

## TUTORIAL GIMP



## DIGITISING IMAGES

# Scanning success

Scanners seem rather old fashioned now we have digital cameras... or do they? **Michael J Hammel** shows why not.



**T**hese days the push is on to go all digital, starting right at the moment of image capture with digital cameras. But the cost of digital cameras is still a limiting factor to many users. Considering the cost of which you can acquire still photos, a scanner still offers a more cost effective solution. But cost is not the only factor to consider here. The quality of the image you get from a \$200 digital camera hardly compares to the quality you get from a \$200 scanner that can produce images from slide film or colour negatives. In fact, you can get scanners for under \$100 that will produce very high quality scans – made all the better when processed inside the GIMP.

This issue, we're going to look at what you can do when you start with a scanner, edit in *The GIMP* and process with other, less well-

known freely available graphics tools. There are a number of tools available to Linux users for scanning. Most start with the *SANE* backend drivers, a set of drivers for various scanners including USB and SCSI attached versions. Front ends include *XSane*, *QuiteInsane*, and *Kooka* among others. Epson also provides drivers for its scanners.

## GIMP2 Unleashed!

At the time of writing, final preparations were being made to *GIMP2*; the plug-in Application Programming Interface (API) had been frozen and plug-in developers were being urged to port their work across to the new system. On the surface, *GIMP 2.0* has a brand new, colourful interface complete with an accessible menu bar gracing the top of each work window (a long-requested feature) and a collection of Dockers that can be opened and closed underneath the main tool box. Big changes have also taken place under the surface, including support for the CMYK colour model: *The GIMP* is finally able to deal with colour as it is represented in the real world! The new version is imminent, available at the usual mirrors via [www.gimp.org](http://www.gimp.org). Version 2 will work alongside the previous stable iteration (1.2) so migration issues should be ironed out quite well.



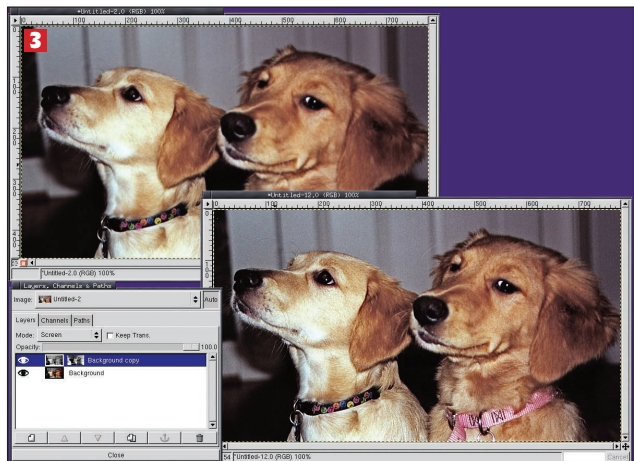
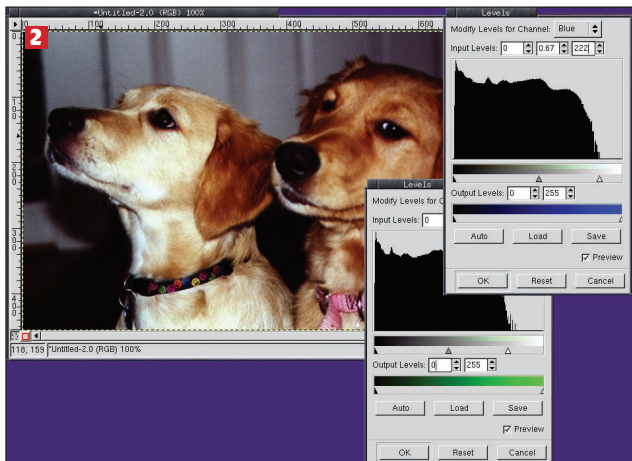
## SCANNING BASICS

Nearly all scans have basic problems that you need to become familiar with, before moving on to more complex projects. In this first tutorial we'll examine four common issues: adjusting the white and black points to fix colour balance, brightening and sharpening the image, and aligning bad scans. You'll need to have the following packages installed: *The GIMP*, the *SANE* drivers, and either *XSane* or *Quitelnsane*.

