Using xcv

Karen McGaul, Andrew Zisserman and Andrew Fitzgibbon

{kym,az,awf}@robots.ox.ac.uk

Contents

1 Introduction

`xcv` is an application built using the VXL libraries. The aim here is to familiarise you with the facilities offered in `xcv`. These facilities are a selection from the multitude of functions contained in the VXL system. As will be seen, the main menus of `xcv` correspond to the VXL package organisation.

The main libraries used in `xcv`are: vcl (the C++ compatibility layer), vbl (basics), vnl (numerical algorithms), vil (imaging), vgl (geometry), vsl (segmentation) and vgui (graphical user interface).

Historical note: This document describes the new version of `xcv` which uses VXL (a cut-down version of the old TargetJr libraries). The main change is in the graphical user interface which can now use one of a selection of GUI toolkits (including MFC, GTK and Qt).

Getting started: There are a number of example images in *xcv-examples*`/images`. All the images have an ".tif"extension.

Running `xcv`: The program can take as an argument the filename of the image that you wish to load. Most common image formats are supported (e.g. tif, pgm, jpeg, gif), e.g. type

```
xcv basement00.tif
```

2 The Graphical User Interface

2.1 Manipulating a single image

To try the following functions, read in an image by selecting **File → Load Image** from the menu bar.

Normal zooming: The image may be panned directly using the mouse in combination with the control key. While the control key is down, the buttons have the following effects:

- The middle button allows the image to be dragged around the display area.
- The left and right buttons zoom the image about the point where the button was pressed. The left button zooms in and right zooms out.

Smooth zooming: The image may also be zoomed "smoothly". To activate this press the "z" key. While the control key is down the mouse has the following effects:

- As before the middle mouse button allows the image to be dragged around the display.
- The right mouse button now has no effect.
- Clicking the left mouse button anywhere in the image and dragging it upwards zooms into the image. Dragging the mouse downwards zooms out from the image.

2.2 Manipulating a stack of images

To try the following functions, load another image using **File → Load Image** so there are now two images on the stack.

1. To scroll through the deck, use the 'PageUp' and 'PageDown' keys to move forwards and backwards through the deck respectively.
2. Use '+' to add a column of panes to the stack manager and '-' to remove a column. Type '-' to return to the original display. The window can be resized to better fit the images by dragging the window corner with the mouse.
3. Use CTRL and '+' to add a row of panes to the stack manager and CTRL and '-' to remove a row. Type CTRL and '-' to return to the original display.
4. When several images are displayed, clicking on an image makes that image the active pane.
5. To remove the top image, use the menu item **File → Remove Image**.

3 Printing

As well as allowing the user to load and remove images, the File menu also has selections to print the image.

- Selecting **File → Print Image to File** prints the active image to a Postscript file. Entering a value larger than 1 as the 'Reduction factor' will reduce the resolution of the image saved by that factor, i.e. choosing a reduction factor of 4 will save an image with only 1/4 the detail of the original. There is also an option whether or not to save any spatial objects (circles, lines, points) drawn on the image.

4 Display

Display operations are actions that affect only the way in which VXL images and geometry are presented to the user. The following operations are under the **Display** menu. Some take several seconds to run.

1. Image Magnifying Glass
   Enable a magnifying glass by applying: **Display → Toggle magnify lens**.
   A magnifying glass appears. Move mouse around. The area over the mouse indicator is enlarged. To disable the magnifying glass select again : **Display → Toggle magnify lens**.

5 Image Processing

These are image operations where the input is an image, and the output is an image, e.g. image enhancement. The operations can be linear, such as Gaussian smoothing, or non-linear, such as morphology.

Selecting an operation from the Image Processing menu performs that operation on the current active pane. The result of the operation is an image which is placed at the end of the stack of images.

Load an image (the basement ones are good for the following).

Try any operation under "Image-processing". Note that these operations are actually creating a new image that can be further processed. This new image may be saved to disk using the **File → Save Image** menu item.

To get back to the original image, use PageUp and PageDown. If images appear to look completely black, they may just be very dark.

- **Enhance Lens**
  In the following example the enhance lens is used so that the original and processed ROI can be compared.

  From the original image, select **Display → Toggle enhance lens** and select the image corresponding to the latest image processing. Holding down the left mouse button on the image will now superimpose the results of the operation on a small portion of the original image. To switch of the enhance lens select **Display → Toggle enhance lens** again.

- **Region of Interest**
  As image processing can take a long time, use a ”region of interest” (ROI) to quickly preview the results of an operation. Most operations are designed so that they can operate on a ROI in the original image. This is useful if you want to apply a time-consuming operation to a large image—you can first test it on a small ROI, and then when the parameters are correct, apply it to the whole image. To set the ROI use the menu item **Display → Make ROI**.

6 Segmentation

This is the detection of features in images (such as corners, edges ...). Here we consider examples of *local* feature detection.

