26

Comparing 2-D Electrophoresis Gels Across Internet Databases

Peter F. Lemkin and Gregory C. Thornwall

1. Introduction

In **refs.** *1–3* and in the previous edition of this book, we described a computer-assisted visual method, Flicker, for comparing two two-dimensional (2-D) protein gel images across the Internet, http://www.lecb.ncifcrf.gov/flicker/. This approach may be useful for comparing similar samples created in different laboratories to help putatively identify or suggest protein spot identification. Two-dimension gels and associated databases are increasingly appearing on the Internet *(4–17)* in World Wide Web (Web) servers *(18)* and through federated databases (DB) *(19)*. As this is an update on Flicker, we will not review the recent literature on 2-D gel databases or 2-D gel analysis systems here. **Table 1** lists some Web URL addresses for a number of 2-D protein gel databases that contain 2-D gel images with many identified proteins. This opens up the possibility of comparing one's own experimental 2-D gel image data with gel images of similar biological material from remote Internet databases in other laboratories. The image analysis method described here allows scientists to more easily collaborate and compare gel image data over the Web.

When two 2-D gels are to be compared, simple techniques may not suffice. There are a few ways to compare two images: (1) slide one gel (autoradiograph or stained gel) over the other while back lighted; or (2) build a 2-D gel quantitative computer database from both gels after scanning and quantitatively analyzing these gels using publicly available research *([20–27])* or commercial systems (recent systems are not reviewed here as the primary goal of this revision is the update on Flicker). These methods may be impractical for many investigators because the first case the physical gel or autoradiograph from another lab may not be locally available. In the latter case, the method may be excessive if only a single visual comparison is needed because of the costs (labor and equipment) of building a multigel database solely to answer the question of whether one spot is probably the same spot in the two gels.

This distributed Flicker gel comparison program runs on any World Wide Web-connected computer. It is a Java applet invoked from the user's Java-capable Web browser where it is then loaded from the NCI Flicker Web server. Alternatively, it may be downloaded to the user's computer and run locally (*see* **Subheading 2, step 26**).

From: *The Protein Protocols Handbook, 2nd Edition*
Edited by: J. M. Walker © Humana Press Inc., Totowa, NJ

Table 1
Partial List of World Wide Web 2-D Electrophoretic Gel Databases
Individual gel images with identified proteins
are available within these databases. An on-line version of this table
is available at **http://www.lecb.ncifcrf.gov/EP/table2Ddatabases.html**
The WORLD-2DPAGE Index to 2-D PAGE databases is available
at **http://expasy.cbr.nrc.ca/ch2d/2d-index.html.**

SWISS-2DPAGE
Liver, plasma, HepG2, HepG2SP, RBC, lymphoma, CSF, macrophage-CL, erythroleukemia-CL, platelet, yeast, E.coli, colorectal, kidney, muscle, macrophage-like-CL, Pancreatic Islets, Epididymus, dictyostelium
http://www.expasy.ch/
Argonne Protein Mapping Group
Mouse liver, human breast cell lines, pyrococcus
http://www.anl.gov/BIO/PMG/
Danish Centre for Human Genome Research
Human: primary keratinocytes, epithelial, hematopoietic, mesenchymal, hematopoietic, tumors, urothelium, amnion fluid, serum, urine, proteasomes, ribosomes, phosphorylations.
Mouse: epithelial, new born (ear, heart, liver, lung)
http://biobase.dk/cgi-bin/celis/
Joint Protein Structure Lab
Human Colorectal-CL, Placental lysosomes
http://www.ludwig.edu.au/jpsl/jpslhome.html
UCSF 2D PAGE
A375 melanoma cell line
http://rafael.ucsf.edu/2DPAGEhome.html
ECO2DBASE
E.coli
http://pcsf.brcf.med.umich.edu/eco2dbase/
or ftp://ncbi.nlm.nih.gov/repository/ECO2DBASE/
PROTEOME Inc
Yeast
http://www.proteome.com/
Yeast 2D gel DB, Bordeaux
Yeast
http://www.ibgc.u-bordeaux2.fr/YPM
HSC-2DPAGE, Heart Science Centre, Harefield Hospital
Human, rat and mouse heart
http://www.harefield.nthames.nhs.uk/
HEART-2DPAGE, German Heart Inst. Berlin
Human heart
http://www.chemie.fu-berlin.de/user/pleiss/
HP-2DPAGE, MDC, Berlin
Human heart
http://www.mdc-berlin.de/~emu/heart/
Immunobiology, Univ. Edinburgh
Embryonal stem cells
http://www.ed.ac.uk/~nh/2DPAGE.html

Table 1 (*continued*)

Large Scale Biology Corp
Rat, mouse, human liver, corn, wheat
http://www.lsbc.com/
Maize Genome Database, INRA
Maize
http://moulon.moulon.inra.fr/imgd/
Univ. Greifswald
Bacillus subtilis
http://pc13mi.biologie.uni-greifswald.de/
IPS/LECB, NCI/FCRDC
Phosphoprotein, prostate, phosphoprotein, breast cancer drug screen, FAS (plasma), Cd toxicity (urine), leukemia
http://www.lecb.ncifcrf.gov/ips-databases.html
Washington Univ. Inner Ear Protein Database
Human: Inner Ear
http://oto.wustl.edu/thc/innerear2d.htm
Protein Project of Cyanobacteria
Cyano2Dbase - Synechocystis sp. PCC6803
http://www.kazusa.or.jp/cyano/cyano2D/
2-D PAGE Aberdeen
Haemophilus influenzae & Neisseria meningitidis
http://www.abdn.ac.uk/~mmb023/2dhome.htm
Lab. de Biochimie et Tech. des Proteines, Bobigny
Human leukemia cell lines
http://www-smbh.univ-paris13.fr/lbtp/biochemistry/biochimie/bque.htm
Max-Planck-Institut f. Infektionsbiologie
Mycobacterium tuberculosis, vaccine strain M. bovis BCG,
http://www.mpiib-berlin.mpg.de/2D-PAGE/
ToothPrint DB
Dental tissue in rat
http://bioc111.otago.ac.nz:8001/tooth/home.htm
Siena 2D-PAGE
Chlamydia trachomatis L2, Caenorhabditis elegans, Human breast ductal carcinoma and Histologically normal tissue, Human amniotic fluid
http://www.bio-mol.unisi.it/2d/2d.html
PHCI-2DPAGE
Parasite host cell interaction, IFN-gamma induced HeLa cells
http://www.gram.au.dk/
PMMA-2DPAGE
Human colorectal carcinoma
http://www.pmma.pmfhk.cz/
BALF 2D_AGE
Mouse, Human broncho-alveolar lavage fluid
http://www.umh.ac.be/~biochim/BALF2D.html
Mito-pick
Human Mitochondria
http://www-dsv.cea.fr/thema/MitoPick/Mito2D.html