### Adjust white/black points

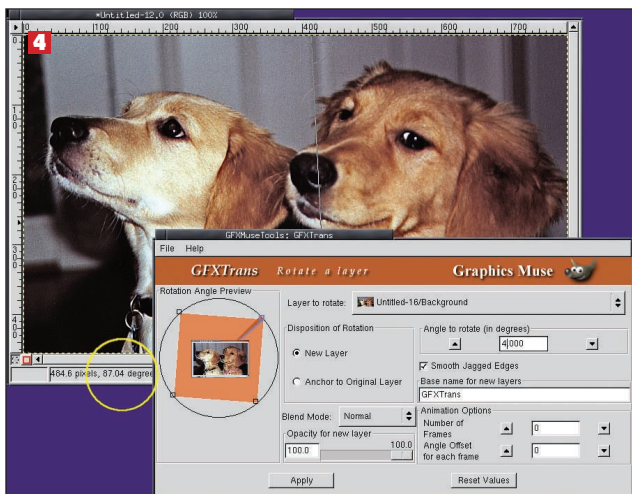
**1** Our scan comes via an Epson Perfection 1260 scanner. The original photo is dark and the colours are a bit too red and too blue. This image was run through AutoLevels once, but the result was not good enough, so the Levels must be adjusted manually.

Start by adjusting the Gray channel in the Levels dialog. Set the white point to 224 but leave the black and midpoints alone. Don't close the dialog – keep it open to adjust the other channels.



### Sharpen

**2** Adjust the Green, Blue and Red Channels individually. Too much red requires adjustment to the Blue and Green Channels. Too much blue requires adjustments to the Red and Green. The result is better colour balance, but the image is still too dark.



### Brighten and sharpen

**3** Duplicate the original layer and desaturate it. Add a white layer mask. Select the duplicate layer and copy it into the mask. Invert the mask and blur it lightly, perhaps by 2 or 3 pixels. Increase the contrast in the mask using the Curves dialog by raising the midpoint up and to the left (effectively making dark pixels lighter). Now change the layer mode to Screen for the duplicate layer. That brightened the dark spots, but it also made it clear that the image is slightly out of focus. Flatten the image and sharpen it using the Sharpen plug-in. The results of before and after sharpening are shown here along with the duplicate layer used to brighten the image.

### Alignment

**4** The image is good now, but we can clean it up just a little more. Many scans will not be aligned vertically or horizontally. This scan has background panels that appear to be slanted. Trace the Measure Tool along one of the panels to see the angle is 87 degrees (outlined in yellow), about 3-4 degrees from vertical. Using the GFXTrans tools (commercial, but you can use the Transform Tool in the toolbox to do the same thing) the image is rotated clockwise by 4 degrees.



