failure case, navigate forward one frame, and add an additional keyframe at the last good tracking result by clicking **Add Keyframe to Current Frame**.

In most cases, adding a keyframe should give the Target Tracker the additional information it needs to carry on tracking. Rarely, you may also have to force the tracking to continue by:

- Editing the keyframes by hand in the failure frame; or
- Lowering the Break Threshold value slightly to force it to track

9. Continue Steps 7 and 8 until you have filled the gap completely, as shown in the following illustration. For a very demanding take, you may need many keyframes to fill the gap.



Tracking parameters for alternative marker sets

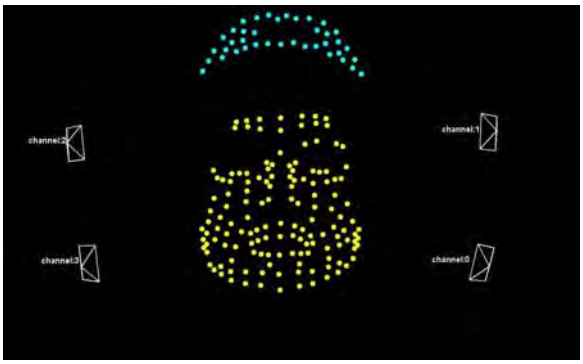
The *Vicon CaraPost User Guide* describes how to perform tracking for a typical (fairly dense) marker set, using the default set of tracking parameters. The following topics describe how to alter the tracking parameters to work better with alternative marker sets of differing marker density.

Tracking settings are suggested for the following marker sets:

- *Tracking parameter values for a dense 156-marker set*
- *Tracking parameter values for a sparse 35-marker set*

Tracking parameter values for a dense 156-marker set

The following illustration shows a dense marker set consisting of 156 facial markers.

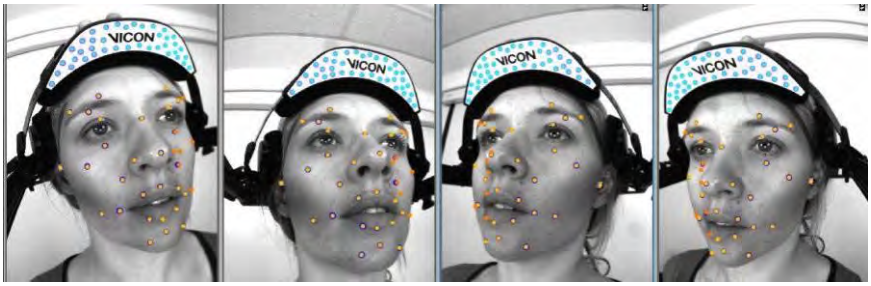


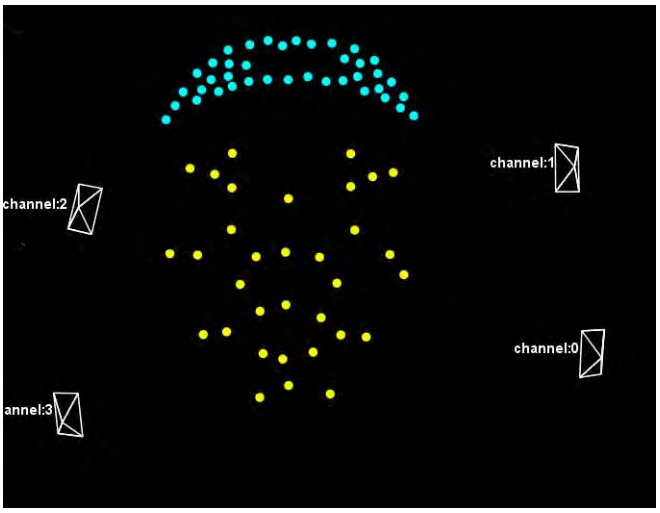
The suggested tracking parameters for this marker set are given below, and are identical to the default settings:

Tracking parameter	Value
Patch Match Threshold	0.8
Face Point Tracking Distance	35 pixels
Face Point Max Elastic Prediction Distance	15 pixels
Brim Point Tracking Distance	15 pixels
Patch Match Size	15 pixels
Remove Bad 2D Projections	True
Max Allowed 2D Projection Error	5 pixels
Remove Bad 3D Points	True
Max Allowed 3D Point Error	5mm
Allow Single Sensor Tracking	True
Single Sensor Tracking Threshold	0.5mm
Blob Size Difference Threshold	0.75

Tracking parameter values for a sparse 35-marker set

The following illustration shows a sparse marker set consisting of 35 facial markers.





The suggested tracking parameters for this marker set are given below, with the three suggested changes from the default settings highlighted in yellow:

Tracking parameter	Value
Patch Match Threshold	0.8
Face Point Tracking Distance	35 pixels
Face Point Max Elastic Prediction Distance	25 pixels
Brim Point Tracking Distance	15 pixels
Patch Match Size	15 pixels

Tracking parameter	Value
Remove Bad 2D Projections	True
Max Allowed 2D Projection Error	5 pixels
Remove Bad 3D Points	True
Max Allowed 3D Point Error	10mm
Allow Single Sensor Tracking	False
Single Sensor Tracking Threshold	0.5mm
Blob Size Difference Threshold	0.75

For this more sparse facial marker set, it is advisable to relax a couple of tracking parameter values to reflect the fact that the facial markers are more widely spaced, and thus the tracking elastic prediction algorithm, which makes use of the relationships between neighboring marker points, will be less accurate. You should therefore relax both the **Face Point Max Elastic Prediction Distance** and **Max Allowed 3D Point Error** parameters, and also (probably) turn off **Single Sensor Tracking**, which also relies upon the accuracy of the elastic prediction algorithm.

Vicon CaraPost algorithms

The following topics give an overview of the core algorithms which are used within CaraPost, with the aim of supplying information that will enable you to make better use of CaraPost. The algorithms covered are:

- *Blob detection algorithm*
- *Solve calibration brim algorithm*
- *3D initialization algorithm*
- *Tracking algorithm*
- *Initialize from ROM algorithm*
- *Bundle adjustment algorithm*
- *Gap interpolation algorithm*

Blob detection algorithm

CaraPost uses a four-step process to detect and locate blobs on each camera image. The Vicon CaraPost blob detection algorithm is based upon machine learning techniques, producing a classifier that has learned what a blob looks like, based on many examples. The classifier gives as output a probability (between 0 and 1) that a given image pixel is at the center of a blob.
Vicon CaraPost has two classifiers: one to detect white blobs, and the other to detect black blobs.

Having used the classifier to detect possible blobs, CaraPost then uses image processing techniques to extract the boundary of each blob, and finally fits an ellipse to each extracted blob.

Using the blob-detection algorithm simply involves defining the range of blob sizes which should be considered, and setting a threshold on the probability above which a blob is considered to be present. Usually a probability threshold of around 0.5 will work well for a wide range of skin-types and lighting scenarios.

Note

Although the blob-detection algorithm copes well with very low lighting conditions, it will fail to detect blobs in parts of an image which are fully saturated (i.e. pixel values for a region are all set to 255).

Solve calibration brim algorithm

The aim of the solve calibration brim algorithm is to initialize 3D points (and corresponding 2D image blob matches) for the points on the calibration brim, and also calculate initial estimates of the positions and orientations of the four cameras.

The algorithm takes as input a frame in the take containing:

- Blob detections, and
- A small number (4 or more) of brim points that you specified, which have been matched to blob detections across all four camera views.

The algorithm has three basic steps:

1. Calculate an initial estimate for the four camera positions and orientations, using the brim correspondences you made.
2. Look for additional matched blob detections across the four camera views, using the estimated camera positions and (known) 3D brim structure.
3. Refine the camera positions, using all matched brim points.

If you provide four brim correspondences across the four camera views, this should be sufficient to enable the algorithm to work well.

3D initialization algorithm

The aim of the initialize 3D algorithm is to initialize 3D points (and corresponding 2D image blob matches) for as many facial markers as possible. The algorithm uses as input a frame in the take containing all of the following:

- A solved calibration brim; and
- Blob detections; and
- A small number of (around 5–10) matched facial marker points that you have specified, which are used as seed points to search for additional facial marker points.

The algorithm is summarized below:

1. Exclude all brim points, and all unused detections in the image regions corresponding to the brim.
2. For each pair of camera caddies, starting with the seed-points you specified, look iteratively for new matched 2D points across the two camera views, until no more matches are found. At each iteration, use the epipolar constraint (i.e. for pair of cameras of known relative position/orientation, a point in one camera image will lie upon a corresponding line in the other camera image) to find any unambiguous candidate matches which generate lie within a specified distance of the current 3D points.
3. Take the results from the two pairs of camera caddies, and merge any points which are within a specified 3D distance tolerance.
4. Perform bundle adjustment to optimize the final 3D point and camera positions.

Tracking algorithm

The aim of CaraPost's automatic tracking algorithm is to track each 3D marker in the marker set from one frame to the next, tracking the 3D positions of as many marker points as possible through the whole take. It does this by matching blob appearance in each camera view from one frame to the next.

The algorithm is summarized below. For each marker point in the initial frame:

1. Estimate the 3D position of the marker in the next frame in the take, and calculate the corresponding 2D position of the marker in each camera view.
2. For each camera view, find the blob detection in the next frame which is the best match, based upon its position and appearance. If no matching blob detection is found, try and find a possible match by tracking a gray-level image patch.
3. Recalculate the 3D marker positions in the next frame based upon the matched blobs, and perform bundle adjustment to re-estimate the camera positions in the next frame.
4. Based upon the new 3D marker positions, reject any blob match candidates (and corresponding 3D points) which we believe to be incorrectly matched. Blob matches may be rejected either if the 2D re-projection error(s) for the 3D point are too high, or if the new 3D marker position is too inconsistent its neighbors.

Note the following important points:

- Fast facial motions are more difficult to track than sequences where the face does not move very much. This is because the marker may move further from one frame to the next, and the algorithm must consider many more blob match-candidates, which may result in more incorrect matches.
- Dense marker sets are in general harder to track than sparse marker sets. Again, for a dense marker set, the algorithm must consider many more blob match-candidates than for a sparse marker set, which may result in more incorrect matches.

Initialize from ROM algorithm

The aim of the initialize from ROM algorithm is to enable you to use an existing ROM (or range of motion) take to initialize a new take of the same actor and marker set. The function gives an efficient workflow for re-initializing a new take, using everything from the ROM take, including: camera calibration parameters, calibration brim structure, brim markers, facial markers, marker labels, and sticks.

Note

You can choose whether to initialize the full face and brim markers, or just the brim markers only.

Initialize from ROM uses the automatic tracking algorithm to match the brim markers and facial markers from a frame in the ROM take to a frame in the uninitialized take. Thus, the values set in the **Tracking Parameters** properties also affect the results of the Initialize from ROM algorithm. Note that the most useful tracking parameters for ROM initialization have been copied as a separate **Initialize from ROM Parameters** section, so that they can have different values for ROM initialization than for normal frame-to-frame tracking.

For example, if the positions of the facial markers in the current frame of the uninitialized take are a long way from the corresponding facial markers in the ROM frame, you may need to increase the **Face Point Tracking Distance** tracking parameter, and similarly if the positions of the brim markers are a long way from the corresponding brim markers in the ROM frame, you may need to increase the **Brim Point Tracking Distance** tracking parameter.

Bundle adjustment algorithm

The bundle adjustment algorithm is a process whose aim is to take an initial estimate of the camera extrinsic parameters (their 3D positions and orientations), and an initial estimate of a set of 3D points in the take, and improves these estimates by a non-linear optimization process. The optimization refines both camera parameters and 3D point locations by finding the values that most accurately predict the locations of the observed corresponding 2D blob centers in the four camera images.

Bundle adjustment should generally give an improvement to the quality of the 3D point data (and camera positions). The most important time to perform bundle adjustment is during the 3D initialization process, after matching a set of facial seed point correspondences, and before performing automatic 3D initialization.

Bundle adjustment may also give a slight improvement to the overall results if performed once all tracking and editing and clean-up operations have been completed for a take.

Gap interpolation algorithm

The aim of the gap interpolation algorithm is to provide a means to fill in any remaining gaps in the 3D data, once all approaches for tracking marker data through the take have been exhausted.

Gap interpolation is performed using a non-linear interpolation technique, which predicts the location of a missing marker point in a frame, based on its last known location, and also how its neighboring marker points move between the frame where the point was last present and the current frame. The final predicted point position will be (optionally) a blend of the following two results:

- The predicted point position calculated by predicting forwards based upon its last known location just before the gap being filled.
- The predicted point position calculated by predicting backwards based upon its last known location just after the gap being filled.

The gap interpolation algorithm also includes an option to use single sensor tracking gap-filling. When this option is enabled, for any cases where a 3D point is attached to a blob detection in a single camera view, the interpolation algorithm constrains its 3D result to be consistent with the attached detection, ie, the 3D interpolated point must re-project exactly onto the attached 2D blob center.

Vicon CaraPost user interface

The following topics contain lists of the controls, options, properties and their values that are found in Vicon CaraPost, together with brief descriptions and other relevant information.

Vicon CaraPost menus

The following menus are available at the top of the CaraPost window:

- *File menu*
- *Edit menu*
- *View menu*
- *Process menu*
- *Help menu*

Tip

As in most Windows applications, to toggle the display of keyboard shortcuts for menu items (that is, to see the relevant letter underlined), press the Alt key. You can then use the Alt key in combination with the required letter to use the shortcut. Additional shortcuts are displayed on the menus next to the relevant options. For more information, see *Shortcuts and mouse interactions* on page 96.