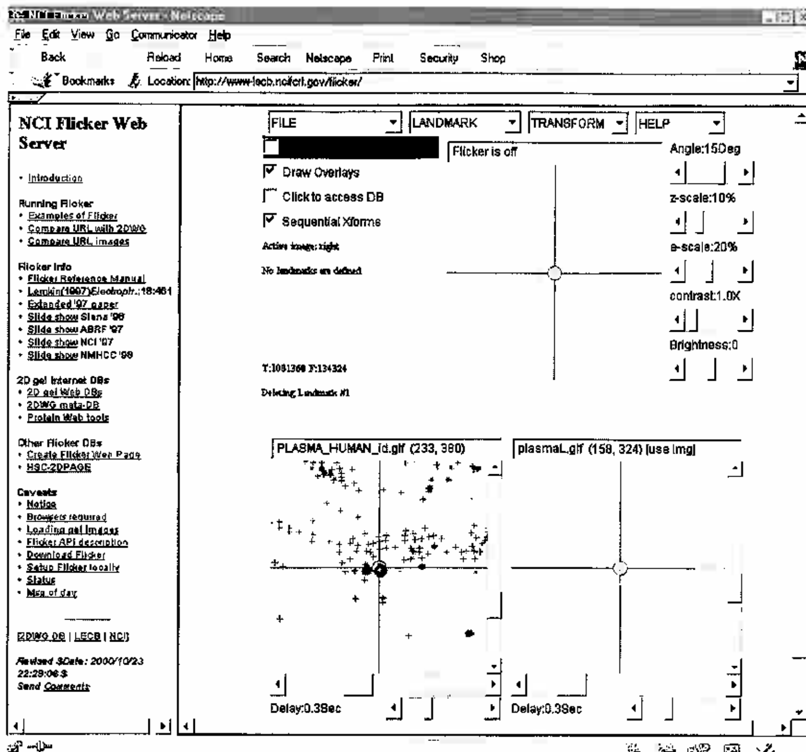

Fig. 1. Screen view of Netscape running the Flicker Java applet. Control-menus at the top invoke file operations, landmarking, image transforms. A set of scroll bars on the right determine various parameters used in the transforms. File menu options includes resetting images after a transform, aborting the current transform in progress, and help. Check-boxes on the left activate flickering and control display options. The "click to access DB" checkbox is available if one of the gel images is linked to a federated 2-D gel database. When this is set, clicking on a spot will get a pop-up browser report on that protein from the federated DB. A set of status lines appear below the check-boxes and indicate the state of operation. The flicker image is in the upper middle of the frame when it is enabled. The two labeled human blood plasma gel images are shown in the bottom scrollable windows that may be positioned to the region of interest. These windows also have associated flicker time-delays used when flickering. Image plasmaH (or PLASMA_HUMAN_id) is an IPG non-linear gradient gel from SWISS-2DPAGE in Geneva and plasmaL is a carrier-ampholyte linear gradient gel from the Merril Lab at NIMH. Transformed image results are shown in the same scrollable windows.

One gel image is read from any Internet 2-D gel database (e.g., SWISS-2DPAGE, etc.), the other may reside on the investigator's Web server where they were scanned or copied; or the two gel images may be from either Web server source. Portions of this paper were derived from **ref. *3***.

Flicker is also capable of interacting with federated 2-D gel databases to retrieve data on individual protein spots. Once gels are aligned in Flicker, you enable federated DB access; click on a spot in the gel belonging to the federated DB (*see* **Fig. 1**). This causes a Web page to pop up with information from the federated server describing that protein. We have set up a Web page to let users compare a gel from the Web (or their Web server if the gel image was copied there) to some of the SWISS-2DPAGE gels, http://www.lecb.ncifcrf.gov/flicker/swissProtIdFlkPair.html. First, select the resolution

you want to use (it defaults to 1×). Second, select the SWISS-2DPAGE gel image to use using the pull-down menu (e.g., select Human Plasma). Third, enter the URL of the gel on the Web that you wish to compare. As a convenience, we show a sample URL of a human plasma gel from a NIMH database to illustrate how you should enter the URL. You could use this gel to demonstrate how it works. Enter the URL, after which press "Go Flicker." This methodology could be replicated with any federated 2-D gel server and is discussed in more detail in the on-line Flicker API description available on the main Flicker Web page.

The Flicker program is written in Java, a general purpose, object-oriented programming language developed by Sun Microsystems *(28)* http://java.sun.com/. Java has become a standard for portable Internet Web applications. A Java "applet" is the name Sun gave for a mini-application that runs inside of a Web browser. When a user accesses a Web page containing an applet reference, it automatically loads the applet into the user's Java-capable.

Normally, users interact with the NCI Flicker Web server located at http://www.lecb.ncifcrf.gov/flicker, using the client/server paradigm shown in **Fig. 2**. The Flicker program may be thought of as a client that makes requests of a 2-D gel database Web server. Because it runs on the user's computer, Java now gives us the ability to perform real-time comparisons of local 2-D gel image data with gel images residing in various remote databases on the Internet. Then Flicker will load two images regardless of their respective source. Sources include: data from the Flicker Web server, other Web servers, or locally. The latter case applies when the applet file is copied to your local computer, you can create a HTML document to start Flicker on your local data (discussed in **Subheading 2, step 2b**).

Although the original images may be compared directly, this may be more easily achieved by first applying spatial warping or other image enhancement transforms. For gels with a lot of geometric distortion, it is useful to adjust one gel so that the geometry of the local regions being compared match that of the other gel. By local geometry, we mean the relative positions, distances, and angles of a set of spots in corresponding regions. One technique to do this is called spatial warping. When doing spatial warping, regions of interest are (1) landmarked with several corresponding points in each gel image in the region of interest, and (2) then one gel image is warped to the geometry of the other gel (*see* **Eqs. 1** and **2**). A landmark is a corresponding spot that is present in both gels. Spatial warping does not change the underlying grayscale values of the synthesized warped image to the extent that cause local structural objects to would appear and disappear and thus spot artifacts might be created. Instead, it samples pixels from the original image to be transformed and places them in the output image according to the geometry of the other input image.