## TUTORIAL GIMP

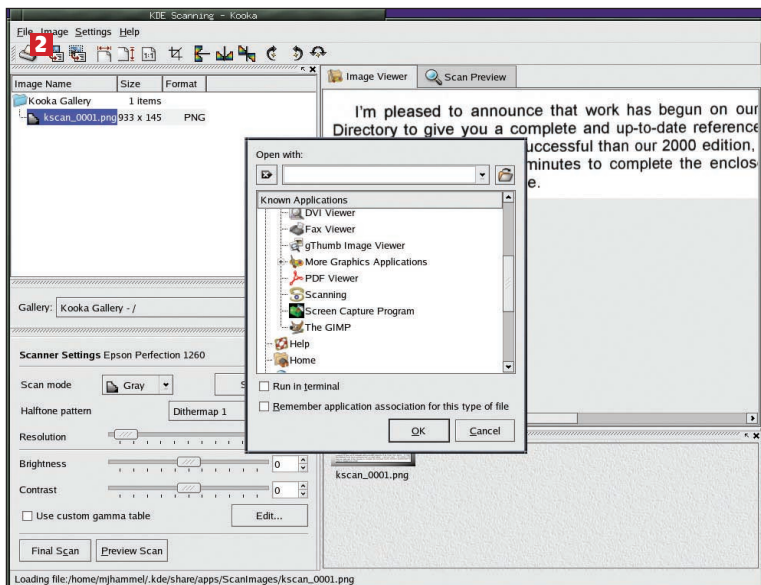
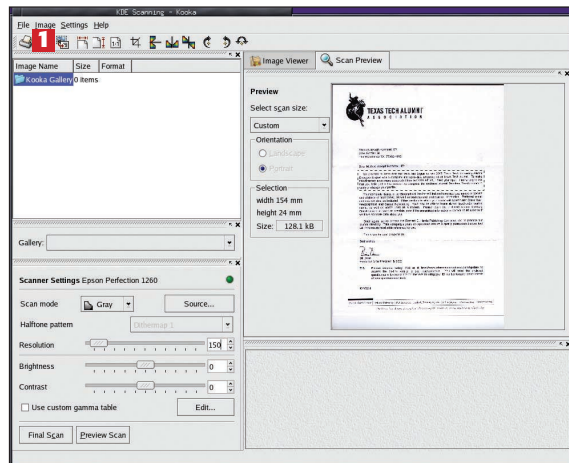
## « SCANNING FOR OPTICAL CHARACTER RECOGNITION (OCR)

Scanning doesn't have to be just about photos. You can scan documents and convert them to text as well. In this example a printed notice is scanned and a section is chosen for conversion to text using *Kooka*, a KDE tool that runs fine under GNOME (as

long as you have the KDE libraries installed). The packages you should have for this tutorial include the scanner drivers from *SANE*, *Kooka* (and KDE), the character recognition software called *gocr*, and *The GIMP*.

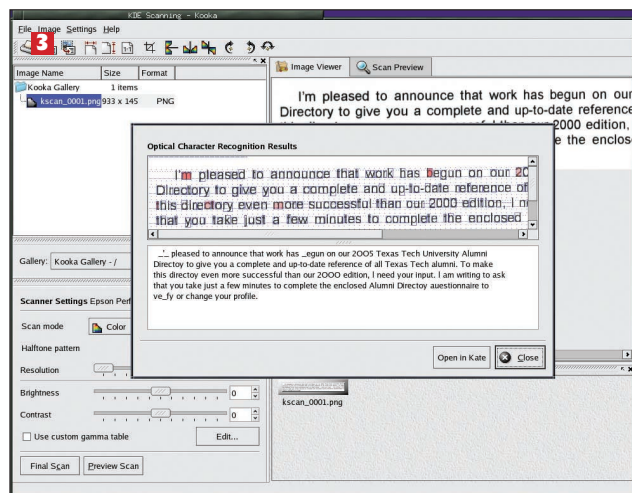
### Scan the document

**1** Place a document on the scanner and open *Kooka*. Get a preview of the document by clicking on 'Preview Scan' at the bottom of the window. In the 'Scan Preview' tab (right side of *Kooka* window) click and drag around a portion of the document. Set the resolution to 150 in the scan option on the lower left. You might want to adjust the brightness and contrast, but those changes won't be visible in the preview. When you're satisfied with the settings, click on 'Final Scan'. Save the image as a PNG. The image will be placed in a *Kooka*-specific directory.



### Select a region and edit it

**2** After scanning, the selected section will be available in the Image Viewer tab. If the image needs contrast adjustment, open it in *The GIMP* (Image>Open Image in Graphic Application) and make your adjustments. *Kooka* will open *The GIMP* with the image already named, allowing you to simply save your changes. Saving to a new name will add the image to the Gallery list. Make sure you select the updated version in *Kooka* before moving to the next step.