File menu

The File menu enables you to carry out various operations on your project files

Option	Description
Open (Ctrl + O)	Opens .*cara* project files
Clear	Deletes the currently open project file.
Save (Ctrl + S)	Saves the current project state to a file. If you are working on a new project, you are prompted for a name.
Save As	Saves the current project state to a file that you specify.
Import Pico Files	Imports a collection (the files are grouped) of .*pico* video files
Import Calibration Brim [svg]	Enables you to import your own custom calibration brim into Vicon CaraPost as an .*svg* file. For information, see *Modify the calibration brim* on page 24.
Import XCP	Enables you to import an .*xcp* (calibration file), usually created in Vicon CaraLive.

Option	Description
Import Labels From Cara File	Enables you to import a set of marker labels from an existing Cara file. You can then use this information in any take captured using the same marker set as used in the existing *.cara* file.
Import Labels From Text File	Enables you to import a set of marker labels from a text file. You can then use this information in any take captured using the same marker set defined in the text file. The text file format is defined in section **. Note that importing labels from a text file adds to any existing labels. Any names in the text file which duplicate existing label names will have the string – `Copy` appended to the label name.
Export XCP [Current Frame]	Enables you to export the camera intrinsic (FOV and radial) values for individual cameras as well as their extrinsic parameters (their spatial relationship to one another). You can use this information (exported in *.xml* format) in another take captured during the same session as the current project. For more information, see *About calibration data (XCP) files* on page 103.
Export C3D	Enables you to export the processed data as a *c3d* file. It can then be further cleaned up and/or re-targeted in an animation package, such as Vicon Blade.
Export FBX	Enables you to export the processed data as an *fbx* file. It can then be loaded into an animation package such as Autodesk MotionBuilder.
Recent Files	Point to this option to display a list of the last eight project files that were opened in CaraPost.
Clear Recent Files	Clears the **Recent Files** list.
Exit	Closes Vicon CaraPost.

Edit menu

The Edit menu contains the following options:

Option	Description
Undo (Ctrl+Z)	Undoes the last action.
Redo (Ctrl+Y)	Repeats the last action that was undone.
Clear History	Clear all items from the Undo/Redo history. If you use this option, you can no longer Undo/Redo any previous actions, but will free up a significant amount of application memory.
Create Points From Detections	For each detection in the current selection set, creates a new point and attaches the detection.
Delete Selected (Ctrl+D)	Deletes the currently selected point(s) or detection(s) for the entire sequence
Split Selected Point	From the current frame forward, splits the currently selected point into two points, with two different identifiers. This is useful if you want to delete mismatched or swapped points.
Merge (M)	Merges the currently selected points/detections into one point.
Clear Selected [Current Frame]	For selected points, removes 3D information from the current frame only, but does not delete detections.
Clear Selected [Active Range]	For selected points, removes 3D information within the current Active Range, but does not delete detections.
Detach Detections [Current Frame] (Ctrl+H)	With a camera view and a point selected, detaches the 2D information in the active camera view, or explicitly selected camera view(s) from the selected points. Operates on detections that are present in the current frame only, Tip: Because this option operates on detections that are present in the current frame, it is most useful when you are configuring the first frame before tracking.
Detach Detections [Active Range] (Ctrl+Shift+H)	With a camera view and a point selected, detaches the 2D information in the active camera view, or explicitly selected camera view(s) from the selected points. Operates on detections that are present in the current Active Range,

Option	Description
Relocate Files	Opens the **Relocate Files** dialog box. When you save a *.cara* file, the absolute file locations of the *.pico* files are stored. If you store the *.pico* files in the same folder as the *.cara* file and move the folder, if possible CaraPost automatically finds the *.pico* files, but if the *.pico* files cannot be found (or if you wanted to change one or more of the *.pico* files used in a project) this dialog box enables you to locate the *.pico* files.
Increment Image Sequence Offset	Fixes out of sync *.pico* files. You must have one or more camera views selected: these camera views are moved one frame forward. Do not use after 2D information is given.
Add Target Track (Shift+A)	Enables you to create a new point and a target track keyframe where you click in a camera view. When active, **Add Target Track** is displayed below the channel labels in each camera view.
Edit Target Track (Shift+E)	With a point selected, enables you to click in a camera view with a connected detection to remove that detection and add a target track keyframe where you click. If there is no connected detection in the camera view, a target track keyframe is added to the point at the location you clicked on. When active, **Edit Target Track** is displayed below the channel labels in the camera views.
Stop Point Edit mode	Cancels Edit mode.
Gaps To Keys	With a point selected, adds keyframes either side of any gaps in the tracking.
Remove Unused Detections	Deletes any detections in all camera views on all frames that have not been added to a point.
Remove Interpolated 3D Points	Removes 3D data for selected points for the active range, where that data has been generated via (3D) interpolation (i.e. there are less than two camera views of 2D data for that point and so its 3D position is unconstrained). If no points are selected, removes interpolated 3D data for all points for the active range.
Rename Point (Ctrl+R)	Enables you to change the label of the selected point.
Clear All Labels	Removes all current labels from the list in the Labeling panel.

Option	Description
Delete Selected Labels	Removes selected labels only from the list in the Labeling panel.
Unlabel Selected Points (Ctrl+U)	Detaches any labels from all selected points.
Create Stick from Selected Point (S)	With two points selected, creates a stick between them.
Create Stick Mesh	Using the current frame, creates a simple mesh of all points on the face that have 3D data.
Edit Stick Color	Enables you to change the color of selected sticks.
Clear Active Range	Detach all detections (ie clear 3D points) from the active range, but does not delete detections.
Restore Properties To Defaults	Resets all properties to the system defaults.
Read Blob Parameters From Pico Files	Reverts blob detection parameters to their settings when the currently loaded *.pico* files were imported. This is useful if you make changes to the blob detection settings (the values in the **Blob Parameters** section of the **Properties**), but then need to revert to their original settings.

View menu

The View menu enables you to change the way the Vicon CaraPost displays information:

Option	Description
Panels	Enables you to select or clear any of the User Interface Panels to be displayed: Labeling, History, Selected Point, Log, Properties, Selection, Tracker.
New Floating Workspace	Displays a new floating window which contains a workspace with all the options available in the main workspace. Defaults to a 3D view. Tip: Opening a new floating workspace is useful when you have multiple monitors or if you want to see all your camera views and display another view type at the same time.

Option	Description
Show Calibration Brim SVG	If you have imported an svg of a custom calibration brim, **Show Calibration Brim SVG** opens a floating window showing the svg image.
Full Screen (F)	Enables CaraPost to occupy the full screen, removing the title bar and covering the start bar.
Toggle Dock Widget Visibility (Ctrl+F)	Toggles the display of all items in the main window other than the main workspace.
Toggle Blob Preview (Ctrl+B)	Toggles the display of blob previews (displays the blobs that would be detected with the current **Blob Parameters** settings if you were to run the **Detect Blobs** process).
Toggle Show Stabilized (Ctrl+Shift+S)	Toggles the display of stabilization results. When this option is selected, stabilized results are displayed in 3D view. When cleared, unstabilized results are displayed.
Go To Frame (Ctrl + G)	Displays a dialog box that enables you to enter a frame number and to go to that frame.
Go to Start Frame (Home)	Displays the first frame of the take.
Go to End Frame (End)	Takes you to the last frame of the take.
Play (Space)	Plays through the take. The Play button changes to Stop while the take is playing. Unlike scrubbing on the timebar, plays every frame, so this option is slower.
Next Frame (Right arrow)	Scrubs to the next frame. When held down, acts like Play (see above).
Prev Frame (Left arrow)	Scrubs to the last frame, when held down acts like reverse Play.
Lock To Point	Toggles whether or not the currently displayed view(s) are locked to the currently selected point(s). When the views are locked to a point or set of points, those points are always displayed at the center of the displayed views as you play through the take.

Option	Description
Next Key Frame (Shift + Right arrow)	With a point selected that contains target track keyframes, displays the next keyframe. If the last keyframe is currently displayed, this option displays the first keyframe. Note: This option acts on keyframes on the selected point, that is, in any camera view. If no point is selected or there are no keyframes on the selected point, then this option produces the same result as Next Frame (see above).
Prev Key Frame} (Shift + Left arrow)	With a point selected that contains target track keyframes, displays the previous keyframe. If the first keyframe is currently displayed, this option displays the last keyframe. Note: This option acts on keyframes on the selected point, that is, in any camera view. If no point is selected or there are no keyframes on the selected point, then this option produces the same result as Prev Frame (see above).
Select Point (Ctrl + P)	Displays the **Select Point** dialog box, in which you can enter the ID of a point that is to be selected
Get Point Information	Provides information about the currently selected point, this includes the 3D coordinates on the current frame, its dimensions in each sensor, the distance to centroid of the cameras in each caddy (pair of cameras) and to the centroid of all the cameras. (In this context, centroid is averaged 3D position.)
Rotate Selected Sensors Clockwise (Ctrl + Right arrow)	Rotates clockwise the video from the selected sensor(s). To select a sensor, either Shift + click in its view or select it the **Selection** panel. The orientation of the video is saved to the *.cara* project file.
Rotate Selected Sensors Anticlockwise (Ctrl + Left arrow)	Rotates anticlockwise the video from the selected sensor(s). To select a sensor, either Shift + click in its view or select it the **Selection** panel. The orientation of the video is saved to the *.cara* project file.
Select Point With Fewest Gaps (Ctrl + A)	Searches for the point with the fewest gaps and selects it. This helps you select the next point on which to work during data clean. Only works with a tracked take.

Option	Description
Go To Next Gap of Selected Point	If the selected point has a gap in it, this option navigates to the frame before the gap.
Distance Between Two Points	Writes the 3D distance between two selected points to the log. The units are defined by the calibration brim coordinates (by default, all Vicon calibration brims [1.0, 1.1 and 2.0], in mm).
Set Active Range Start (Ctrl+1)	Sets the start of an Active Range to the current frame.
Set Active Range End (Ctrl+2)	Sets the end of an Active Range to the current frame.
Reset Active Range (Ctrl+3)	Resets the Active Range to the whole take.
Set View Range to Active Range	Sets the view range to the current Active Range. Setting a view range is especially useful in long takes, where you want to work on a particular section of the whole take. It enables you to use the Active Range to run operations on a subsection within the view range.
Reset View Range	Resets the view range to the whole take.
Center Selection All 2D Views (C)	Adjust all 2D views so that the currently selected point(s) are displayed in the center of the view(s). If no points are selected, this menu item has no effect.
Frame Selection All 2D Views (Shift+C)	Adjust all 2D views so that the currently selected point(s) are close-framed in the view(s). If no points are selected, this menu item has no effect.

Process menu

The Process menu provides the following options:

Option	Description
Detect Blobs [Active Range]	Uses the settings from the **Blob Parameters** section of the **Properties** pane to find blobs in every frame in the active range of the sequence to create detections. The detections are not joined between frames (i.e. there can be a detection for the same marker on frames 0 and 1, but this option does not connect them). For more information on how blob detection works, see *Blob detection algorithm* on page 44. This process can be quite time-consuming so before running it, ensure you have the correct settings by enabling **Blob Preview** and scrubbing though the take to check the detections. Note: Using **Detect Blobs [Active Range]** does not enable you to spot gaps in the tracking using the **Continuity** or **Selected Point** panes.
Detect Blobs [Current Frame]	Uses the settings from the **Blob Parameters** section of the **Properties** pane to find blobs in the current frame in the sequence to create detections. For more information on how blob detection works, see *Blob detection algorithm* on page 44. This option is useful for detecting blobs in the first frame prior to initializing the 3D points for the frame, before tracking the entire take.
Load ROM	Loads a *.cara* file (normally a ROM file) that can be used to initialize the current take. For more information, see *Initialize from a ROM file* in the *Vicon CaraPost User Guide* and also *Initialize from ROM algorithm* on page 48.
Initialize from ROM	Initializes the current take using the data (the face as well as the brim) from a *.cara* file that has been loaded using the **Load ROM** option (see above), using the parameters set in the **Initialize From ROM Parameters** section of the **Properties** pane. For more information, see *Initialize from a ROM file* in the *Vicon CaraPost User Guide* and also *Initialize from ROM algorithm* on page 48.

Option	Description
Initialize from ROM [Brim Only]	Initializes the current take using the data (the brim only) from a *.cara* file that has been loaded using the **Load ROM** option (see above), using the parameters set in the **Initialize From ROM Parameters** section of the **Properties** pane. For more information, see *Initialize from a ROM file* in the *Vicon CaraPost User Guide* and also *Initialize from ROM algorithm* on page 48.
Initialize Default Calibration Brim [v2.0]	Loads a 3D model of the markers on the current version (V2.0) of the default calibration brim. Important: The default calibration brim is an object that is used by Vicon CaraPost to calculate the positions of the cameras on a per-frame basis. To enable CaraPost to do this correctly, it must use the correct calibration brim, If you are using a custom calibration brim, you must import it using the **Import Calibration Brim [svg]** option on the **File** menu instead.
Initialize Default Calibration Brim [v1.1]	Loads a 3D model of the markers on V1.1 of the default calibration brim. Important: The default calibration brim is an object that is used by Vicon CaraPost to calculate the positions of the cameras on a per-frame basis. To enable CaraPost to do this correctly, it must use the correct calibration brim, If you are using a custom calibration brim you must import it using the **Import Calibration Brim [svg]** option on the **File** menu instead.
Initialize Default Calibration Brim [v1.0]	Loads a 3D model of the markers on V1.0 of the default calibration brim. Important: The default calibration brim is an object that is used by Vicon CaraPost to calculate the positions of the cameras on a per-frame basis. To enable CaraPost to do this correctly, it must use the correct calibration brim, If you are using a custom calibration brim you must import it using the **Import Calibration Brim [svg]** option on the **File** menu instead.