Gels are then compared by flickering them rapidly by alternately displaying them in a third "flicker" display window. Using the mouse, the user may drag one gel image over the other to visually align corresponding spots by matching local morphology.

1.1. Image Flickering

The basic concept of using flickering as a dynamic visualization technique is simple. If two images may be perfectly aligned then one could simply align them by overlaying one over the other and shifting one image until they line up. However many images

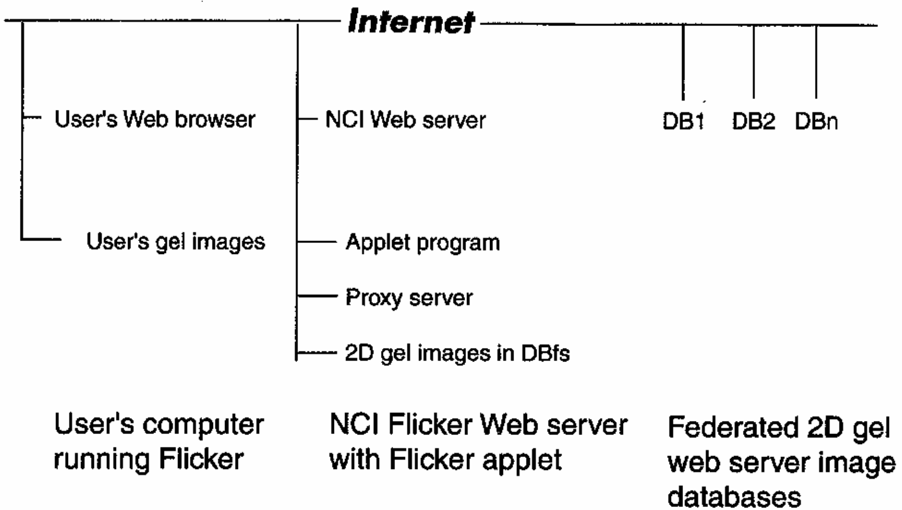

Fig. 2. This illustrates the client–server relationship between the user's Web browser with local 2-D gel images, the Web server that contains the Flicker program, and the 2-D gel image Web databases (DB) on other Web servers. Two gels to be compared may come from the Internet Web databases or from the user's local file system. The images may be from either the NCI Flicker 2-D gel image DB Web server DB_{fs} or from other 2-D gel image Web databases DB_1, DB_2, ..., DB_n. For example, DB_1 might be the SWISS-2DPAGE database, DB_2 might be the Danish Keratinocyte database, DB_3 might be the Cambridge heart database, DB_4 might be the Argonne breast cell line database, etc. For gel images on non-NCI servers, the request goes to the NCI server which contacts the other servers to get the image data and then sends it back to the user's browser.

such as 2-D polyacrylamide gel electrophoresis (PAGE) gels have "rubber-sheet" distortion (i.e., local translation, rotation, and magnification). This means there is more distortion in some parts of the image than in others. Although it is often impossible to align the two whole images at one time, they may be locally aligned piece-by-piece by matching the morphology of local regions.

If a spot and the surrounding region do match, then one has more confidence that the objects are the same. This putative visual identification is our definition of matching when doing a comparison. Full identification of protein spots requires further work such as cutting spots out of the gels and subjecting them to sequence analysis, amino acid composition analysis, mass spectrometry, testing them with monoclonal antibodies, or other methods.

1.2. Image Enhancement

It is well known that 2-D gels often suffer from local geometric distortions making perfect overlay impossible. Therefore, making the images locally morphologically similar while preserving their grayscale data may make them easier to compare. Even when the image sub regions are well aligned, it is still sometimes difficult to compare

images that are quite different. Enhancing the images using various image transforms before flickering may help. Some of these transforms involve spatial warping, which maps a local region of one image onto the geometry of the local region of another image while preserving its grayscale values. Other transforms include image sharpening and contrast enhancement. Image sharpening is performed using edge enhancement techniques such as adding a percentage of the gradient or Laplacian edge detection functions to the original grayscale image. The gradient and Laplacian have higher values at the edges of objects. Another useful operation is contrast enhancement that helps when comparing light or dark regions by adjusting the dynamic range of image data to the dynamic range of the computer display. In all cases, the transformed image replaces the image previously displayed. Other functionality is available in Flicker and is described in the Flicker on-line "help," **Subheading 3.** of this chapter, and in **refs. *1,3***.

1.3. Image Processing Transforms

As mentioned, there are a number of different image transforms that may be invoked from the control panel.

1.3.1. Affine Spatial Warping Transform

The spatial warping transforms require defining several corresponding landmarks in both gels. As we mentioned, one gel image can be morphologically transformed to the geometry of the other using the affine or other spatial warping transformations. These transforms map the selected image to the geometry of the other image. It does not interpolate the gray scale values of pixels — just their position in the transformed image. As described in **refs. *1,3***, this might be useful for comparing gels that have some minor distortion, comparing local regions, gels of different sizes or gels run under slightly different conditions. Flicker uses the affine transform as an inverse mapping as described in *(29)*. Let $(u_{xy},v_{xy}) = f(x, y)$ where (x, y) are in the output image, and (u, v) are in the input image. Then, in a raster sweep through the output image, pixels are copied from in the input image to the output image. The affine transformation is given in **Eqs. 1–2**:

$$u_{xy} = ax + by + c \tag{1}$$

$$v_{xy} = dx + ey + f \tag{2}$$

When the affine transform is invoked, Flicker solves the system of six linear equations for coefficients (a, b, c, d, e, f) using three corresponding landmarks in each gel.

1.3.2. Pseudo-3-D Transform

As described in **refs. *1,3***, the pseudo-3-D transform is a forward mapping that generates a pseudo 3-D relief image to enhance overlapping spots with smaller spots seen as side peaks. The gel size is width by height pixels. The gray value determines the amount of y shift scaled by a percentage z_{scale} (in the range of 0–50%). Pseudo-perspective is created by rotating the image to the right (left) by angle theta (in the range of –45 to +45 degrees). The transform is given in **Eqs. 3–5** for image of size width X height, shift in the horizontal dimension computed as d_x.

$$d_x = \text{width} \sin(\theta) \tag{3}$$

$$x' = [d_x (\text{height} - y)/\text{height}] + x \tag{4}$$

$$y' = y - z_{scale} * g(x, y) \tag{5}$$

where $g(x, y)$ is in the original input image and (x',y') is the corresponding position in the output mapped image. Pixels outside of the image are clipped to white. The pseudo-3-D transform is applied to both images so that one can flicker the transformed image.