On the **Segmentation** menu, try the following menu items:

1. **Corner Detection**
   Select the “Harris Corner Detector” on the Segmentation menu, and set “Maximum corner count” = 50 — this restricts display to only the 50 most salient corners. Select OK to apply the corner detector.

   Read in the basement00 image, and select a ROI which includes the front left column (e.g. top left around (50, 160), size about 150×250). Apply the corner detector.

2. **Edge Detection**
   Select “Canny: Oxford” on the Segmentation menu. Apply the operator — this includes both edge detection (convolving with the derivative of a Gaussian, and Non Maximum Suppression) and edge linking using hysteresis. The edges are displayed in green. Note that the linked edges are selected (shown in yellow) as the cursor moves close to them. Now fit lines to the edges by selecting “Dectect lines: Oxford”.

6.1 Modifying Geometric Objects

`xcv` allows you to modify the colour and other properties of the geometric objects produced by the segmentation algorithms.

To set the colour of specific objects, first load an image and run **Segmentation → Canny: Oxford** to create an image with associated edges. Select a few edges by clicking on them with the left mouse button. To deselect a line click on it with the middle mouse button.

Then use the **Geometry → Change selected objects → Change colour** option to select a new colour for the selected edges. The edges will be unselected and displayed in the new colour.

The **Geometry → Change selected objects → Change line width** menu item can be used to change the width of all selected edges in the image.

7 Multiple View Geometry

`xcv` displays and computes the multiview geometric relations between images, for example the epipolar geometry between two images.

7.1 Two Views

Start `xcv` with two images. If the image file names are `basement00.mit` and `basement01.mit` for example then start `xcv` by typing:

```
xcv basement0[01].mit
```

or,

```
xcv basement00.mit basement01.mit
```

To compute F or H matrices we will first need to load or compute corners in both images.

- Use **Multiview → Corners → Load → Load Corners** to load a file containing corners. The corners for the image `basement00.mit` will traditionally be contained in the file `basement00.corners`.
- Use **Multiview → Corners → Compute → Compute Corners - All views** to compute the corners.

Note that if you start `xcv` with more than two images loaded then when performing a two view relation you will be asked to select which two images the operation is to be performed on. Do this by clicking the left mouse button once on each required image.

7.1.1 Viewing epipolar geometry

To view the epipolar geometry between images either load or compute an F matrix for the two views.

- To load an F matrix select **Multiview → Two view→ Load → Load FMatrix**. If the two images loaded are called `basement00.mit` and `basement01.mit` then the standard name for the F matrix would be `basement.00.01.F`.
- To compute an F matrix first compute or load the corners from both images. Then select **Multiview → Two view→ Compute → Compute FMatrix**.

Now pressing the left mouse button in one of the windows displays the epipolar line of the selected point in the other image. Pressing the middle mouse button will leave the point and corresponding line on the display. To stop the display of epipolar lines select **Multiview → Two view→ Display → Toggle display FMatrix**.

7.1.2 Corner matches and tracks

Computing a fundamental matrix also computes correspondences between the corners in each image.

- Choosing **Multiview → Two view→ Display → Display corner tracks** will display a track on each matched point showing its corresponding position in the other image. Note that displaying the tracks will make it difficult to select a point and you may have to remove them from the display (by clicking the right mouse button on the image and choosing **Line Tracks → Visible**.) to select a point.

Corner matches can also be loaded using **Multiview → Two view→ Load → Load corner matches**. The file containing corner matches between `basement00.mit` and `basement01.mit` would normally be called `basement.00.01.cc`.

7.1.3 Viewing planar transformations

To view a planar tranformation between images either load or compute an H matrix.

- To load an H matrix select **Multiview → Two view→ Load → Load HMatrix2D**. A file containing an H matrix normally ends in a `.H` suffix, for example `basement.00.01.groundplane.H`.
- To compute an H matrix, select **Multiview → Two view → Compute → Compute HMatrix2D**.

Now pressing the left mouse button in one of the images displays the selected point as a cross on the other image. To hid this display select **Multiview → Two view→ Display → Toggle display HMatrix2D**. Pressing the middle mouse button will leave the point and corresponding cross on the display.

7.2 Three Views

Start `xcv` with three images. It is now possible to load or compute a trifocal tensor for these three images.

- To load a tri-tensor choose **Multiview** → **Three view** → **Load** → **Load TriTensor**. A file containing a tri-tensor normally ends in a `.T`, for example `basement.00.01.02.T`.
- To compute a tri-tensor first compute or load the corners for all three images. Then choose **Multiview** → **Three view** → **Compute** → **Compute TriTensor**.

Now pressing the left mouse button in any one of the three images displays the epipolar lines in the other two views.

- **Line transfer:**
  Compute lines two of the views. Then select a pair of corresponding lines in these views and select **Multiview** → **Three view** → **Display** → **Transfer Line**. The corresponding infinite line is drawn in the third view.