Option	Description
Solve Calibration Brim [Manual]	If you have a camera calibration loaded, have initialized the correct calibration brim, and connected four or more detections to the calibration brim, this process calculates the camera positions based on the available information. For more information, see *Solve calibration brim algorithm* on page 45.
Initialize 3D	Takes the current camera positions and tries to connect detections in different camera views and calculate 3D positions for them, using the settings defined in the **Initialize 3D Parameters** section of the **Properties** pane. To enable this process to work well, you must have reasonably accurate camera positions, as well as a number of hand-matched 'seed' points. The camera positions and 'seed' marker points are normally derived from solving the calibration brim, manually adding blob correspondences to give the 'seed' points, and finally, using a **Bundle Unconstrained** option (see below). For more information, see *3D initialization algorithm* on page 46.
Bundle Current Frame [Unconstrained]	Adjusts the 3D positions of all marker points so that the difference between each reprojected marker point and the center of the detection it is based on (re-projection error) is minimized across all camera views. Acts on the current frame only and adjusts the camera positions to minimize the re-projection error. For more information, see *Bundle adjustment algorithm* on page 49.
Bundle Sequence [Unconstrained]	Adjusts the 3D positions of all marker points so that the difference between each reprojected marker point and the center of the detection they are based on (re-projection error) is minimized across all camera views. Acts on the entire sequence and adjusts the camera positions to minimize the reprojection error. For more information, see *Bundle adjustment algorithm* on page 49.

Option	Description
Track Forwards [Sequence]	Takes the points in the current frame and tracks them for the rest of the take by matching detections between frames, using the settings defined in the **Tracking Parameters** section of the **Properties** pane. This option also calculates the 3D positions of cameras and points for every frame, and performs blob detection if it hasn't already been done. For more information, see *Tracking algorithm* on page 47.
Track Backwards [Sequence]	Takes the points in the current frame and tracks them to the beginning of the take by matching detections between frames, using the settings defined in the **Tracking Parameters** section of the **Properties** pane. This option also calculates the 3D positions of cameras and points for every frame, and performs blob detection if it hasn't already been done. For more information, see *Tracking algorithm* on page 47.
Track Forwards [Single Frame]	Takes the points in the current frame and tracks them into the next frame by matching detections between frames, using the settings defined in the **Tracking Parameters** section of the **Properties** pane. It also calculates the 3D positions of cameras and points for the next frame, performs blob detection if it hasn't already been done, and aligns the calibration brim based on the previous frame. For more information, see *Tracking algorithm* on page 47.
Track Backwards [Single Frame]	Takes the points in the current frame and tracks them into the previous frame by matching detections between frames, using the settings defined in the **Tracking Parameters** section of the **Properties** pane. It also calculates the 3D positions of cameras and points for the previous frame, performs blob detection if it hasn't already been done, and aligns the calibration brim based on the current frame. For more information, see *Tracking algorithm* on page 47.

Option	Description
Stabilize Data [Manual]	Takes a minimum of three gap-free points that you select as being relatively unaffected by changes in expression throughout the Active Range, and then for each frame maps those points on the current frame to the same points on the base frame. Vicon CaraPost reports the stabilization error at the end of the stabilization process. For more information see *Manual Stabilization* in the *Vicon CaraPost User Guide*.
Stabilize Data [Assisted]	Like **Stabilize Data [Manual]** (see above), operates on a minimum of three gap-free points that you select. Normally produces better results than manual stabilization, as the assisted process enables you to customize the stabilization parameters to suit your particular take, and to benefit from Vicon CaraPost's ability to contribute further stable points, based on your original selection, minimizing the stabilization error across the whole of the Active Range. For more information see *Assisted Stabilization* in the *Vicon CaraPost User Guide*.

Option	Description
Align Data	Aligns the face data into a coordinate frame that you define by selecting three 3D face points. Note: If you are stabilizing a take it is preferable to perform stabilization before you align the data. This is because if you stabilize after aligning data, the markers are returned to their original positions, as stabilization operates on the original, raw data, so you will need to repeat the **Align Data** operation. For more information, see *Align face data in 3D space* in the *Vicon CaraPost User Guide*.
Interpolate Gaps	After you have tracked and corrected a take, use this option to fill in any remaining gaps, using the settings defined in the **Gap-Filling Parameters** section of the **Properties** pane. Fills in (or interpolates) gaps in the 3D point data, where either a 3D point is not present at all, or a point may be present but with only a detection connected in a single view. The function fills all gaps in the 3D information in the active range of the take for all selected points, or if no points are selected, for all points. Gap-filling is performed using a non-linear interpolation technique (see *Gap interpolation algorithm* on page 50), using the options selected in the **Gap-Filling Parameters** section of the **Properties** pane. If necessary, you can remove the interpolated data afterwards by selecting **Remove Interpolated 3D Points** on the **Edit** menu.

Help menu

The **Help** menu provides the following options:

Option	Description
View latest help on vicon.com	Enables you to view the latest version of the help files and PDFs on the Vicon website (requires an internet connection).
View installed help	Enables you to view the help files and PDFs that were installed with Vicon CaraPost.
Check For Updates	Checks for updates to CaraPost over the internet.
Licensing	Opens VAULT, the Vicon licensing tool, where you can manage the licensing options for CaraPost.
Show 3rd Party Licenses	Opens a window detailing all the third party licenses used by CaraPost.
About	Opens a dialog box that displays the CaraPost software version, license used, and copyright information.

Labelling pane

The **Labelling** pane enables you to add text labels to the 3D marker points. The following screenshot shows a **Labelling** pane displaying five point labels.

The **Labelling** pane contains the following controls:

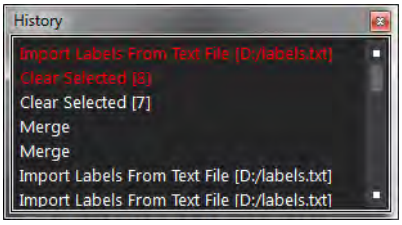
Control	Description
Clear All Labels button	Removes all current labels from the list in the **Labelling** panel and detaches from any labeled points.
Delete Selected Labels button	Removes selected labels only from the list in the **Labelling** panel and detaches from any labeled points.
List of labels	Displays the list of current labels. Labels which have not yet been assigned to points are shown in red; labels which have been assigned to points are shown in gray; and labels which have been assigned to points, but which are not present in the current frame, are shown in yellow. To edit a label name, double-click on the label in the list.

For information on how to label points, see *Label face markers* in the *Vicon CaraPost User Guide*.

History pane

The **History** pane displays a list of all commands in the Undo/Redo list which have been applied to the current Vicon CaraPost take. You can undo all commands displayed in white, using **Edit** > **Undo** (or Ctrl-Z), and all commands displayed in red can be re-done using **Edit** > **Redo** (or Ctrl-Y).

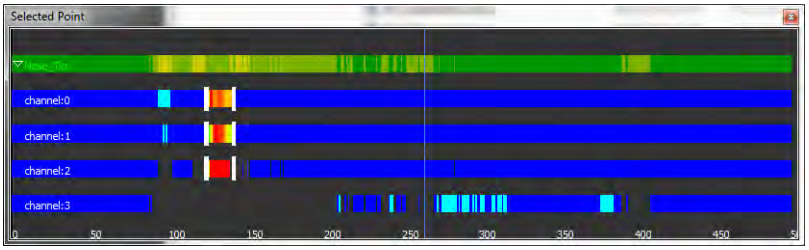
The following screenshot of part of the History pane shows an Undo list of seven actions, with the first two actions having been undone by the user.



Selected Point pane

The **Selected Point** pane (**View** > **Panels** > **Selected Point**) displays detailed information about the tracking continuity of the currently selected point. If multiple points are selected, or no points are selected, the **Selected Point** pane is empty.

The following screenshot shows a **Selected Point** pane.



The Selected Point Pane displays five colored bars indicating the tracking continuity results for the pane across the take:

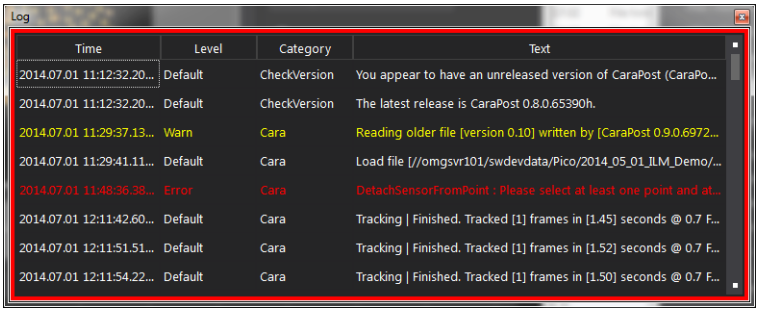
- The top bar displays the point label, and the overall continuity for the point. It gives identical results to those displayed for the point in the Continuity Chart. The colors indicate the continuity of the point across the four cameras:

- Dark green indicates the point is tracked in all 4 channels.
- Light green indicates that the point is tracked in 3 channels.
- Yellow indicates that the point is tracked in 2 channels.
- Orange indicates that the point is tracked in 1 channel only (i.e. single sensor tracking or interpolation).
- Red indicates that the point is simply an interpolated or predicted result.
- A gap in the bar indicates that no result exists for the point at that frame in the take.

- The remaining four bars display the continuity for the point in each individual camera sensor or channel. The colors indicate the continuity of the point in each individual camera:
  - Dark blue indicates that the point has been tracked in the sensor by matching blob detections.
  - Light blue indicates that the point has been tracked in the sensor by patch-tracking (and may indicate slightly less good quality results).
  - Vertical white lines indicate keyframes which have been added by the user using the Target Tracker. If the Target Tracker has been used to track the point between the keyframes, the bar will be colored between green and red to reflect the quality of the target tracking results: green indicates a good target-track match score; red indicates a poor target-track match score.
  - A gap in the bar indicates that no result exists for the point at that frame in the take, for that sensor.

Log pane

The **Log** pane displays logging information output from Vicon CaraPost. The following screenshot shows output from a **Log** pane.



Informational messages are displayed in white; warnings are displayed in yellow, and errors (indicating that an operation as failed or something unexpected has happened) are displayed in red.

Properties pane

The Properties pane contains the following sections:

- *Display section*
- *Selection section*
- *Blob Parameters section*
- *Initialize 3D Parameters section*
- *Initialize From ROM Parameters section*
- *Tracking Parameters section*
- *Gap-Filling Parameters section*
- *Stabilization Parameters section*
- *Export section*
- *Shortcut section*
- *Show Again Flags section*

Display section

The **Display** section of the **Properties** pane enables you to control which point labels are displayed, and also how they are displayed.

Control	Description
Point Labels : All	If selected, all assigned point labels are displayed in 2D and 3D views. By default this check box is cleared (False).
Point Labels : Selected	If selected, assigned point labels are displayed for selected points on 2D and 3D views. By default this check box is selected (True).
Point Labels : Pre-Selected	If selected, assigned point labels are displayed for pre-selected points in 2D and 3D views. Pre-selected points are points that the mouse is hovering over and that are flashing. By default this check box is selected (True).
Font Size	Enables you to set the font size of displayed point labels. By default this is set to 10.

Selection section

The Selection section of the Properties pane enables you to control which objects may be selected in the 2D and 3D views.

Control	Description
Points	If selected, points are selectable 2D and 3D views. By default this check box is selected (True).
Detections	If selected, detections are selectable 2D and 3D views. By default this check box is selected (True).
Sensors	If selected, sensors (cameras) are selectable 2D and 3D views. By default this check box is selected (True).
Sticks	If selected, sticks are selectable 2D and 3D views. By default this check box is selected (True).

Blob Parameters section

The Blob Parameters section of the Properties pane enables you to control the way in which Vicon CaraPost detects blobs. For more information, see *Blob detection algorithm* on page 44.

Control	Description
White Min Radius	The minimum length of the longest axis of the ellipse that makes up a white blob (in pixels). Default is 4 pixels. In a similar way to Max Radius, when specifying this value, you have to balance the possibility of setting this value too high and missing markers against setting it too low and slowing the tracking and detecting unwanted blobs.
White Max Radius	The maximum length of the longest axis of the ellipse that makes up a white blob (in pixels). Default is 50 pixels. If you set this value higher than required, tracking is slowed and unwanted blobs may be detected. If you set the value too low, markers may not be tracked for the entire take if they change shape. If you suspect this may happen, use **Blob Preview** on the relevant frames to make sure you will get good tracking throughout the take.
Black Min Radius	The minimum length of the longest axis of the ellipse that makes up a black blob (in pixels). Default is 4 pixels. In a similar way to Max Radius, when specifying this value, you have to balance the possibility of setting this value too high and missing markers against setting it too low and slowing the tracking and detecting unwanted blobs.