### TIPS AND TRICKS

#### Scanning sussed out

These basic scanning tips should help you in starting your pipeline:

- Scan larger than needed.
- Scan at 100 DPI for the Web (which is generally closer to 72DPI).
- Scan at 300 to 600 DPI for any image destined for a print. Scaling down is better than scaling up, but you'd rather not do either unless you are scanning halftones (newsprint). If working with halftones you might get Moire patterns - scaling down, rotating or just plain blurring can help reduce this effect.
- Save as TIFFs to work on scans - not in GIFs.
- Adjust contrast after scans.
- Set the white and black points.
- Try AutoLevels first, then experiment with manual changes. Sharpen after adjusting contrast. Use Unsharp Mask for quick work, Sharpen for more interactive work. Check out [www.scantips.com/](http://www.scantips.com/) for other helpful tips.

### Convert to text

**3** You can now convert the entire image or a portion of it to text. In this example a selection of the text is made and the 'Image>OCR Image' menu option chosen. A configuration options dialog is displayed. You can use the defaults to start. This generates a window that shows how the text was processed in graphical form with the processed text below. You can edit the text directly or retry the OCR process with modified settings. Results aren't perfect, but are better than retyping complete documents.



## FROM DRAWING TO VECTOR TO GIMP

A common use of scanners is to import hand drawings: this is often what artists call the start of the 'graphics pipeline'. Other parts of the pipeline include converting to vector format, editing in vector format, importing to raster format and colouring the

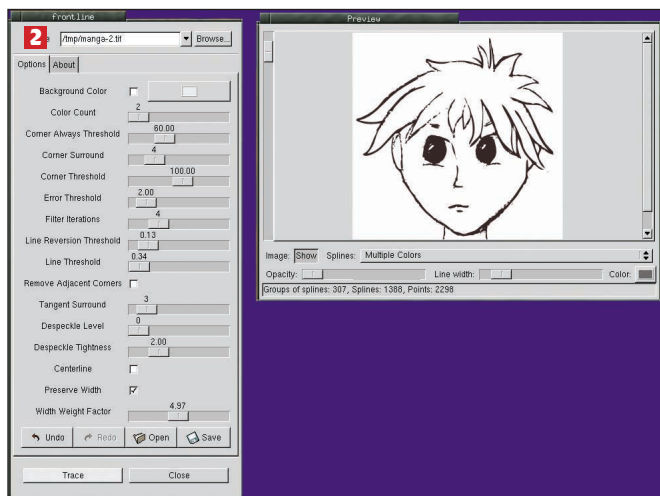
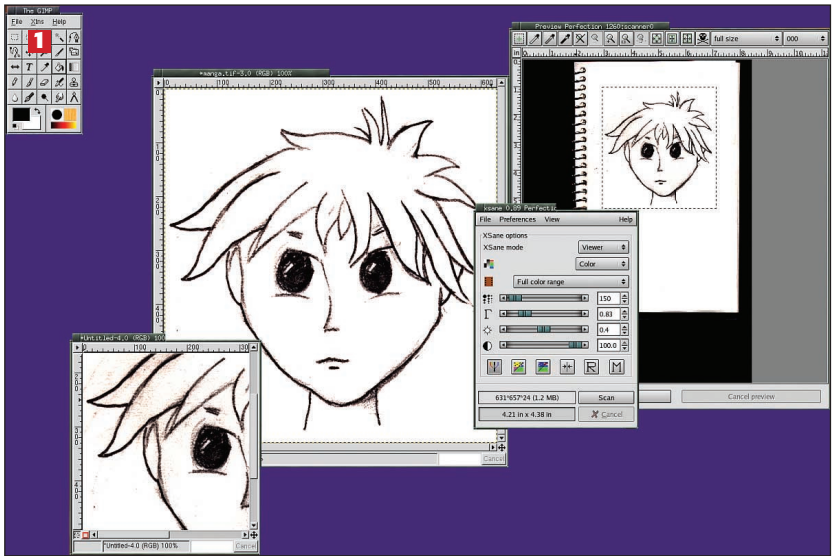
final work. Each of these steps can be covered by a freely available tool under Linux. This tutorial requires the following packages: the *SANE* scanner drivers, *GIMP*, *AutoTrace*, *Frontline* (the *AutoTrace* graphical interface), and *Sodipodi* or *Sketch*.