1.3.3. Edge Sharpening

Edge sharpening may be useful for sharpening the edges of fuzzy spots. The sharpened image function $g'(x, y)$ is computed by adding a percentage of a 2-D edge function of the image to original image data $g(x, y)$ as shown in **Eq. 6**. The edge function increases at edges of objects in the original image and is computed on a pixel by pixel basis. Typical "edge" functions include the eight-neighbor gradient and Laplacian functions that are described in **ref. *1*** in more detail. The e_{scale} value (in the range of 0–50%) is used to scale the amount of edge detection value added.

$$g'(x, y) = (e_{scale} * \text{edge}(x, y) + (100 - e_{scale}) * g(x, y)/100 \tag{6}$$

2. Materials

The following lists all items necessary for carrying out the technique. Since it is a computer technique the materials consist of computer hardware, software and an Internet connection. We assume the user has some familiarity with computers and the World Wide Web.

1. A computer with a Java-compatible browser and an Internet connection is required. The actual computer could be a Windows-PC, Macintosh or Unix X-window system. The computer should have a minimum of 16 Mbytes of memory or more since intermediate images are held in memory when image transforms occur. If there is not enough memory, it will be unable to load the images, the transforms may crash the program or other problems may occur. Because a lot of computation is being performed, a computer with at least the power of an Intel 486/66 PC or better is suggested. A Pentium class machine with 32 Mb is more than adequate.

2a. To use Flicker directly from the NCI Flicker server, each time it is invoked it loads the applet into your Java-compatible Web browser. Browsers such as Netscape version 4.6 or later or Internet Explorer version 4.0 or later are required.

2b. Alternatively, if you decide to run Flicker locally, then you will need to download the Flicker Jar file and copy it to a directory with your GIF images. You will then need to edit an HTML file that invokes Flicker and also indicates that two GIF images you want to use. You may download the README, FlkJ2.jar, sample HTML, and other files from the http://www.lecb.ncifcrf.gov/FlkMirror Web site. There is a "tar" file at this site which bundles these files. You might also read the Flicker application programming interface (API) document located at http://www.lecb.ncifcrf.gov/flicker/flkParamList.html.

3. You will need a list of specific GIF image URLs from Internet 2-D gel image databases and/or copies of locally scanned gel images in GIF format. You can use the list of 2-D gel Web databases in **Table 1** as a starting point for finding gel GIF images you could download with Flicker. You might also investigate the ExPASy 2-D-hunt Web page which is a

Note that in the example, CODEBASE was set to the NCI Flicker server. If you run it on your server, simply use the URL where the Flicker FlkJ2.jar file and GIF images reside. If your are running this applet with your browser on a local disk, then omit the CODEBASE line and change the image1 and image2 VALUE's to the names of your GIF image files.

3.2. Graphical User Interface for Flickering

Figure 1 shows the screen of the Flicker applet as seen from a Netscape browser. Control pull-down menus at the top invoke file operations, landmarking, image transforms. Scroll bars on the side determine various parameters used in the transforms. The two images to be compared are loaded into the lower scrollable windows. A flicker window appears in the upper middle of the screen. Check-boxes on the left activate flickering and control display options. A group of status lines below the check-boxes indicate the state of operations.

Only part of an image is visible in a scrollable window. This subregion is determined by setting horizontal and vertical scroll bars. Another, preferred, method of navigating the scrollable images is to click on the point of interest while the CONTROL key is pressed. This will recenter the scrollable image around that point. This lets the user view any sub-region of the image at high resolution. These images may be navigated using either the scroll bars or by moving the mouse with the button pressed in the scrollable image window. Then, each image in the flicker window is centered at the point last indicated in the corresponding scrollable image window.

A flicker window is activated in the upper-middle of the screen when the "Flicker" check-box is selected. Images from the left and right scrollable images are alternatively displayed in the flicker window. The flicker delay for each image is determined by the adjusting the scroll bar below the corresponding scrollable image window. Various graphic overlays may be turned on and off using the "Overlays" check-box.

Clicking on either the left or right image selects it as the image to use in the next transform. However, clicking on the flicker image window indicates the transform should be applied to both left and right images.

3.3. Loading Images

When Flicker is running under a Web browser, the names of the images are fixed and are specified in the HTML (as shown earlier).

3.4. Flickering

3.4.1. Use of Flicker for Comparing Images

When flickering two images with the computer, one aligns putative corresponding subregions of the two rapidly alternating images. The flicker-display overlays the same space on the screen with the two images and is aligned by interactively moving one image relative to the other using the cursor in either or both of the lower images. Using the mouse, the user initially selects what they suspect is the same prominent spot or object in similar morphologic regions in the two gel images. The images are then centered in the flicker window at these spots. When these two local regions come into alignment, they appear to pulse and the images fuse together. At this point, differences

search engine for 2-D gel electrophoresis Web sites at http://expasy.cbr.nrc.ca/ch2d/2DHunt/.

4. If the investigator will be using his or her own scanned gels, they will need to either have the gel scanner on the machine where Flicker will run or arrange to transfer the image files to that machine. In addition, the gel images may need to be converted to GIF images required by Flicker software (e.g., many scanners generate TIFF formatted images). Image format conversion software may be part of your scanner software. If not, there are a number of image file format converters available as part of various desktop publishing packages. Also some converter software is available free from the Internet (use a Web search engine such as Alta Vista to help find it).

3. Method

We now describe the operation of the Flicker applet from the point of view of the user. You first start up Flicker. This may include the specification of particular images from the NCI Flicker server. Otherwise, you need to specify the gel images to load once Flicker is running. Then you simply flicker the gel images or use image enhancement transforms first and then flicker them.

3.1. Using Flicker with the NCI Gel Image Proxy Server

3.1.1. Using Flicker Under a Web browser — No Installation of Any Software is Needed!

Assuming you have a working Web browser on your Internet-connected computer, there is nothing to do since Flicker is automatically downloaded into your browser every time you invoke its URL on the NCI Web server.

Using the NCI proxy server, you can use the following Web page to request two image URLs that you enter (http://www.lecb.ncifcrf.gov/flicker/urlFlkPair.html) and it will get the images and start Flicker in your Web browser.