Control	Description
Black Max Radius	The maximum length of the longest axis of the ellipse that makes up a black blob (in pixels). Default is 50 pixels. If you set this value higher than required, tracking is slowed and unwanted blobs may be detected. If you set the value too low, markers may not be tracked for the entire take if they change shape. If you suspect this may happen, use **Blob Preview** on the relevant frames to make sure you will get good tracking throughout the take.
Find Black Blobs	When selected, the blob detector searches for black markers. Default is True (selected).
Find White Blobs	When selected, the blob detector searches for white markers. Default is False (cleared).
Black Threshold	Range is 0–1, default is 0.4. This value changes the level at which black blobs are detected. Vicon CaraPost compares each potential blob it finds with an internal model of blob appearance, learned from many example blobs. The threshold value determines how rigorously it applies the criteria for deciding whether it has found a blob. At zero almost everything is detected as a blob, whereas at 1, only blobs that match precisely are detected. To detect more blobs, lower this value. If too many blobs are detected, you can delete them later.
White Threshold	Range is 0–1, default is 0.4. This value changes the level at which white blobs are detected. Vicon CaraPost compares each potential blob it finds with an internal model of blob appearance, learned from many example blobs. The threshold value determines how rigorously it applies the criteria for deciding whether it has found a blob. At zero almost everything is detected as a blob, whereas at 1, only blobs that match precisely are detected. To detect more blobs, lower this value. If too many blobs are detected, you can delete them later.

Initialize 3D Parameters section

The **Initialize 3D Parameters** section of the Properties pane enables you to control the way in which Vicon CaraPost performs the Initialize 3D algorithm. For more information, see *3D initialization algorithm* on page 46

Option	Description
Max Allowed Depth	The maximum 3D distance (in mm) that a new 3D points can be from one of the existing points during 3D Initialization. Thus, this value should be a bit larger than the maximum spacing between 3D marker points in the marker set. Generally, you should lower the value for very dense marker sets (to avoid possible mismatched detections), and raise the value for a more sparse marker set. The default value is 35mm.
Max 3D Distance To Merge	The maximum 3D distance (in mm) for which two 3D points are considered to be the same point when merging 3D point reconstructions between the two separate camera caddies. The value should be large enough to tolerate some initial reconstruction/camera error between the two camera caddies, but not so large as to merge points which are truly separate. The default value is 5mm.
Max Allowed Symmetric Epipolar Distance	The maximum allowed 2D error metric (in pixels) when trying to match a detection in one camera to a corresponding detection in the other camera on the same caddy. The default value is 5 pixels.

Initialize From ROM Parameters section

The options in the Initialize from ROM Parameters section enable you to specify the settings that affect how the take is initialized from a loaded ROM take.

For further information, see *Initialize from ROM algorithm* on page 48. Note also that the **Tracking Parameters** properties also affect the results of the Initialize from ROM algorithm.

Control	Description
ROM Frame Index	This value determines which frame of the ROM take is used to initialize the current take when the **Initialize from ROM** processing step is called. Select the index of the ROM frame in which the 3D position of the facial markers and brim markers is closest to the current frame being initialized.
ROM Face Point Tracking Distance	This value determines how far (in pixels) CaraPost looks from the ROM frame to the current frame of the take when matching points on the face. The higher this value, the further the face points in the ROM frame can be from their positions in the current frame, but increasing this value may also result in more mismatched points between the ROM frame and the current frame. The default is 50 pixels.

Control	Description
ROM Brim Point Tracking Distance	This value determines how far (in pixels) CaraPost looks from the ROM frame to the current frame of the take when matching points on the brim The default is 30 pixels. The higher this value, the further the brim points in the ROM frame can be from their positions in the current frame, but increasing this value may also result in more mismatched points between the ROM frame and the current frame.
ROM Max Allowed 3D Point Error	This value specifies the distance that a 3D point has to be from its expected position in 3D, estimated based upon the locations of its neighboring points in the ROM frame. 3D points that are found at a distance greater than this value from their expected position are regarded as bad points and detections that cause the incorrect position of the point are detached from the point in the current frame. If you increase the value of **ROM Max Allowed 3D Point Error**, fewer detections (and corresponding 3D points) will be detached (and corresponding 3D points removed) due to being regarded as bad, but a greater number of mismatches may also occur; conversely, if you are getting mismatched points during ROM initialization, try decreasing the value of **ROM Max Allowed 3D Point Error**. The default is 5mm. Note that for this parameter to have an effect, the **Remove Bad 3D Points** tracking parameter must be set to its default value of True.

Tracking Parameters section

The options in the **Tracking Parameters** section of the **Properties** pane enable you to control Vicon CaraPost tracking.

For more information, see *Tracking algorithm* on page 47

Option	Description
Patch Match Threshold	This value sets the threshold at which a potential match (either between two detections, or two patches), is considered to be a true match. At a value of 1, each frame must perfectly match; at a value of 0, almost anything is a match. The default is 0.8.
Face Point Tracking Distance	This value determines how far (in pixels) CaraPost looks from one frame to the next when performing blob matching (or patch tracking) for marker points on the face. The default is 35 pixels. The higher this value, the more facial movement CaraPost can handle, but increasing this value may also result in more mismatched points. For sequences involving normal facial motion (that is, with no extreme expressions or big changes), if you need to reduce the number of mismatched points, turn on **Outlier Removal Threshold 3D** (see below). If you still have too many mismatches, decrease the **Tracking Distance** value.
Face Point Max Elastic Prediction Distance (pixels)	This value determines how far (in pixels) CaraPost looks from one frame to the next when looking for matching blob (or patch tracking), for marker points on the face, after some initial blob matches have been made and Elastic Prediction techniques can be used to estimate a more accurate position for each tracked facial marker point. The default is 15 pixels, and in general, you should set this value to be smaller than that of the **Face Point Tracking Distance**. The higher this value, the more facial movement CaraPost can handle, but increasing this value may also result in more mismatched points.

Option	Description
Brim Point Tracking Distance	This value determines how far (in pixels) CaraPost looks from one frame to the next when looking for an initial matching blob (or patch tracking), for marker points on the brim. The default is only 15 pixels, as the brim is intended to be relatively fixed from one frame to the next. The higher this value, the more movement of the brim CaraPost can handle, but increasing this value may also result in more mismatched brim points.
Patch Match Size	This parameter controls the size of the area that is used to compare potential matches between two frames, when matching detections. The default is 15 pixels, which means that patch-tracking searches for patches of 15x15 grayscale pixels that are centered on a detection. If you increase this value, fewer outlier matches are found, but the overall number of matches may be reduced.
Remove Bad 2D Projections	This toggles whether matched detections are removed from reconstructed 3D points based on their 2D reprojection error. Enabling this setting reduces the number of mismatched points. The default is selected (that is, set to True).
Max Allowed 2D Projection Error	A 2D reprojection error is the difference between the position of a 3D point when it is projected back onto the 2D image, and the center of the 2D detection in the image. The **Max Allowed 2D Projection Error** defines the maximum allowed value for 2D reprojection error. If **Remove Bad 2D Projections** is enabled, all detections with reprojection errors above the threshold are detached from the corresponding 3D point. If you increase the value of **Max Allowed 2D Projection Error**, fewer detections are detached (and corresponding 3D points removed) due to being regarded as mismatches, but a greater number of mismatches may occur. For takes containing a lot of movement or dense clouds of markers, ensure **Remove Bad 2D Projections** is selected (set to True) and adjust **Max Allowed 2D Projection Error** to obtain the most accurate results. The default is 5 pixels.

Option	Description
Remove Bad 3D Points	If selected, this option further reduces the number of mismatched points/detection pairings by detaching all detections that cause a 3D point to be reconstructed in a location which is inconsistent with the location of neighboring points in the marker set. Its use results in cleaner data, but it slightly increases the processing time. The default is selected (that is, set to True).
Max Allowed 3D Point Error	This value specifies the distance that a 3D point has to be from its expected position in 3D, estimated based on the locations of its neighboring points. 3D points that are found at a distance greater than this value from their expected position are regarded as 'bad' points and detections that cause the incorrect position of a point are detached from the point. If you increase the value of **Max Allowed 3D Point Error**, fewer detections (and corresponding 3D points) are removed due to being regarded as bad points, but a greater number of mismatches may also occur. As for 2D removal of bad points, for takes containing a lot of movement or dense clouds of markers, ensure **Remove Bad 3D Points** is selected (set to True) and adjust the Max Allowed 3D Point Error value to obtain the most accurate results. The default is 5mm.

Option	Description
Allow Single Sensor Tracking	If selected (set to True), CaraPost enables the single sensor tracking option during the tracking algorithm. This feature allows an estimate of 3D point positions to be made based upon a correspondence between a 3D point and a detection in a single sensor (or camera) in certain cases, by constraining the 3D point to lie on the ray associated with the single detection. Enabling this feature can allow CaraPost to track points in certain cases where they are occluded in three of the four camera views, resulting in better continuity of tracking across the take. As long as the Single Sensor Tracking Threshold (see below) is set sufficiently tightly, enabling this feature should not decrease the overall accuracy of the tracked 3D points. With this option cleared (set to False), single-sensor tracking is disabled.
Single Sensor Tracking Threshold	This value specifies the maximum allowed 3D distance (in mm) that a 3D point calculated using Single Sensor Tracking can be from its expected position in 3D, estimated based upon the locations of its neighboring points. 3D points calculated at a distance greater than this value from their expected position are not included as Single Sensor Tracking results. The default value for this threshold is 0.5mm, and the value should be set no higher than the maximum tolerated error in mm of the final 3D point positions.
Blob Size Difference Threshold	This value specifies a threshold which determines how similar a blob must be in width and height from one frame to the next to be considered a possible match. The value varies from 0 (no difference in blob width or height allowed) to 1 (any difference in blob width or height allowed). The default value for this threshold is 0.75, which allows quite a large frame-to-frame difference in blob width or height to be considered a possible match. In cases where you get markers occluding each other, resulting in merged blobs, (for example a hemispherical marker occluding a flat marker), it may be useful to reduce the threshold to around 0.1 so that the merged blobs are discounted as possible blob match candidates.

Gap-Filling Parameters section

The options in the **Gap-Filling Parameters** section of the **Properties** pane enable you to control how gap-filling is performed during the Interpolate Gaps processing step.

For more information, see *Gap interpolation algorithm* on page 50.

Control	Description
Overwrite Existing Filled Gaps	This value determines whether existing filled gaps (ie existing 3D points with either zero or one detection only attached) should be recalculated and overwritten during the gap-filling process. By default, this check box is cleared (False).
Use Forwards Prediction	This value determines whether or not the 3D point at the start of a gap (if it exists) is used during the Interpolate Gaps calculation. Generally, you will obtain the smoothest and best-looking results by using both Forwards and Backwards Prediction, resulting in a 3D point position which blends nicely between the start and finish of the gap. By default, this check box is selected (True).
Use Backwards Prediction	This value determines whether or not the 3D point at the end of a gap (if it exists) is used during the Interpolate Gaps calculation. Generally, you will obtain the smoothest and best-looking results by using both Forwards and Backwards Prediction, resulting in a 3D point position which blends nicely between the start and finish of the gap. By default, this check box is selected (True).
Allow Single Sensor Tracking Gap-Filling	This value determines whether 3D points should use a single correspondence between a 3D point and a detection to help calculate the 3D interpolated point position (by constraining the 3D point to lie on the ray associated with the detection). In general, enabling this feature will result in slightly more accurate interpolated 3D point positions than basic interpolation. By default, this check box is selected (True)

Stabilization Parameters section

The **Stabilization Parameters** section of the **Properties** pane enables you to control the settings that are used for assisted stabilization:

For more information, see *Stabilize a take* in the *Vicon CaraPost User Guide*.

Property	Descriptions
Use Brim Points	Whether brim points are included in the set used for stabilization. Default is cleared. Caution: Only use brim points if the brim is relatively stable with respect to the head throughout the shot.
Base Frame Weighting	A number >=0 which gives the weight applied to the Base Frame Stabilization Errors in the overall stabilization error. Giving a higher weighting to this property places more emphasis on minimizing the total amount that each point moves in the take relative to the base frame. Default is 1.0
Inter Frame Weighting	A number >=0 which gives the weight applied to the Inter Frame Stabilization Errors in the overall stabilization error. Giving a higher weighting to this property places more emphasis on minimizing the frame-to-frame point noise (jitter). Default is 1.0
Stable Points Percent	Number between 0 and 1, representing your estimate of how many points in each frame of the take are stable. For takes with very little movement set this number to be as high as possible. For takes with a lot of facial movement, lower this value to reflect the fact that in each frame a significant proportion of the face is moving, relative to the points in the base frame. Default is 0.25.
Max Processing Iterations	Maximum number of stabilization optimization iterations. A larger number results in a slower but potentially better stabilization. Default is 50.

Export section

The **Export** section of the **Properties** pane enables you to change the options for exporting FBX and C3D files.

For more information, see *Export a take* in the *Vicon CaraPost User Guide*.

Shortcut section

The **Shortcut** section of the **Properties** pane displays commonly used Vicon CaraPost shortcuts. For more information, see *Shortcuts and mouse interactions* on page 96.

Show Again Flags section

The **Show Again Flags** section of the **Properties** pane enables you to control whether certain dialogs appear at application startup.

Control	Description
Show Version Check Dialog On Startup	If selected, a dialog box is displayed at start-up if a more recent version of Vicon CaraPost is available for download. By default this check box is selected (True).

Selection pane

The **Selection** pane enables you to select explicitly which channels (or cameras) particular editing or processing functions are applied to.

For example, **Process** > **Detect Blobs**, **Edit** > **Detach Detections** makes use of any explicit camera selections defined in the **Selection** pane.

In the following example, channel (camera) 0 is selected in the **Selection** pane.