### Scan and adjust curves, save to tiff

**1** The pipeline starts with the scan. Select a clean drawing, one with high contrast between the paper and the drawing. The higher the contrast the better chance of producing a good vector version of the image. Scan the image and import it into *The GIMP*. Use either grayscale or color mode for the scan – do NOT use line art mode! This latter mode won't produce enough image information from the scan to make it viable as input to the next step in the pipeline. Using the *Curves* tool, adjust the image contrast as much as possible without sacrificing the quality of the drawing.

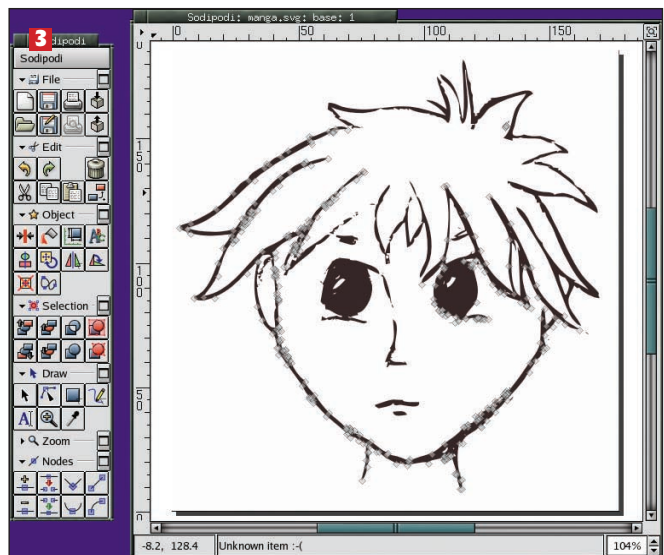
### Frontline/AutoTrace, save to SVG

**2** The next step in the pipeline is to transform the image to a vector format to make editing the lines easier. There are two ways to do this – either import that saved TIFF as a background image in your vector program (*Sketch* or *Sodipodi*) and trace it manually, or load the image in *Frontline* and trace it with *AutoTrace*. Use *Frontline* to adjust options to reduce the spline count. If you've adjusted the contrast well enough, then keeping the Color Count to 2 or 3 should help greatly.



### Sodipodi

**3** Next, the image is edited in vector format. Using a vector format allows you to clean up lines by dragging them to stretch or reposition them, something not possible with raster image tools like *The GIMP*. Importing the SVG file into *Sodipodi* we can edit lines as well as individual points on those lines. You can reduce the number of points on some lines (*AutoTrace* may produce a few extra beyond what are needed at a minimum) as well as breaking grouped items into smaller pieces. In this example part of the face and hair were grouped together. Once the items have been edited to your satisfaction, export to PNG or save as SVG and reopen in *GIMP* to colourise the image. With *Sodipodi* you can save groups of points as different images, allowing you to open them and copy them as new layers in a single *GIMP* image. Much of the colour can also be added directly in *Sodipodi*, though Fill colours might be easier in *The GIMP*.



### IMPORTANT NOTE

Get the tools you need

Some of the tools we'll be looking at this month may not be included in all stock Linux distributions. We have one simple answer for that: *apt-get*. If you aren't using *apt-get* for RPMs yet, it's time to start. Red Hat 9 includes this package management tool by default, and all the applications you need for these tutorials can be easily installed with just a few commands and a network connection. Once *apt-get* is installed, try

```
apt-cache search <string>
```

to look for a given package. Once you find the name, use

```
apt-get install <pkgname>
```

to install it. Dependencies will be resolved for you, which is what makes *apt-get* so wonderful. *apt-get* is available for both Debian (.deb) and Red Hat (.rpm) based distributions.