PROCESS TOMOGRAPHY Ltd.

ELECTRICAL CAPACITANCE TOMOGRAPHY SYSTEM

TYPE PTL300E

OPERATING MANUAL

Issue 7

September 2010

VOLUME 3. SUPPLEMENTARY SOFTWARE

Recal section updated for v 1.14

ECT32v2 Software Version 2.38
Firmware version 1.47
System hardware .DLL DAM200E_2_27

PROCESS TOMOGRAPHY LTD
86 Water Lane, Wilmslow, Cheshire. SK9 5BB United Kingdom.
Phone/Fax 01625-549021
(From outside UK +44-1625-549021)
email: enquiries@tomography.com Web site: www.tomography.com

Registered in England No. 2908507. Registered Office 15, Croft Road, Wilmslow, Cheshire. SK9 6JJ United Kingdom.

VOLUME 3

SUPPLEMENTARY SOFTWARE

SECTION 9 (continued)**APPENDICES 8 to 13**

APPENDIX 8

ADVANCED RECALIBRATION SOFTWARE RECAL

RECAL is an offline software untility for use with **PTL ECT** systems. The **RECAL** software modifies an **existing sensor calibration file** which contains **at least one set of valid calibration data** at the **low permittivity calibration point**. It can be used, for example, to generate a valid calibration file by carrying out a set of capacitance measurements with an empty sensor only. It can also be used to modify the measurement range of an ECT sensor. The **RECAL** software is supplied as a stand-alone executable program file.

CONTENTS

A8.1. INTRODUCTION

In principle, it should be possible to operate an ECT sensor in the same way as any other passive physical transducer. However, many ECT sensors use electrodes located outside a dielectric tube and for this type of sensor, the relationship between the capacitances measured between pairs of electrodes and the permittivity of the sensor contents can be complex. This is particularly the case for the capacitances between adjacent electrodes. To overcome this problem, two sets of calibration measurements are normally made at the nominal limits of the range of operation of the sensor before it is used to carry out data capture. The information from these calibration measurements is then used to produce an approximately linear relationship between the measured capacitances and the permittivity of the sensor contents, as described in equation 1 below.

The standard method for calibrating an ECT sensor for use with a mixture of 2 dielectric materials is to measure two sets of inter-electrode capacitances for the capacitance sensor, one set with the sensor filled with the lower permittivity material and a second set with the sensor filled with the higher permittivity material. These capacitances are measured by applying a fixed alternating voltage to each electrode in turn (the source electrode) and measuring the currents which flow into the remaining (detector) electrodes (which are held at virtual earth potential), using current detector circuits. The outputs from these current detectors are DC voltages, which are proportional to the inter-electrode capacitances..

An additional set of apparent capacitances (one for each detector electrode) is measured for the case where there is no source electrode, that is, in the absence of any excitation voltage. This third set of (spurious) capacitances results from capacitive coupling of the switching signals inside the capacitance measurement circuitry and are known as "charge-injection" capacitances. They must be subtracted from each of the first 2 sets of measured capacitances to yield the correct values of measured capacitance for each electrode-pair.

The capacitance measurement circuitry operates as a variable-gain DC bridge circuit. The output voltage from each current detector circuit is cancelled (or balanced) by applying an offset voltage from a digital voltage generator circuit. The output voltage from this circuit is proportional to a 10-bit (0-1023) integer count (M1) generated by the control computer. The bridge output passes through a digitally-controlled attenuator whose gain is set in the range 0-1 by a second 10-bit count (M2), before passing to a 12-bit analogue-to-digital (ADC) circuit which gives an output count M3 in the range 0-4095, which is related to the measured value of capacitance.

Each capacitance measurement is therefore represented by a set of 3 integer counts M1, M2 and M3. This arrangement has the advantage that integer constants and integer arithmetic can be used to process the measured data at high-speed. If the values of M1, M2 and M3 are known for a specific capacitance measurement, together with the equivalent charge-injection values, then the absolute capacitances can be calculated from the capacitance measurement circuit constants.