Alternatively, you can compare a Web image URL against one of the images in the 2DWG gel image database *(2)* (http://www.lecb.ncifcrf.gov/2dwgDB). Search the 2DWG. Select ONE gel image you want to compare with one from the Web. Then scroll down to the bottom of the search results page and type in the image URL on the Web. Finally, press the "Go Flicker" button.

For those who are interested in the details on how Flicker is invoked, you can see some of the HTML (HyperText Markup Language) examples of how to start Flicker with pre-specified images from HTML in files linked from the Flicker home page at http://www.lecb.ncifcrf.gov/flicker/# Flicker-examples. There are additional examples given in the Flicker API. These use the HTML <APPLET> and </APPLET> tags. The following is an example of the <APPLET> HTML code required to start Flicker with two images.

```
<APPLET
CODEBASE=http://www.lecb.ncifcrf.gov/flicker/
CODE=FlkJ2.class ARCHIVE=FlkJ2.jar
WIDTH=650 HEIGHT=700 ALIGN=absmiddle
ALT="A java-enabled browser is needed to view Flicker applet.">
<PARAM NAME=image1 VALUE=plasmaH.gif>
<PARAM NAME=image2 VALUE=PlasmaL.gif>
</APPLET>
```

are more apparent and it is fairly easy to see which spots or objects correspond, which are different, and how they differ. We have found that the user should be positioned fairly close to the flicker window on the screen to optimize this image-fusion effect.

3.4.2. Selecting the Proper Time Delays When Flickering

The proper flicker delays, or time each image is displayed on the screen, is critical for the optimal visual integration of image differences. We have also found that optimal flicker rates are dependent on a wide variety of factors including: amount of distortion, similarity of corresponding subregions, complexity and contrast of each image, individual viewer differences, phosphor decay-time of the display, ambient light, distance from the display, and so forth. We have found the process of flickering images is easier for some people than for others.

When comparing a light spot in one gel with the putative paired darker spot in the other gel one may want to linger longer on the lighter spot to make a more positive identification. Because of this, we give the user the ability to set the display times independently for the two images (typically in the range of 0.01 s to 1.0 s with a default of 0.20 s) using separate "Delay" scroll bars located under each image. If the regions are complex and have a lot of variation, longer display times may be useful for both images. Differential flicker delays with one longer than the other are also useful for comparing light and dark sample gels.

3.5. Image Processing Methods

As mentioned, there are a number of different image transforms that may be invoked from the menus. These are useful for changing the geometry, sharpness, or contrast making it easier to compare potentially corresponding regions. As we go through the transforms we will indicate how they may be used. Some affect one image while some affect both. Flickering is deactivated during image transforms to use most computational power for doing the transforms.

The TRANSFORM menu has a number of selections that include warping, grayscale transforms and contrast functions. The two warp method selections: "Affine Warp and "Poly Warp" are performed on only one image (the last one selected by clicking on an image). Unlike the warp transforms, the grayscale transforms are performed on both images. These include: "Pseudo 3-D," "SharpenGradient," "SharpenLaplacian," "Gradient," "Laplacian," "Average." The contrast functions are "Complement" and "ContrastEnhance."

3.5.1. Landmarks: Trial and Active

The affine transform requires three active landmarks to be defined before it can be invoked. A trial landmark is defined by clicking on an object's center anywhere in a scrollable image window. This landmark would generally be placed on a spot. Clicking on a spot with or without the CONTROL key pressed still defines it as a trial landmark. After defining the trial landmark in both the left and right windows, selecting the "Add Landmark" option in the Landmark menu defines them as the next active landmark pair and identifies them with a red letter label in the two scrollable image windows. Selecting the "Delete Landmark" option deletes the last active landmark pair defined.

A

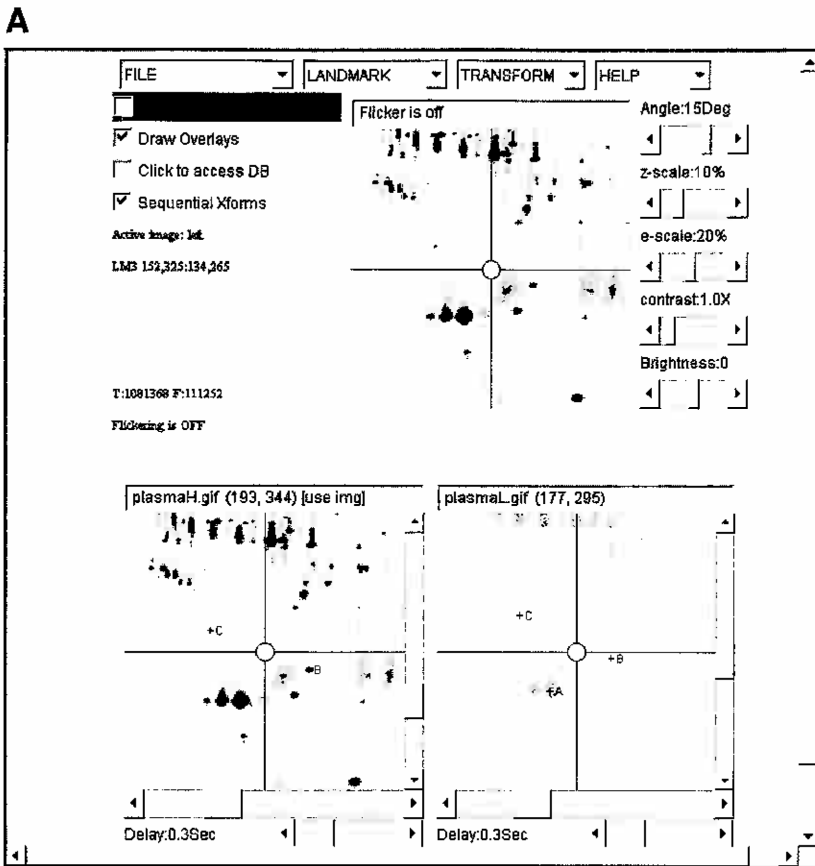

Fig. 3. Screen views of affine transform of human plasma gel image. The transform warps the geometry of a local region defined by the three landmarks so it more closely resembles the geometry of the corresponding local region in the other gel. **(A)** Scrollable image windows with three "active" landmarks defined in both gel images that were selected interactively in preparation for doing the affine image transform. Corresponding landmark spots are picked so as to be

3.5.2. Affine Transform

The two warping transforms, affine (*see* **Eqs. 1** and **2**) and polynomial, require three and six landmarks respectively. Attempting to run the transform with insufficient landmarks will cause Flicker to notify you that additional landmarks are required. The image to be transformed is the one last selected. You must select either the left or right image. **Figure 3** shows the landmarks the user defined in the two gels before the affine transform. **Figure 3b** shows the affine transform done on the right gel image.