Tracker pane

The Tracker pane contains the target tracking controls and settings:

Control	Description
< Track back to start	Tracks the currently selected point(s) back to the beginning of the take if possible, given the current settings. It also does this in all camera views which have a 2D point in the currently active frame,
|< Track back one frame	Tracks the currently selected point(s) backward one frame, if possible, given the current settings. It does this in all camera views that have a 2D point in the currently active frame.
>| Track forward one frame	Tracks the currently selected point(s) forward one frame, if possible, given the current settings. It does this in all camera views that have a 2D point in the currently active frame.
> Track forward to end	Tracks the currently selected point(s) all the way to the end of the take, if possible, given the current settings. It does this in all camera views that have a 2D point in the currently active frame
Clear Tracked Points	Deletes any 2D information (including keyframes) for the currently selected point only.
Add Keyframe To Current Frame	Adds keyframes to all selected points in all camera views which have a 2D location on the current active frame. It also does this in all camera views which have a 2d point in the currently active frame.
Settings	When selected, the following settings are displayed in the **Tracker** pane.
Stop after N Frames, N=	When selected, you can specify the number of frames that are affected by Track backward (<) and Track forward (>).
Overwrite Existing Tracks	When selected, causes the Target Tracker functions (>, >|, < or |<) to overwrite existing 2D data for selected points during tracking.
Automatically Add Keyframes	When selected, if you use one of the tracking controls, a keyframe is added for all selected points before starting tracking if a keyframe does not already exist.

Control	Description
Stop On Next Keyframe	When selected, forces tracking to stop when a keyframe for any selected point in any camera view is encountered.
Allow Single Sensor Tracking	If selected, the Target Tracker attempts to reconstruct 3D points based upon tracking one detection only, by constraining the 3D point to lie on the ray associated with the single detection.
Break Threshold	Defines how well the best patch found in a new frame has to match the patch of the previous keyframe (and the next keyframe, if that has been defined). The higher this number is, the more likely it is that tracking will stop, but the less likely it is that the target track will drift off course. This value determines the minimum quality of the track. Lower values allow tracking to continue when the patch in the current view has changed in appearance from the patches on the keyframes. The parameter value can vary between 0 and 1.0, with a default value of 0.9.
Update Timeline Every Frame	When selected, the continuity data indicating the quality of the target tracking results is updated in the Continuity Chart and Selected Point pane during target tracking.
Update 2D View Every Frame	When selected, any video or 3D view in your workspace is updated to show the frame being tracked. This option slows down the tracking process but enables you to monitor progress.

Graph view

The **Graph** view enables you to visualize the 3D (world) and 2D (image) coordinates of selected point(s), for a selected sensor. It can help you to investigate point stability and/or jitter for selected point(s).

To display the Graph view:

1. Ensure that the view in which you want to view the graph is active and that you have selected the point(s) for which to display a graph.
2. At the top right of the camera views, click **Graph**.

   A **Graph** view for Sensor 0 of the currently selected point(s) is displayed, with the x-axis displaying the frame number, and the y-axis displaying the coordinate value, which for 3D coordinates is in mm, and for 2D coordinates is in pixels.

Use the following mouse/menu actions to work in the **Graph** view:

Required action	Menu/mouse combination
Display a graph for a sensor other than sensor 0	Right-click on the view, and on the on the context menu, point to **Sensor**, and then click the required sensor name.
Zoom in to the graph range for the data	Right-click on the view and then click **Fit to Data**.
Pan in the Graph view	SHIFT+left+right mouse buttons (or middle mouse button) and drag
Zoom in the Graph view	SHIFT+right mouse button and drag

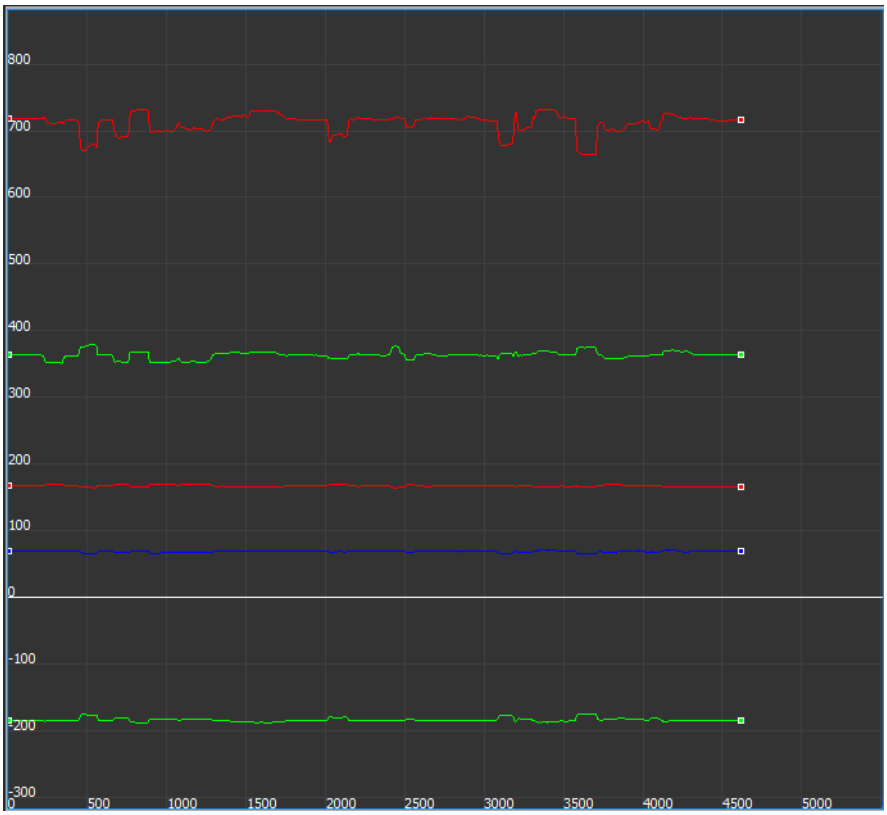
For each selected point, the **Graph** view displays either:

- Three lines (if the point is not visible in the selected sensor), representing the 3D point x, y, and z coordinates plotted against frame number; or
- Five lines (if the point is visible in the current sensor), representing 3D point x, y, and z coordinates, and additionally, 2D image x and y coordinates, plotted against frame number.

The line color indicates which coordinate is being plotted:

- Red = x
- Green = y
- Blue = z

The following example **Graph** view shows the 3D and 2D coordinate data for a single point, for sensor 0.



For a complete description of the options available for the view, see *2D, 3D, Graph and Continuity view shortcuts* on page 98 and *Graph context menu* on page 95.

Continuity chart

The **Continuity** chart is a view in Vicon CaraPost that displays information about which camera views (channels) contribute data to each reconstructed 3D point.

To display the Continuity Chart:

1. Ensure that the view in which you want the **Continuity** chart to appear is active.
2. At the top right of the camera view(s), click **Continuity**.

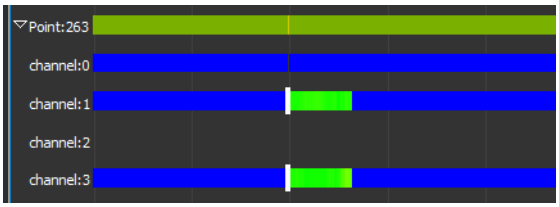
In the **Continuity** chart, for each point, a colored bar indicates how many channels are contributing to the reconstruction of the 3D point. The meaning of the colors is:

Color	Number of contributing channels
Dark Green	4 channels
Light Green	3 channels
Yellow	2 channels
Orange	1 channel: single-sensor tracking or interpolation
Red	0 channels: fully interpolated 3D point
None (ie a gap)	1 or 0 channels, but with no interpolated 3D point present

In the **Continuity** chart you can double-click a point to reveal more detailed channel information for that point, identical to that displayed in the **Selected Point** pane. The meaning of the colors is:

Color	Meaning
Dark blue	Blob detection used.
Light blue	Patch tracking used. (Patch tracking is used if blob detection fails to find a blob and may indicate lower quality data.)
White bar	A keyframe target-tracked detection is present in that channel.
Colored bar between bright green and red in color	A non-keyframe target-tracked detection is present in that channel. The color of the bar (between green and red) indicates the health of the 2D target track: green indicates good target tracking results in that channel; red indicates poor target tracking results.
Light blue	Patch tracking used. (Patch tracking is used if blob detection fails to find a blob and may indicate lower quality data.)

The following screenshot of part of a continuity chart shows the detailed channel contributions for Point 263, which is present in three camera views, with (good) target tracking results in channels 1 and 3 for part of the range.



Context (right-click) menus

Vicon CaraPost provides the following context (right-click) menus:

- *View context menu*
- *2D context menus*
- *3D context menu*
- *Graph context menu*
- *Continuity context menu*

View context menu

The View context menu is displayed when you right-click on the current view (3D, Graph, Continuity, or camera view). It contains the following options:

Option	Description
View > 3D, Graph, Continuity, Sensor 0, Sensor 1, Sensor 2, Sensor 3	Displays the selected view type
Sensor > Sensor 0, Sensor 1, Sensor 2, Sensor 3	Enables you to change between the camera views

2D context menus

If one of the 2D (camera view) options is selected in the View context menu (that is, Sensor 0, Sensor 1, Sensor 2 or Sensor 3), and you right-click in a camera view, the following options are available:

Option	Description
Fit To Window	Fits the whole video from the current camera view to the view area.
Actual Size	Zooms the video so every pixel in the video is shown as one pixel in the view.

Option	Description
Start Measurement	Enables you to drag a line and display its length in pixels. This can be useful for working out which **Max Radius** and/or **Min Radius** to use for blob detection.
Clear Measurement	Clears any measurement lines from the current camera view.
Center Selection	Center the selected object(s) in the current camera view.
Frame Selection	Frame the selected object(s) in the current camera view.
Border	When selected, if **Image** is not selected, the 2D view displays a checkerboard pattern as the background. When not selected, the background is gray.
Image	When selected, the video image is rendered for all camera views. Tip: For faster scrubbing, clear this option.
Detections	When selected, detections are created when the following Process menu options are selected, and are rendered in all views: ▪ Detect Blobs [Sequence] ▪ Detect Blobs [Current Frame] ▪ Track Forwards [Sequence] ▪ Track Forwards [Single Frame] For a definition of detections, see the *Glossary* on page 127.
Points	When selected, points are displayed in all camera views. For a definition of points, see the *Glossary* on page 127.
Projections	When selected, projections are rendered in all the camera views. Projections will be rendered differently, depending on whether they are connected or unconnected, but neither is rendered if this option is not selected. For a definition of projections, see the *Glossary* on page 127.

Option	Description
Connected	When selected, connected projections are rendered. Note: If **Projections** (see above) is not selected, selecting **Connected** does not render connected projections. For a definition of connected projections, see the *Glossary* on page 127.
Unconnected	When selected, unconnected projections are rendered. Note: If **Projections** (see above) is not selected, selecting **Unconnected** does not render unconnected projections.
Errors	When selected, the error is rendered in the camera view. The error is the difference between the center point of a projection and its associated (connected) detection or target track in that camera view.
Blob Preview	When selected, displays a preview of the result of running **Detect Blobs** on the current view with the current **Blob Parameters** settings. This is useful for tuning these settings before running the process.
Pixel Report	When selected, a pixel report is displayed in the top-left of the camera view. The pixel report shows the current x and y coordinates of the mouse pointer in the video camera view, followed by the graylevel value (0–255). This can be useful for working out the intensity value inside and outside a blob, so you can set the **Threshold** correctly.
ROM	When selected, displays information from the currently loaded ROM file. (For more information, see For more information, see *Initialize from a ROM file* in the *Vicon CaraPost User Guide*.)
Sticks	When selected, displays any sticks in the current file. For information on creating sticks and meshes, see *Create a mesh* in the *Vicon CaraPost User Guide.*
Point Labels: All	When selected, displays all custom point names. For more information, see *Label face markers* in the *Vicon CaraPost User Guide*.

3D context menu

If the **3D** option is selected in the View context menu, and you right-click in the workspace, the following options are available:

Option	Description
Fit to Points	Trucks\translates the 3D view to fit the selected points into the view. If no points are selected, it will try to fit all the points in the view.
Center Selection	Center the selected object(s) in the current camera view.
Frame Selection	Frame the selected object(s) in the current camera view.
Face Points	If selected, the face points are rendered in the 3D view.
Calibration Brim Points	If selected, the calibration brim points are rendered in the 3D view.
Reconstruction Type Colors	If selected, face-points in the 3D view are colored according to their reconstruction type, ie yellow indicates that the point was reconstructed from a number of matched detections, orange indicates that the point contains one or more patch-matched, and red indicates that the point is fully predicted or interpolated. If cleared, face-points in the 3D view are displayed in yellow.
Cameras	If selected, the cameras are rendered in the 3D view. If cleared, cameras are not displayed in the 3D view.
Rays	Camera rays are lines from points back to the point in each camera that it is connected to. If the **Rays** option is selected, camera rays are displayed for the selected points.
ROM	If selected, displays points in the ROM frame in the 3D view (if one is loaded) as a set of pale blue dots. If cleared, ROM points are not displayed in the 3D view.
Sticks	If selected, displays sticks in the 3D view if you have created a stick mesh. If cleared, sticks are not displayed.
Point Labels :All	If selected, displays all point labels in the 3D view.

Graph context menu

If the **Graph** option is selected in the View context menu, and you right-click in the workspace, the following options are available:

Option	Description
Fit to data	If selected, the graph view is zoomed\translated to fit the data for the selected point(s). If nothing is selected, this option has no effect.
Track Data	If selected, the 2D information for the selected point is displayed.
Recon Data	If selected, the x, y and z coordinates of the selected point are plotted in the Graph view (Red=x, Green=y and Blue=z).