In practice, the capacitances are stored and displayed by the system software as normalised values, where all of the normalised inter-electrode capacitances have the value zero when the sensor is filled with the lower permittivity material and 1 when the sensor is filled with the higher permittivity material.

The capacitance normalisation is carried out using the equation:

$$Cn=(Cm-Cl)/(Ch-Cl) \qquad (A8.1)$$

where:

Cn is the normalised capacitance
Cm is the absolute value of the measured capacitance in fF
Cl is the absolute capacitance in fF measured at the lower permittivity calibration point.
Ch is the absolute capacitance in fF measured at the higher permittivity calibration point.

The permittivities of the image pixels are similarly normalised so that they have zero values when the sensor is filled with the lower permittivity material and 1 when filled with the higher permittivity materials. This method of operating the ECT system means that the calculated image pixel permittivities should always be accurate at the lower and upper calibration points, although there will not be a completely linear relationship between permittivty and normalised capacitance between these points.

A8.2 ECT32 PROGRAM RECALIBRATION OPTIONS

The standard method for generating an ECT calibration file involves measuring the ECT sensor capacitances with the sensor first filled with the low permittivity material and then filled with the high permittivity material. However, in many cases, it would be helpful to be able to modify the calibration data following initial calibration without carrying out a full recalibration. A limited range of recalibration options is provided within the **ECT32** software and these are described in the next section. A more comprehensive range of options is available using the **Recal** software and these are described in detail in section A8.3.

A8.2.1 ADJUSTING THE ECT SYSTEM CALIBRATION

The standard **ECT32 operating software** provides the user with a number of calibration options via the **calibration window** shown in figure A8.1 below:

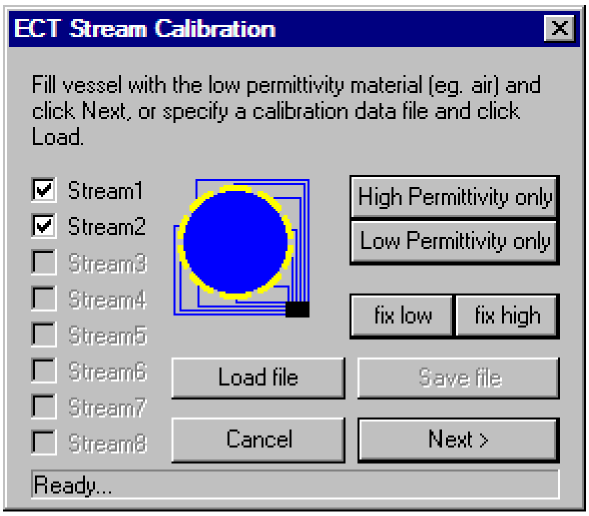


Figure A.8.1. ECT32 Calibration window

Apart from carrying out a standard sensor calibration, the **calibration window** allows the sensor calibration file to be modified in several ways as described below by the use of the appropriate buttons in the calibration window:

Low Permittivity only. This operation is useful for correcting an existing calibration file as it allows the sensor to be recalibrated at the low permittivity level only. It can be used if the sensor is known to contain only the lower permittivity material. A full low-level calibration is carried out, producing new values for all of the counts M1, M2 and M3 at the low permittivity calibration point and also for the equivalent low-permittivity charge-injection capacitances. As the M1 parameter is also used in the high-permittivity calibration data, new values of M2 and M3 are then calculated for the high permittivity calibration parameters to ensure that the calibration file remains valid at the high permittivity calibration point.

High Permittivity only. This operation is similar to the previous button but is intended for use when the sensor contains the higher permittivity material. Again a full high level recalibration is carried out, generating new high-permittivity calibration values for the M2 and M3 counts (the M1 count is not changed by the high-level recalibration). A new set of high-permittivity chage injection capacitances are also measured. This operation has no effect on the low-permittivity calibration capacitances, because the low permittivity calibration is carried out at the circuit balance point, where the gain setting has no effect.