3.5.3. Pseudo-3-D Transform

As described in (1) and as shown in **Eqs. 3–5**, the pseudo-3-D transform generates a pseudo-3-D relief image to enhance overlapping spots with smaller spots seen as side peaks. The gray value determines the amount of "*y*" shift scaled by a percentage (set by scroll bar z_{scale} (in the range of 0–50%). Pseudo perspective is created by shifting the image to the right or left by setting by scroll bar "angle" degrees (in the range of –45 to +45 degrees).

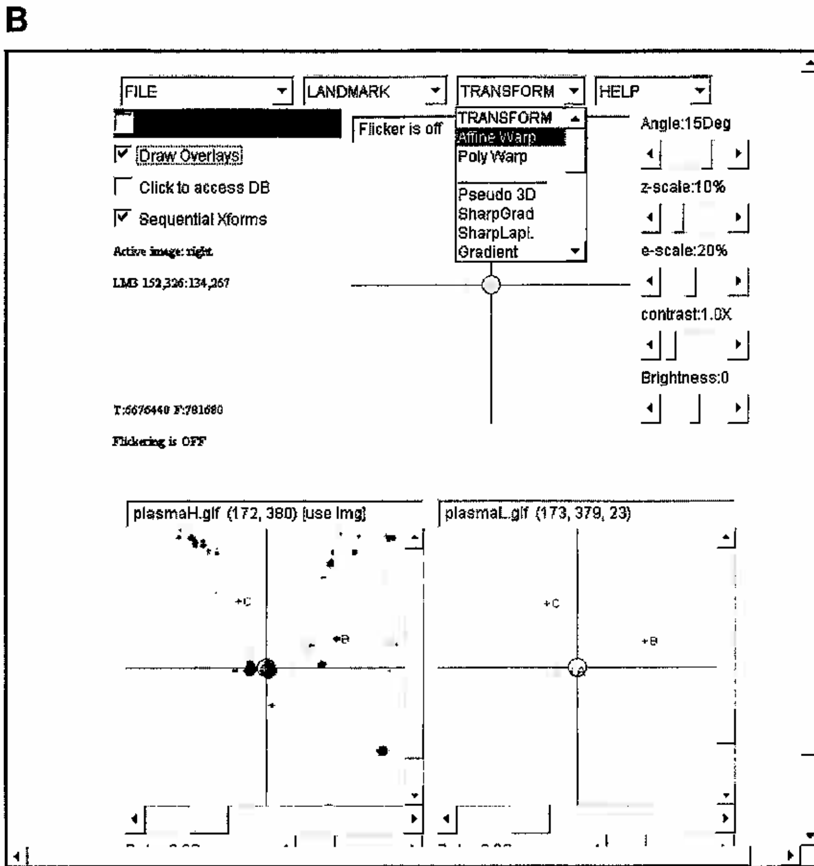

Fig. 3. (*continued from opposite page*) unambiguously defined in both gel images. **(B)** Scrollable image windows after the affine warp transform of the right (plasmaL, non-IPG) image to the geometry of the left (plasmaH, IPG gel) image. The plasmaH image is the same gel as PLASMA_HUMAN_id but without the graphic overlays.

Negative angles shift it to the right and positive angles to the left. The image to be transformed is the one last selected. If neither was selected (i.e., you clicked on the flicker-window), then both images are transformed. **Figure 4** shows the results of applying the pseudo-3-D transform to both images.

3.5.4. Edge Sharpening

Edge sharpening may be useful for improving the visibility of the edges of fuzzy spots. You can select either a Gradient or Laplacian edge sharpening function using the "SharpenGradient" or "SharpenLaplacian" operation in the "TRANSFORM" menu where the image to be transformed is the one last selected. The Laplacian filter generates a "softer" edge than the Gradient. You can set the scroll bar e_{scale} value (in the range of 0–50%) to scale the amount of edge detection value added. The image to be transformed is the one last selected. If neither was selected (i.e. you clicked on the flicker-window), then both images are transformed. **Figure 5** shows the results of applying the image-sharpening Laplacian transform to both images.

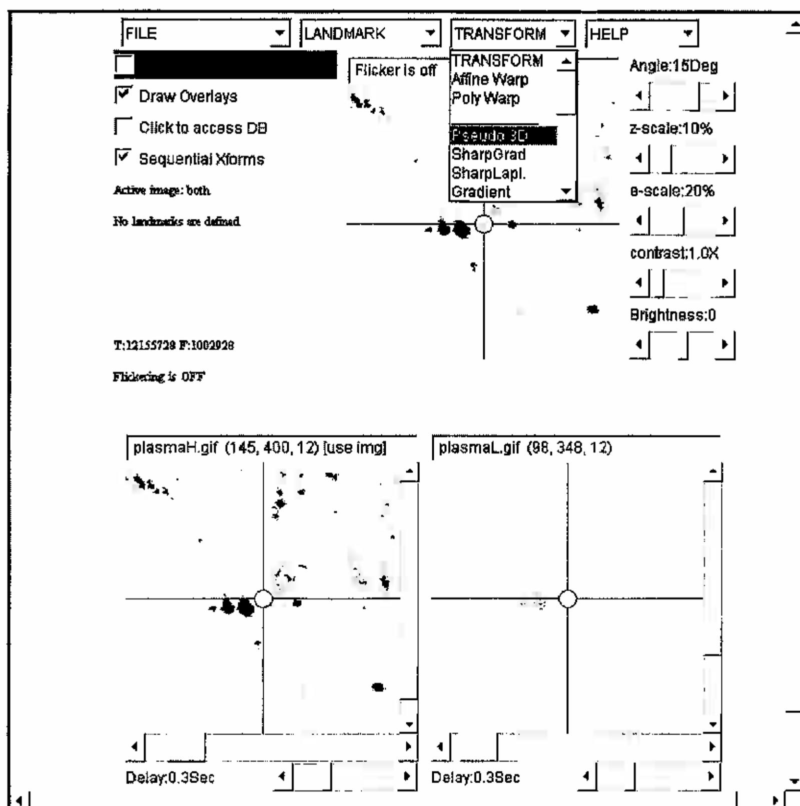

Fig. 4. Human plasma gel images after the pseudo-3-D transform was applied to both gel images. The parameter settings for angle and z_{scale} were 15 degrees and 20% respectively.