Continuity context menu

If the **Continuity** option is selected in the View context menu, and you right-click in the workspace, the following options are available:

Option	Description
Collapse all	Collapses all the points in the Continuity view so none of the individual channel data is shown.
Expand all	Expands all the points so you can see all the sensor data for every point.
Fit to data	Displays the selected points in the continuity view. If no points are selected then all points are displayed.

Shortcuts and mouse interactions

Many keyboard shortcuts are displayed in the Vicon CaraPost dialog boxes and menus. For lists of available shortcuts, also see the following topics:

- *File management shortcuts*
- *Playback shortcuts*
- *Display shortcuts*
- *Camera view shortcuts*
- *2D, 3D, Graph and Continuity view shortcuts*
- *Navigation shortcuts*
- *Selecting points*
- *Point editing shortcuts*

Tip

As in most Windows applications, to toggle the display of Alt keyboard shortcuts for menu items (that is, to see the relevant letter underlined), press the Alt key. You can use the Alt key in combination with the underlined letter to use the shortcut. Additional shortcuts are displayed on the menus next to the relevant options.

File management shortcuts

To open and close Vicon CaraPost files:

Action	Key
Open .*cara* file	Ctrl+O
Save currently open .*cara* file	Ctrl+S

Playback shortcuts

To control playback:

Action	Key
Start and stop playback (equivalent of clicking the Play button next to the timebar	Spacebar

Display shortcuts

To display/hide Vicon CaraPost dialog boxes and panes and information within them:

Viewing option	Key combination
Display context menu	Right-click
Hide/display currently displayed windows (so that only the camera views and timebar are displayed)	Ctrl+F
Toggle full screen display	F
Toggle blob preview	Ctrl+B
Display hidden window	Right-click on menu bar and select required window.
Toggle display of stabilized results	Ctrl+Shift+S

Camera view shortcuts

To manage camera views (channels):

Camera view option	Key combination
Select camera view	Shift+click on the camera view
Zoom in and out of the selected camera view	Shift+right-click and drag
Rotate selected camera view 90° (clockwise)	Ctrl+right arrow key
Rotate selected camera view -90° (anticlockwise)	Ctrl+left arrow key
Move camera view	Shift+middle mouse button/ wheel and drag
Restore all camera views to their default zoom, rotation and scrolling status	Click required camera view layout button: 

2D, 3D, Graph and Continuity view shortcuts

To change the display of the CaraPost views

View option	Key combination
Zoom in and out of the view	Alt or Shift+right-click and drag
Pan view	Alt or Shift+middle mouse button (or Alt or Shift+left+right buttons) and drag
Rotate in 3D view	Alt or Shift +drag
Zoom to selection (in Graph and Continuity views)	Alt+Shift+drag
Set Active Range	Ctrl+drag

Navigation shortcuts

To move through a take:

Navigation option	Key combination
Go to frame by number	Ctrl+G
Go to start frame	Home key
Go to end frame	End key
Go to next frame	Right arrow key
Go to previous frame	Left arrow key
Go to next keyframe	Shift+right arrow key
Go to previous keyframe	Shift+left arrow key
Shuttle time	Middle mouse button+drag (or mouse wheel)
Set active range start	Ctrl+1
Set active range end	Ctrl+2
Reset active range	Ctrl+3

Selecting points

To select and view points in the camera views and **Continuity** window:

Selection option	Key combination
Individually select point or points	Click (or for multiple points Ctrl+click)
Multi-select points	Click and drag
Lasso-select point in camera views and 3D view	Shift+Alt+click and drag
Select point by number	Ctrl+P
Select point with fewest gaps	Ctrl+A
Expand points in Continuity window	Double-click point

Point editing shortcuts

To work with points:

Points option	Key combination
Merge selected points	M
Delete selected	Ctrl+D
Detach detection from point	Ctrl+H
Detach sensor from point	Ctrl+Shift+H
Add target track	Shift+A
Edit target track	Shift+E
Stop editing points	Shift+S
Create stick from selected points	S
Rename point	Ctrl+R
Track the points in the current frame into the next frame	Ctrl+W

Supported file formats and standards

This chapter covers the following topics:

About .pico files

Vicon CaraLive records captures as *.pico* files, which are stored on the selected logger.

The name, description and unique ID of the logger at the time of capture are included as metadata in the *.pico* file, together with other capture details. This information is used by Vicon CaraPost.

To use the recorded and processed data in an animation package, you must export the CaraPost files to the appropriate format (for more information, see *Export formats* on page 107).

About calibration data (XCP) files

To calculate the 3D locations of the markers (and cameras), Vicon CaraPost needs to know the radial and field of view parameters for the cameras used for capture. The calibration *.xcp* file is created in Vicon CaraLive, based on a take of a calibration grid being waved in front of the cameras during the take day.

XCP is the file format used to store the camera calibration information of a Vicon motion capture system. The XCP format uses XML syntax. The following explanation does not contain details of all the valid XML nodes and node-attributes, but is limited to the information required to perform lens distortion correction and to compute the P matrix.

Camera sub nodes

A typical VSK file will contain a `Cameras` node which includes a set of `Camera` sub nodes, each one defining the calibration information for a particular device. In addition to the attributes used to define the identity of the device, the important attributes of a `Camera` sub node are:

- `PIXEL_ASPECT_RATIO` The ratio of the physical dimensions of a pixel
- `SENSOR_SIZE` Defined in pixels
- `SKEW` The skew coefficient between the camera x and y axis

KeyFrame sub nodes

For each `Camera` sub node, most XCP files will have only one `KeyFrame` sub node containing the actual calibration information. However, the XCP file format can also support moving cameras. In this case, calibration information is run-length encoded. Each `KeyFrame` node includes the `FRAME` index attribute, which defines the initial validity of the calibration values in this `KeyFrame`. The calibration is valid until the index defined in the next `KeyFrame`. (Although an XCP with static cameras can be associated with a whole recording session (ie, multiple trials), an XCP with moving cameras is valid for only one specific trial.)

The important attributes of a `KeyFrame` node are:

- `ORIENTATION` 3D rotation using quaternion representation (the last value is the real part)
- `POSITION` 3D translation
- `FOCAL_LENGTH` The camera focal length (in mm)
- `PRINCIPAL_POINT` The point (x,y) in pixels, where the camera principal axis intersects the image. This is usually close to the center of the image.

- VICON_RADIAL Radial distortion information
- IMAGE_ERROR Mean error in pixels (result of the calibration procedure)

Example Cameras node

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<Cameras NAME="" VERSION="1.0">

 <Camera DEVICEID="17980" NAME="T160_17980" PIXEL_ASPECT_RATIO="1"
    SENSOR_SIZE="4704 3456" SKEW="0" >
  <KeyFrames>
   <KeyFrame FRAME="0" FOCAL_LENGTH="4570.1" IMAGE_ERROR="0.3"
      ORIENTATION="0.3 0.9 -0.2 0.1" POSITION="-27.1 33.4 63.1"
      PRINCIPAL_POINT="2400.4 1711.6" VICON_RADIAL="2.7e-009 -2.0e-016"/>
   <KeyFrame FRAME="11" FOCAL_LENGTH="4570.1" IMAGE_ERROR="0.3"
      ORIENTATION="0.3 0.9 -0.2 0.1" POSITION="-27.1 33.4 63.1"
      PRINCIPAL_POINT="2400.4 1711.6" VICON_RADIAL="2.7e-009 -2.0e-016"/>
  </KeyFrames>
 </Camera>

 <Camera DEVICEID="18912" NAME="T160_18912" PIXEL_ASPECT_RATIO="1"
   SENSOR_SIZE="4704 3456" SKEW="0">
  <KeyFrames>
   <KeyFrame FOCAL_LENGTH="4563.6" FRAME="0" IMAGE_ERROR="0.35"
      ORIENTATION="-0.1 0.6 -0.1 0.6" POSITION="72.4 31.3 -24.8"
      PRINCIPAL_POINT="2381.7 1711.2" VICON_RADIAL="2.7e-009 -2.6e-
      016"/>
   <KeyFrame FRAME="11" FOCAL_LENGTH="4570.1" IMAGE_ERROR="0.3"
      ORIENTATION="0.3 0.9 -0.2 0.1" POSITION="-27.1 33.4 63.1"
      PRINCIPAL_POINT="2400.4 1711.6" VICON_RADIAL="2.7e-009 -2.0e-
      016"/>
  </KeyFrames>
 </Camera>

</Cameras>
```

Radial distortion model

Where:

- PIXEL_ASPECT_RATIO = a
- VICON_RADIAL = w_0 w_1
- PRINCIPAL_POINT = x_{pp} y_{pp}

Raw 2D point: $p_r = \begin{bmatrix} x_r \\ y_r \end{bmatrix}$ = Corrected 2D point: $p_C = \begin{bmatrix} x_C \\ y_C \end{bmatrix}$

$$dp = \begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} x_r - x_{pp} \\ a(y_r - y_{pp}) \end{bmatrix}$$

Calculate the radius as $r = \|dp\|$

Calculate the scale factor $s = 1 + w_0 r^2 + w_1 r^4$

Correct radial distortion $p_C = \begin{bmatrix} x_C \\ y_C \end{bmatrix} = \begin{bmatrix} s \cdot dx + x_{pp} \\ (s \cdot dy + y_{pp})/a \end{bmatrix}$

Projection matrix

Where:

- POSITION = t (3-vector)
- ORIENTATION = q (4-vector, quaternion with last component as real part)
- FOCAL _LENGTH = f
- SKEW = k

Convert the quaternion q to 3x3 rotation matrix R

Compose the 3x4 matrix: $P_a = \begin{bmatrix} R & -Rt \end{bmatrix}$

Compose the 3x3 matrix: $K = \begin{bmatrix} f & k & x_{pp} \\ 0 & f/a & y_{pp} \\ 0 & 0 & 1 \end{bmatrix}$

Calculate the 3x4 projection matrix as: $P = K \times P_a$

About label (TXT) files

A label text file provides a convenient way to import a large number of marker labels into Vicon CaraPost (using File > Import Labels From Text File).

To create a label text file, ensure your text file conforms to the following basic formatting rules:

- Labels are defined as space-delimited text fields.
- The name of a label cannot contain a space.
- The file can contain either multiple or single label text fields on each line.

The following example *.txt* file defines eight label names:

```
Mouth_L_Corner
Mouth_R_Corner
Chin_Center
Chin_L_Pt1
Chin_L_Pt2
Chin_R_Pt1
Chin_R_Pt2
Nose_Tip
```

For information on how to use an imported label *.txt* file to label face markers, see *Label face markers* in the *Vicon CaraPost User Guide*.

Export formats

Vicon CaraPost enables you to export motion capture data to the following formats:

- *.xcp* (calibration file). See *About calibration data (XCP) files* on page 103.
- *.c3d* A public domain binary file format for storing 3D motion capture data.
- *.fbx* Filmbox file format: a proprietary binary or ascii file format for 3D motion capture data developed by Kaydara and now owned by Autodesk.

For information about how to export files using Vicon CaraPost, see *Export a take* in the *Vicon CaraPost User Guide*.

Troubleshooting

The following topics contain troubleshooting advice for common issues that can occur when you are using Vicon CaraPost.

- *Blob detection troubleshooting*
- *Initialize 3D troubleshooting*
- *Initialize from ROM troubleshooting*
- *Tracking troubleshooting*
- *Jittery 3D reconstructions troubleshooting*

Blob detection troubleshooting

The following topics offer solutions to commonly encountered blob detection issues.

- *Many blobs on the face or brim have not been detected.*
- *Many blobs have been detected that are not true blobs.*

Many blobs on the face or brim have not been detected.

In the **Blob Parameters** section of the **Properties** pane, ensure that the following parameters have been set appropriately:

- Check that you are detecting the right color of blobs for your marker/ brim setup, and if necessary, select **Find Black Blobs** and/or **Find White Blobs**.
- Check that the **White Min Radius**, **White Max Radius**, **Black Min Radius**, and **Black Max Radius** parameters are set correctly. The minimum radius must be low enough not to reject small blobs, and the maximum radius must be high enough to include all true blobs.
- Check the **Black Threshold** and **White Threshold**. The default value of 0.4 generally finds most blobs, but you may sometimes need to lower this value.

Note that under extreme lighting conditions, where the take appears either very dark or completely saturated (white), blob detection will not be possible.

Many blobs have been detected that are not true blobs.

A moderate number of false blob detections will have no adverse effect on the final tracking results, so unless the effect is very noticeable, you can probably simply ignore this problem.

If Vicon CaraPost is displaying a large number of unwanted blob detections, in the **Blob Parameters** section of the **Properties** pane, ensure that the following blob parameters have been set appropriately:

- Check the **White Min Radius**, **White Max Radius**, **Black Min Radius**, and **Black Max Radius**. If the minimum radius is set too low, the blob detector may detect lots of false small blobs, and similarly, if it is set too high, the detector may detect large false blobs.
- Check the **Black Threshold** and **White Threshold**. The default value of 0.4 generally finds most blobs without detecting many false blobs, but you may sometimes need to raise this value.

Initialize 3D troubleshooting

The following topics offer solutions to commonly encountered initialization issues.

- *Initialize 3D fails to initialize all the face marker points.*
- *Initialize 3D initializes all the face marker points, but some points are mismatched.*
- *Initialize 3D initializes all face marker points, but some are repeated, and are not merged between camera caddies.*

Initialize 3D fails to initialize all the face marker points.

When fixing this problem, you can either add to the existing results (if they are correct but incomplete), or you can first undo (Ctrl+Z) the results of the previous **Initialize 3D** process, before making some or all of the following changes and re-running the Initialize 3D process. Either way should give acceptable results.

You can use two possible approaches to solving this problem, either separately or in combination.

- The first approach is to change the Initialize 3D parameters and run the **Initialize 3D** process again, with the aim of solving the problem automatically. To do this, in the **Properties** pane, try changing the following **Initialize 3D** parameters:
  - Increase the **Max. Allowed Depth** parameter value, especially if the facial marker points are widely spaced.
  - Increase the **Max. Allowed Symmetric Epipolar Distance** parameter.
- The second approach is to add in some or all of the missing marker points by hand-matching detections in regions where the automatic 3D initialization has failed. To do this:
  a. Select matching detections in multiple 2D views and then click **Edit** > **Merge** (or press M).
  b. If needed, you can then re-run the **Initialize 3D** process.
  c. If you have performed purely manual corrections to the initialization, perform a final bundle adjustment to get the best results, by clicking **Process** > **Bundle Frame [Unconstrained]**.

Initialize 3D initializes all the face marker points, but some points are mismatched.

You can use two possible approaches to solving this problem, either separately or in combination:

- The first approach is to undo the previous 3D initialization, change the Initialize 3D parameters and run the **Initialize 3D** process again, with the aim of solving the problem automatically. To do this:
  a. Press Ctrl+Z to undo the results of the previous **Initialize 3D** process.
  b. In the **Properties** pane, try changing the following **Initialize 3D** parameters:
     - Decrease the **Max. Allowed Depth** parameter value, especially if the facial marker points are close together.
     - Decrease the **Max. Allowed Symmetric Epipolar Distance** parameter.
  c. Click **Process** > **Initialize 3D**.
- The second approach is to correct mismatched points by hand. To do this:
  a. Delete any incorrect matches by selecting the incorrectly matched 3D point(s), and clicking **Edit** >**Clear Selected [Current Frame]**.
  b. Re-match the 3D point(s) by selecting matching detections in multiple 2D views and clicking **Edit** > **Merge**.
  c. If you have performed purely manual corrections to the initialization, perform a final bundle adjustment to get the best results, by clicking **Process** > **Bundle Frame [Unconstrained]**.

Initialize 3D initializes all face marker points, but some are repeated, and are not merged between camera caddies.

You can use two possible approaches to solving this problem, either separately or in combination:

- The first approach is to undo the previous 3D initialization, change the Initialize 3D parameters and run the **Initialize 3D** process again, with the aim of solving the problem automatically. To do this:
  a. Press Ctrl+Z to undo the results of the previous **Initialize 3D** process.
  b. In the **Initialize 3D Parameters** section of the **Properties** pane, increase the value for the **Max. 3D Distance to Merge** parameter.
  c. Click **Process** > **Initialize 3D**.
- The second approach is to merge the repeated 3D points by hand. To do this:
  a. For each case, select the two instances of the repeated point and click **Edit** > **Merge**.
  b. To get the best results, perform a final bundle adjustment, by clicking **Process** > **Bundle Frame [Unconstrained]**.

Initialize from ROM troubleshooting

The following topics offer solutions for commonly encountered initialization from ROM issues.

Initialize from ROM does nothing.

Initialize from ROM fails completely if no ROM take has been loaded. This generates an error message in the Vicon CaraPost **Log** pane.

To fix the problem, ensure that you have loaded a ROM take before attempting to run the **Initialize from ROM** process.

Initialize from ROM fails to initialize some or all of the brim points.

Initialize from ROM can fail to initialize the brim either because blob detection has failed, or (more likely) because the tracking parameters used during the process are not set appropriately. Because the **Initialize from ROM** process can only be run once for a take, the recommended workflow for solving problems in ROM initialization is as follows:

1. Undo (Ctrl+Z) the results of the previous **Initialize from ROM** process.
2. Change blob detection, initialize from ROM parameters, or tracking parameters to address the problem, and run **Initialize from ROM** again (see below for suggested changes). Repeat Steps 1 and 2 until you have a reasonable result.
3. Make any final corrections (either deletions or additions) to the initialized points by hand.
4. Perform a final bundle adjustment to get the best results, by clicking **Process** > **Bundle Frame [Unconstrained]**.

   Note that in this case you may also have to assign missing point labels by hand.

In Step 2 above, to decide which blob-fitting, initialize from ROM, or tracking parameters to change:

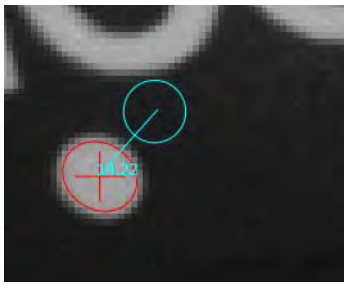
1. Check that the blobs generated for the current frame look appropriate. To do this, right-click in a 2D view and turn on the display of **Points** and **Projections** so that you can see the blob detections.

   If the brim blob detections look incorrect, follow the instructions in *Blob detection troubleshooting* on page 110.

2. Check whether the brim point positions are significantly different in the ROM frame and the current frame. To do this, compare the light blue ROM marker positions for the brim points, and the locations of brim detections in the current frame. If there are significant differences, try one of the following:

   - Increase the **ROM Brim Point Tracking Distance** parameter so that it is larger than the largest distance in pixels between a brim point and its corresponding detection in the current frame.

     To measure this distance, right-click on a camera view and then click **Start Measurement** and click and drag between the two positions. This measures the difference in pixels between the position of a point in the ROM frame and the detection in the current frame;

   or

   - Increase the **ROM Max. Allowed 3D Point Error**, eg, from 5mm to 10mm or higher and also ensure that the **Remove Bad 3D Points** tracking parameter is set to **True**.

Initialize from ROM fails to initialize some face marker points.

Initialize from ROM can fail to initialize the face marker points either because blob detection has failed, or (more likely) because the tracking parameters used during the process are not set appropriately. As Initialize from ROM can only be run once for a take, the recommended workflow for solving problems in ROM initialization is as follows:

1. Undo (Ctrl+Z) the results of the previous Initialize from ROM process.
2. Change blob detection, initialize from ROM parameters, or tracking parameters (see below for suggested changes), and run Initialize from ROM again.
3. Repeat Steps 1 and 2 until you have a reasonable result.
4. Make any final corrections (either deletions or additions) to the initialized points by hand.
5. To get the best results, perform a final bundle adjustment by clicking Process > Bundle Frame [Unconstrained]. Note that in this case you may also have to assign missing point labels by hand.

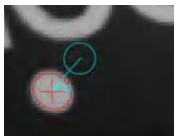
In Step 2 above, to decide which blob-fitting, initialize from ROM parameters, or tracking parameters to change:

1. Check that the blobs generated for the current frame look appropriate. To do this, right-click in a 2D view and turn on the display of Points and Projections so that you can see the blob detections.

   If the facial blob detections look incorrect, follow the instructions in *Blob detection troubleshooting* on page 110.

2. Check whether the face point positions are significantly different in the ROM frame and the current frame. To do this, compare the light blue ROM marker positions for the face points, and the locations of face detections in the current frame. If there are significant differences, try one of the following:

   - Increase the ROM Face Point Tracking Distance parameter so that it is larger than the largest distance in pixels between a brim point and its corresponding detection in the current frame. To measure this distance, right-click on a camera view and then click Start Measurement and click and drag between the two positions. This measures the difference in pixels between the position of a point in the ROM frame and the detection in the current frame.



or

- Increase the **ROM Max. Allowed 3D Point Error**, eg, from 5mm to 10mm or higher and also ensure that the **Remove Bad 3D Points** tracking parameter is set to its default value of True.

Initialize from ROM initializes most/all marker points, but some are mismatched.

Initialize from ROM can mismatch points if the initialize from ROM or tracking parameters used during the process are not set appropriately. Because the **Initialize from ROM** process can only be run once for a take, the recommended general workflow for solving problems in ROM initialization is as follows:

1. Undo (Ctrl+Z) the results of the previous **Initialize from ROM** process.
2. Change the initialize from ROM or tracking parameters to address the problem, and run **Initialize from ROM** again. Repeat Steps 1 and 2 until you have a reasonable result.
3. Make any final corrections (either deletions or additions) to the initialized points by hand.
4. To get the best results, perform a final bundle adjustment, by clicking **Process** > **Bundle Frame [Unconstrained]**.

   Note that in this case you may also have to assign missing point labels by hand.

In Step 2 above, to reduce mismatches, make the following changes to the initialize from ROM and tracking parameters:

1. Check that **Remove Bad 2D Projections** and **Remove Bad 3D Points** are enabled in the **Tracking Parameters** section.

   If they are, decrease the values for **ROM Max Allowed 3D Point Error** in the **Initialize From ROM Parameters** section, and also (less likely to be needed) **Max Allowed 2D Projection Error** in the **Tracking Parameters** section. This is especially true if the facial marker set being used is very dense.
2. (Less likely) Reduce the values of the **ROM Face Point Tracking Distance** or **ROM Brim Point Tracking Distance initialize from ROM parameters**.

Initialize from ROM creates a set of new marker points that appear as new points in the Continuity chart and do not connect to existing marker points.

Initialize from ROM creates a new set of marker points in addition to existing marker points. If you run the **Initialize from ROM** command more than once for a take, it will create a new, independent set of markers each time.

You are therefore recommended to run the **Initialize from ROM** command once only in any given take.

Tracking troubleshooting

The following topics offer solutions for commonly encountered tracking issues.

- *Tracking results in only a few (or one) face markers being tracked.*
- *Tracking results in a significant number of mismatched or swapped marker points.*
- *Tracking results in many brim points failing to track for some frame(s), but face point tracking is OK over the same frame range.*
- *Tracking results in many face marker points failing to track for some frame(s), but brim point tracking is OK over the same range.*
- *When re-tracking selected point(s) during a second tracking pass trying different parameters, the tracking fails, despite good detections in multiple camera views.*
- *Tracking fails to track eyelid marker points.*
- *Tracking fails to track points that move in and out of occlusion in a number of camera views.*

If the Tracking process has generally worked, but reconstructed 3D tracking results for one or more of the points is jittery (or jumpy), see *Jittery 3D reconstructions troubleshooting*

Tracking results in only a few (or one) face markers being tracked.

The most likely cause of this problem is that you have accidentally selected and tracked only a few (or one) points.

If this is the case, undo (Ctrl-Z) the last operation and re-track with all points de-selected; this will cause tracking to be run for all points.

If this was not the cause of the issue, try some of the following techniques for handling tracking failures.

Tracking results in a significant number of mismatched or swapped marker points.

Swapped or mismatched marker points are very time-consuming to fix manually in the final 3D data, so try to prevent this type of tracking error. Depending on the length of the take and the number of errors present, you can choose one of two general workflows to address the problem:

- Undo the tracking result (Ctrl-Z) and re-track the whole take with different tracking parameters.

Or:

1. Delete all mismatched or swapped points. To do this:
   a. Select the points, and set the Active Range to the part of the take where the errors are present.
   b. On the **Edit** menu, click **Clear Selected [Active Range]**.
2. Before re-tracking the data, make sure that in the **Properties** pane, the **Tracking Parameters** are set as follows:
   - Ensure that **Remove Bad 2D Projections** is enabled, and that **Max Allowed 2D Projection Error** is set to a reasonably low value (typically less than 5 pixels).
   - Ensure that **Remove Bad 3D Points** is enabled, and that **Max Allowed 3D Point Error** is set to a reasonably low value (typically less than 5mm, but may need to be lower for a dense marker set).
   - Ensure that the **Face Point Tracking Distance** and **Brim Point Tracking Distance** parameters are not set higher than they need to be for the amount of frame-to-frame motion present in the take. As an example, if the maximum frame-to-frame distance during the take moved by a face point is 50 pixels (as measured using the **Start Measurements** function in the 2D context menu), you should set the **Face Point Tracking Distance** parameter to a value just slightly higher than this, eg, 55 pixels.
3. Re-track only those points that were mismatched with different tracking parameters.

Tracking results in many brim points failing to track for some frame(s), but face point tracking is OK over the same frame range.