3.5.5. Other Image Transforms

There are a number of other image transforms which can be invoked. Like the edge sharpening transforms, the image to be transformed is the one last selected and if neither was selected (i.e., you clicked on the flicker-window), then both images are transformed.

4. Notes

1. *On installing Flicker on your computer*
   There are several advantages of running Flicker directly from the NCI Flicker server. Software updates are completely invisible to users since they don't have to waste time or space installing them on their computers. The technique uses existing low cost Web browser technology that requires little user effort. In addition, it saves time over the alternative ways that scientists might use to compare 2-D gels and other data. However, if you are far from our server, want to run Flicker behind a firewall, or want to use it exclusively on your own data, then it may be advantageous to download Flicker and install it on your computer (*see* **Subheading 2, step 2b**).

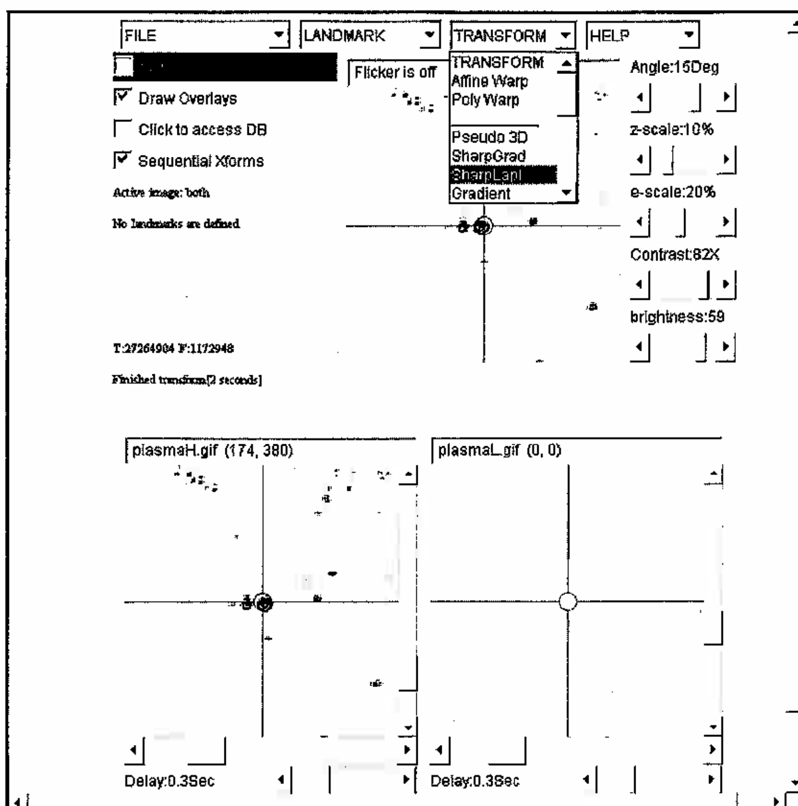

Fig. 5. Human plasma gel images after the SharpenLaplacian transform was applied to both gel images. The parameter settings for e_{scale} was 40%.

2. *On loading images*
   There are several problems with using Java with current Web browsers because of restrictions due to applet security restrictions. Because of fears of security breaches, Netscape and other Web browser providers have disabled Java applets running on their browsers from reading or writing local files. They also restrict access to Web URLs other than the host computer where the Java applet originated (i.e., in this case, the NCI Flicker Web server). Unfortunately, this prevents the Flicker applet from loading your local image files or other 2-D gel image databases not the Flicker Web server. It thus prevents you from comparing data from different sources.
   However, there are two ways to get around this security problem: (1) Use a Web proxy server (such as the NCI Flicker proxy server) to get the image and then pass it back to your browser as if the data came from the Flicker server. (2) Run Flicker as an applet in your Web browser using a local Web page that points to the data on your local computer or your Web server.
   Another restriction is that Flicker itself currently only handles GIF and JPEG image formats. Images in other formats such as TIFF currently need to be converted to GIF format. We do this on our NCI Flicker proxy server. If you are running a local copy, you

could convert your images to GIF using programs such as PhotoShop, and so forth. Because we are doing image pixel processing with the Flicker program, it requires more memory for intermediate images than programs that only manipulate text and so may require a more powerful CPU with more memory than some users currently have available.

3. *On flickering*
   There are some disadvantages in comparing gels this way. It is only good for doing a rough comparison and there is currently no simple way available to do quantitative comparison (as can be done with existing 2-D gel computer database systems *[20–27]*) — although we are working on the latter and will announce it on the Flicker Web site. One should keep these limitations in mind when using the technique.
   The intent of applying image transforms is to make it easier to compare regions having similar local morphologies but with some different objects within these regions. Image warping prior to flickering is intended to spatially warp and rescale one image to the "shape" of the other image so that we can compare them at the same scale. This should help make flickering of some local regions on quite different gels somewhat easier. Of the two warping transforms, affine and polynomial, the latter method handles nonlinearities better. For those cases where the gels are similar, the user which may be able to get away with using the simpler (affine) transform.
   In cases where there is a major difference in the darkness or lightness of gels, or where one gel has a dark spot and the other a very faint corresponding spot, it may be difficult to visualize the light spot. By differentially setting the flicker display-time delays, the user can concentrate on the light spot using the brief flash of the dark spot to indicate where they should look for the light spot. We have found differential-flicker to be very helpful for deciding difficult cases. Changing image brightness and contrast also is useful when flickering and the Flicker program has provision for interactively changing these parameters as well.
4. *On image transforms*
   Other transforms including image sharpening may be useful in cases where spots are very fuzzy, as might be the case when comparing Southern blots. When two corresponding local regions of the two images are radically different so the local morphologies are not even slightly similar (e.g., when high molecular weight regions of gels that are run differently as: IPG vs non-IPG, gradient vs non-gradient sodium dodecyl sulfate [SDS]), then even using these transforms may not help that much.
5. *The current status of Flicker*
   Of the features and operations we have mentioned, some are not fully functional and we are working to resolve this. The current state of Flicker is documented in the Flicker Reference Manual http://www.lecb.ncifcrf.gov/flicker/flkInfo.html. A future release of Flicker will contain quantification facility and be able to be run as a stand-alone Java application.

5. References

1. Lemkin, P. F. and Thornwall, G. (1999) Flicker image comparison of 2-D gel images for putative protein identification using the 2DWG meta-database. *Mol. Biotechnol.* **12,** 159–172.
2. Lemkin, P. F. (1997) 2DWG meta-database of 2D electrophoretic gel images on the Internet. *Electrophoresis* **18,** 2759–2773.
3. Lemkin, P. F. (1997) Comparing two-dimensional electrophoretic gels across the Internet. *Electrophoresis* **18,** 461–470.

4. Appel, R. D., Sanchez, J.-C., Bairoch, A., Golaz, O., Miu, M., Vargas, J. R., and Hochstrasser, D. F. (1993) SWISS-2DPAGE: a database of two-dimensional gel electrophoresis images. *Electrophoresis* **14,** 1232–1238.
5. Appel, R. D., Sanchez, J.-C., Bairoch, A., Golaz, O., Rivier, F., Pasquali, C., et al. (1994) SWISS-2DPAGE database of two-dimensional polyacrylamide gel electrophoresis. *Nucleic Acids Res.* **22,** 3581–3582.
6. Sanchez, J.-C., Appel, R. D., Golaz, O., Pasquali, C., Rivier, F., Bairoch, A., and Hochstrasser, D. F. (1995) Inside SWISS-2DPAGE database. *Electrophoresis* **16,** 1131–1151.
7. Jungblut, P., Thiede, B., Zinny-Arundl, U., Muller, E.-C., Scheler, C., Whittmann-Liebold, B., and Otto, A., (1996) Resolution power of two-dimensional electrophoresis and identification of proteins from gels. *Electrophoresis* **17,** 839–846.
8. Wilkins, M. R., Sanchez, J.-C., Williams, K. L., and Hochstrasser, D. F. (1996) Current challenges and future applications for protein maps and post translational vector maps in proteome projects. *Electrophoresis* **17,** 830–838.
9. Celis, J. E. (ed.). (1992) Special issue: Two-dimensional electrophoresis protein databases. *Electrophoresis* **13,** 891–1062.
10. Celis, J. E. (ed.). (1993) Special issue: Two-dimensional electrophoresis protein databases. *Electrophoresis* **14,** 1089–1240.
11. Celis, J. E. (ed.), (1994) Special issue: electrophoresis in cancer research. *Electrophoresis* **15,** 305–556.
12. Celis, J. E. (ed.), (1995) Special issue: two-dimensional electrophoresis protein databases. *Electrophoresis* **16,** 2175–2264.
13. Bjellqvist, B., Hughes, G. J., Pasquali, C., Paquet, N., Ravier, F., Sanchez, J.-C., et al. (1993) *Electrophoresis* **14,** 1023–1031.
14. Pallini, V., Bini, L., and Hochstrasser, D. (1994) Proceedings: 2D Electrophoresis: From Protein Maps to Genomes. University of Siena, Italy, Sept 5–7.
15. Pallini, V., Bini, L., and Hochstrasser, D. (1996) Proceedings: 2nd Siena 2D Electrophoresis Meeting: From Protein Maps to Genomes. University of Siena, Italy, Sept 16–18.
16. Pallini, V., Bini, L., and Hochstrasser, D. (1996) Proceedings: 3rd Siena 2D Electrophoresis Meeting: From Protein Maps to Genomes. University of Siena, Italy, Aug 31–Sept 3.
17. Pallini, V., Bini, L., and Hochstrasser, D. (2000) Proceedings: 4th Siena 2D Electrophoresis Meeting: From Protein Maps to Genomes. University of Siena, Italy, Sept 4–7.
18. Berners-Lee, T. J., Cailliau, R., Luotonen, A., Henrick, F., and Secret, A. (1994) World Wide Web. *Comm. Assoc. Comp. Mach.* **37,** 76–82.
19. Appel, R. D., Bairoch, A., Sanchez, J.-C., Vargas, J. R., Golaz, O., Pasquali, C., and Hochstrasser, D. F. (1996) Federated two-dimensional electrophoresis database: a simple means of publishing two-dimensional electrophoresis data. *Electrophoresis* **17,** 540–546.
20. Taylor, J., Anderson, N. L. Scandora, A. E., Willard, K. E., and Anderson, N. (1982) Design and implementation of a prototype human protein index. *Clin.Chem.* **28,** 861–866.
21. Lipkin, L. E. and Lemkin, P. F. (1980) Database techniques for multiple PAGE (2D gel) analysis. *Clin.Chem.* **26,** 1403–1413.
22. Lemkin, P. F. and Lester, E. P. (1989) Database and search techniques for 2D gel protein data: a comparison of paradigms for exploratory data analysis and prospects for biological modeling. *Electrophoresis* **10,** 122–140.
23. Appel, R. D., Hochstrasser, D. F., Funk, M., Vargas, J. R., Pellegrini, C., Muller, A. F., and Scherrer, J.-R. (1991) The MELANIE project from biopsy to automatic protein map interpretation by computer. *Electrophoresis* **12,** 722–735.

24. Olson, A. D. and Miller, M. J. (1988) Elsie 4: quantitative computer analysis of sets of two dimensional gel electrophoretograms. *Analyt. Biochem.* **169,** 49–70.
25. Garrels, J. I., Farrar, J. T., and Burwell, IV, C. B. (1984) Analyt. in Two-Dimensional Gel Electrophoresis of Proteins (Celis, J. E. and Bravo, R., eds.), Academic Press, New York, pp. 37–91.
26. Vincens, P. and Rabilloud, T. (1986) in Recent Progress in Two-Dimensional Electrophoresis' (Galteau, M. M. and Siest, G., eds.), University Press of Nancy, France, pp. 121–130.
27. Pleisner, K.-P., Hoffmann, F., Kriegel, K., Wenk, C., and Wegner, S. (1999) New algorithmic approaches to protein spot detection and pattern matching in two-dimensional electrophoresis gel databases. *Electrophoresis* **20,** 755–765.
28. Arnold, K. and Gosling, J. (1996) The Java Programming Language. Addison-Wesley, Reading, MA, pp. 1–704.
29. Wolberg, G. (1990) Digital Image Warping. IEEE Computer Press Monograph, Los Alamitos, CA, 1–318.