If a large number of brim points fail to track at certain frame(s) in the take, to address the cause of the problem you can change the tracking (and possibly blob detection) parameters, and re-run the automatic tracking in the failed frames for all points using **Process** > **Track Forwards [Sequence]** or **Process** > **Track Forwards [Single Frame]**.

Work through the following steps:

1. Are blob detections present for the marker points which failed to track?

   If not, ensure that the **Blob Parameters** are set correctly as explained in *Blob detection troubleshooting* on page 110.

2. Does the brim jump or move significantly at the start of the tracking failures?

   If so, increase the **Brim Point Tracking Distance** tracking parameter so that it is just slightly larger than the maximum frame-to-frame image distance (in pixels) that the brim points move.

3. If steps 1. and 2. fail to address the problem, you may also need to relax other tracking parameters slightly, eg, decrease the **Patch Match Threshold**, and increase the **Max Allowed 2D Projection Error** and **Max Allowed 3D Point Error** parameter values.

Tracking results in many face marker points failing to track for some frame(s), but brim point tracking is OK over the same range.

If a large number of facial marker points fail to track at certain frame(s) in the take, to address the cause of the problem you can change the tracking (and possibly blob detection) parameters, and re-run the automatic tracking in the failed frames for all points using **Process** > **Track Forwards [Sequence]** or **Process** > **Track Forwards [Single Frame]**.

Work through the following steps:

1. Are blob detections present for the marker points which failed to track?

   If not, ensure that the **Blob Parameters** are set correctly as explained in *Blob detection troubleshooting* on page 110.

2. Do the facial marker points which fail to track move significantly at the start of the tracking failures?

If so, increase the **Face Point Tracking Distance** tracking parameter so that it is just slightly larger than the maximum frame-to-frame image distance (in pixels) that the face points move.

3. If steps 1. and 2. fail to address the problem, you may also need to relax other tracking parameters slightly, eg, decrease the **Patch Match Threshold**, and increase the **Max Allowed 2D Projection Error** and **Max Allowed 3D Point Error** parameter values.

When re-tracking selected point(s) during a second tracking pass trying different parameters, the tracking fails, despite good detections in multiple camera views.

Occasionally, you may come across points where, despite trying a range of sensible tracking parameters, and good detections being present, the point still fails to track. This is generally caused by the correct tracked point candidate either:

- Being rejected as a **Bad 2D Projection** or a **Bad 3D Point**; or
- The correct tracked point candidate hitting one of the tracking parameter thresholds, and not being considered further.

To solve this issue, re-track the failed point with the following changes to tracking parameters in the given order:

1. Disable the **Remove Bad 2D Projections** and/or **Remove Bad 3D Points** options, or leave these options enabled, but increase the **Max Allowed 2D Projection Error** and/or **Max Allowed 3D Point Error** parameter values.
2. If there is significant frame-to-frame motion present in the frame where the tracking failure occurs, if the failed point is on the face, increase the value for the **Face Point Tracking Distance** parameter; or if the failed point is on the brim, increase the value of the **Brim Point Tracking Distance**.
3. Decrease the **Patch Match Threshold** parameter value slightly eg, to 0.6.

Tracking fails to track eyelid marker points.

Tracking eyelid marker points can be prone to failure, as these points tend to move in and out of occlusion as the actor blinks, and also may move in a way which is uncorrelated with the motion of neighboring points.

To improve eyelid tracking results, complete the following steps:

1. Re-track the failed point(s) with the following changes to tracking parameters:
   - Ensure that the **Allow Single Sensor Tracking** parameter is enabled.
   - Decrease the **Patch Match Threshold** tracking parameter value slightly e.g. to 0.6.
   - Increase the **Face Point Max Elastic Prediction Distance** tracking parameter value slightly e.g. to 30 pixels.
2. If the point(s) still fail to track, use the Target Tracker to track the failed point(s) manually. For detailed information on how to do this, see *Target tracking* on page 26.

Tracking fails to track points that move in and out of occlusion in a number of camera views.

Tracking marker points which move in and out of occlusion (eg, lip points and/or eye points) can be prone to failure, but there are some options which can improve your chances of tracking the point successfully.

- If the marker is completely occluded in all camera views, then it is will not be possible to track it. In this case, the best you can hope for is to use **Process** > **Interpolate Gaps** to fill in the predicted position of the marker in the frame(s) in which it is completely occluded.
- If the marker point is present in one or more camera views, then try the following:
  a. Ensure that the **Allow Single Sensor Tracking** parameter is set to enabled, and possibly relax the **Single Sensor Tracking Threshold** and **Patch Match Threshold** tracking parameter values slightly (eg, to 0.75mm and 0.6 respectively).
  b. Select the failed points, and re-track automatically in the failed frame(s).
  c. If the point(s) still fail to track, use the Target Tracker to track the failed point(s) manually. For detailed information on how to do this, see *Target tracking* on page 26.

Jittery 3D reconstructions troubleshooting

The following topics describe a number of specific causes of jitter (reconstructed points that, after apparently successful tracking, appear to shake, wobble, or jump slightly as you scrub through a take). In each case, how to detect this case and how to fix it is explained.

Jitter caused by merged blob detections

To detect this issue, look in the four 2D camera views for the jittery point, and check whether any of the blob detections contributing to the point are merged with other blobs as one marker occludes another in one of the camera views.

Fixing merged blob detections

Make use of the **Blob Size Difference Threshold** tracking parameter to force the tracking process not to use the merged blob detections. To do this:

1. Set the Active Range for the take to the frame-range in which the merged detections are present.
2. Select the marker point where the merged detections are present, and detach the incorrect detections across the Active Range by clicking **Edit > Detach Detections [Active Range]**.

   Tip

   If needed you can detach detections in specific camera views only by selecting the appropriate channel(s) in the **Selection** pane.

3. Set the **Blob Size Difference Threshold** tracking parameter to a very low value (e.g. around 0.1).
4. Navigate to the frame before you have detached the detections, and re-track the selected marker point forwards across the required frame range.

   In frames where there are merged detections, the tracker should now patch-track instead.

If the above approach fails, use the Target Tracker (see *Target tracking* on page 26) to perform completely manual re-tracking of the point over the problematic range.

Jitter caused by poorly located detections

To detect this issue, look in the four 2D camera views for the jittery point, and check whether any of the blob detections contributing to the point are poorly located, ie, the displayed blob detection does not outline the blob in the image correctly.

Fixing poorly located detections

If the point is visible in sufficient camera views, detach the camera view(s) containing the poorly located detections from the point across the affected frame-range. To do this:

1. Set the Active Range to the affected range.
2. Select the affected point.
3. In the **Selection** pane, select the camera view to detach.
4. Click **Edit** > **Detach Detections [Active Range]**.

Detaching the detections in the camera view where they are poorly located will cause the 3D point to be recalculated without those detections, giving a less jittery result. If the above approach fails, or if the point is not visible in at least three camera views, use the Target Tracker (see *Target tracking* on page 26) to perform completely manual re-tracking of the point over the problematic range.

Jitter caused by changes in camera contributions

To detect this issue, look in the four 2D camera views for the jittery point, and check whether the number of cameras in which the point is visible changes at the point where the point jumps in 3D.

Fixing changes in camera contribution

Aim to achieve more consistency of camera contributions to the point across the affected range. You can do this either by:

- Detaching contributions from those cameras which are not consistently contributing to the point as described in *Fixing poorly located detections* above; or
- Re-tracking the affected point using different tracking parameters with the aim of filling in missing camera contributions.

  A possible tracking parameter change to make to achieve this is to reduce the Patch Match Threshold slightly (e.g. to 0.6).

If the above approaches fail to solve the problem, use the Target Tracker (see *Target tracking* on page 26) to perform completely manual re-tracking of the point over the problematic range.

Glossary

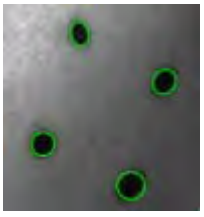
2D reconstruction error The difference between the position of a 3D point when it is projected back onto the 2D image plane, and the 2D detection in that plane.

3D point The path though space that a moving marker follows. This reconstructed data is stored in a file as a time-series of points.

.pico file Capture file that contains video data from a single camera and information about the logger at the point of capture.

.xcp file An XML file, which contains the specified calibration parameters and threshold data for Vicon cameras. An *.xcp* file is created during the calibration process and is used when processing data from the cameras.

Blobs Continuous areas of above-threshold data from the cameras. They are submitted as candidates to Vicon circle-fitting algorithms to determine which are likely to represent markers. Preview blobs are rendered as green rings around potential blobs.



Bundle Using the currently available information, an attempt by Vicon CaraPost to minimize the output error for all cameras by adjusting current values. These may include camera internal data (such as focal length, etc) and external information (such as relationship to other cameras), and blob positions.

Caddy Set of two cameras on a single arm of a Vicon Cara head rig.

Calibration brim A known pattern of markers that is used to calculate the positions of the cameras. The Vicon Cara system is supplied with a default calibration brim. If

required, you can modify the calibration brim, using a template available from Vicon.

Calibration file See *.xcp file*.

Calibration grid A rigid device with a pattern of black blobs on a white background, used by Vicon CaraLive for alignment of cameras for initialization and frame-by-frame calibration.

Camera rays See *Rays*.

Capture See *Take*.

Camera view Onscreen display of view from Vicon camera. Vicon CaraPost enables you to display up to four rectangular views, labeled Channels 0, 1, 2 and 3. These views enable you to preview takes during processing.

Channel See *Camera view*.

Commuter licensing Temporarily moving a seat from a network license to a machine that is not connected to the license server network. This enables you to temporarily use Vicon software on a machine unconnected to the network that contains the license server.

Connected projections Projections that have an associated detection or target track in the current camera view. A connected projection is rendered as a yellow cross within a circle.

Detections Representations of blobs (markers) found in an image. When not selected, they are red circles (or ellipses) with a cross in the middle.

The cross represents the center point, which is used as the 2D coordinate of the detection. The red circle or ellipse represents the boundary of the blob (marker) and is used as part of the blob detection process.

Dolly The moving of a camera along a horizontal axis closer to or further from the subject.

Error The difference between the center point of a projection and its associated (connected) detection or target track in that channel. It is often called the reprojection error. In Vicon CaraPost, it is displayed as a yellow line. The following image shows the center of a detection and a projection, with a yellow line (error) connecting the two.

Headbar See *Calibration brim*.

Interpolation The process by which Vicon CaraLive fills a gap in a reconstruction by calculating a smooth curve between two disconnected points. See also *Reconstruction*.

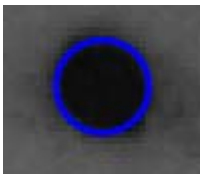
Marker A circle painted on, drawn on, or stuck to an actor; or a sphere, hemisphere or disk coated with highly retroreflective material attached to an actor, whose motion is captured by a Vicon system. Vicon systems capture information from the markers and convert it into images that represent the position and radius of the marker.

Noise Random background interference that is unrelated to the data being captured, causing random spikes, wobbles or jitters in motion data. Some noise is generally present in most data collected and may be caused by a number of factors. Vicon CaraPost offers a number of tools and techniques to help eliminate noise.

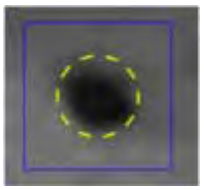
Occlusion The obstruction of a marker from the view of one or more cameras. Occlusion is generally caused by a marker being covered by a body part or another motion capture subject.

Pan The rotation of a stationary camera in a horizontal plane about a vertical axis. See also *Dolly* and *Truck*.

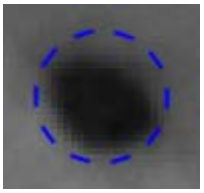
Points Collections of blob detections and\or target tracks as well as projections. When not selected they are blue rings that match up with the boundary of the detection with which they are associated.



If the point is a target track keyframe, it is rendered as a blue box representing the pattern area, and a broken yellow circle representing the middle of the pattern area.



If the point is a target track but not on a keyframe, it will be rendered as a broken blue circle representing the center of the pattern area.



Projections (or Reprojections) See *Reconstruction*.

Rays Lines from points in the 3D world back to the corresponding points in each camera imaging plane.

Reconstruction The process by which Vicon software calculates the position of markers in 3D space and links these points frame-by-frame. This reconstructed data is stored in a file as a time-series of points. 3D points that are calculated from the corresponding 2D image points are known as reconstructions. See also *2D reconstruction error*.

Scrub The movement of the current time cursor left or right along the timebar.

Sensor Camera component that detects light and converts it into an electronic signal, enabling the 2D positions of markers to be recorded over time.

Sequence See *Take*.

Stabilization error The stabilization error is a weighted sum of the following errors:
Base Frame Stabilization Error The root-mean-square of the differences between the percentage specified by the Stable Points Percent most stable points in each frame and the corresponding points in the base frame (calculated across the whole of the Active Range).
Inter Frame Stabilization Error The root-mean-square of the differences between the percentage specified by the Stable Points Percent most stable points in each frame and the previous frame (calculated across the whole of the Active Range).

Take A recording of motion data with a Vicon system, using markers painted on or attached to the subject of the recording (normally an actor). Also known as a capture or sequence.

Target track Target tracking uses pattern matching to calculate tracks.The target tracker tracks the center of the marker.

The blue box contains the pattern that you are tracking (the marker) and should be adjusted to comfortably contain the marker. The green cross represents the comparison region, which is the area in adjacent frames that will be searched to find the pattern that is being matched (ie the marker). The green circle represents the middle of the pattern.

Truck The moving of a camera side-to-side along a horizontal axis. See also *Dolly* and *Pan*.

Unconnected projections Projections that do not have an associated detection or target track in the current camera view. Unconnected projections are rendered as orange x signs surrounded by a circle. See also *Detections* and *Target track*.

Further resources

For more information on Vicon and the Vicon Cara system, see the following resources:

- *Vicon CaraPost video tutorials*
- *Vicon CaraPost sample data*
- *Contact Vicon*

Vicon CaraPost video tutorials

Videos that accompany this guide are available from *Vicon Support*. The videos include the following topics:

- *Introduction and basic overview*
- *Blob detection*
- *Creating and using sticks and labels*
- *Initialization*
- *Tracking*
- *Optimization*
- *Data export*

Vicon CaraPost sample data

You can download sample *.pico*, *.xcp*, and *.cara* files from the *Vicon Support website*, in the same location as the Vicon CaraPost installer.

Contact Vicon

If you need more information than that supplied in the documentation or on the *Vicon Support website*, use the following resources:

Denver, CO

Vicon Denver

7388 S. Revere Parkway Suite 901

Centennial, CO 80112, USA

T: 303.799.8686

F: 303.799.8690

E: *support@vicon.com*

Los Angeles, CA

Vicon LA

5419 McConnell Avenue

Los Angeles, CA 90066, USA

T: 303.799.8686

F: 303.799.8690

E: *support@vicon.com*

Oxford, UK

Vicon Oxford

14 Minns Business Park

West Way, Oxford, OX2 0JB, UK

T: +44.1865.261800

F: +44.1865.240527

E: *support@vicon.com*

Singapore

Vicon Singapore

T: +65 6400 3500

E: *support@vicon.com*

Index

M

P

R

S

T

V