One particularly useful function of the **High Permittivity only** button is to generate a calibration file when a long time period elapses between the 2 calibration measurements. An example would be calibrating an ECT sensor installed on an experimental plant where a gap of hours or days occurs between filling and emptying the sensor. In this case the low permittivity calibration is carried out as normal following the standard procedure. However, the high permittivity calibration is initially carried out while the sensor still contains the low permittivity fluid and the calibration file is saved as an interim file. When the sensor contains the high permittivity fluid, the sensor calibration is concluded by loading the saved interim calibration file. The **High permittivity only button** is then used to complete the calibration process and the the completed calibration file is then re-saved.

There are two further buttons in the **ECT32 calibration window** whose functions are as follows:

Fix low: This carries out a minor recalibration of the sensor when it is filled with the lower permittivity material. However, the M1 and M2 values remain unchanged and only the **low-permittivity M3 ADC count** is adjusted. This has the advantage that the high level calibration counts remain unaffected by the low-level modifications and the high level capacitances therefore remain unchanged. The "**Fix low**" option is guaranteed not to disturb the high permittivity calibration point (unlike the "**low permittivity only**" option which might cause a slight shift in the high point) **The Fix low** button should be used to correct any minor drift in the low permittivity calibration point. This option also measures a new set of low-permittivity charge injection capacitances.

Fix high: This performs a similar function to the **fix low** button but is used when the sensor is filled with the higher permittivity material. In this case, only the **high-permittivity M3 count values** are adjusted., so there is again no affect on the lower permittivity calibration capacitances. This option also measures a new set of high-permittivity charge injection capacitances. As before, use of this option is guaranteed not to disturb the low permittivity calibration point.

A8.3 CALIBRATION OPTIONS USING EXTERNAL (RECAL) SOFTWARE

Recal is a self-contained executable calibration program which provides a number of additional facilities when compared with the ECT32 software, including:

1. Generation of a single point calibration file where a physical calibration is carried out with the sensor containing air (or any other suitable dielectric material) only. In this case the second (upper) calibration point values are predicted from either a **theoretical sensor model** or from a measured **sensor permittivity file.** This option is useful for generating a calibration file for a sensor in cases where it is difficult to completely fill the sensor with the second dielectric fluid, or for situations where there is no physical fluid, such as a flame or plasma.

2. Changing the calibration permittivity values to modify the measurement range.

3. Generating a sensor capacitance/permittivity data file from sets of calibration data.

The **Recal** program uses either a **theoretical model** of the sensor, or a **file of measured capacitance/permittivity data** for the sensor (**sensor file**) to achieve these objectives. In both cases, the **sensor capacitance/permittivity** relationships are expressed as a set of **polynomials** of the form:

$$\mathbf{C = f\,(K)} \qquad \textbf{A8.2}$$

where **C** is the set of inter-electrode pair capacitances, **K** is the permittivity of the fluid inside the sensor and **f (K)** is a set of polynomial expressions, one for each individual electrode-pair.

The theoretical sensor model is reasonably accurate except for the capacitances between adjacent electrodes for sensors with external electrodes, where it is known to be increasingly inaccurate for sensors with walls of increasing thickness (for reasons which are at present unclear). However, these errors can be largely corrected using data from a conventional calibration file for the sensor. It is therefore possible to characterise the sensor capacitance/permittivity relationships with reasonable accuracy using combinations of measured and calculated data.

The most accurate recalibration will be achieved if the capacitance/permittivity characteristics of the sensor have been measured previously using 3 or more dielectric fluids (eg air, polypropylene beads and glass beads) and stored in a **sensor permittivity file**. The **Recal** program provides a simple method for generating this **sensor permittivity file** from a set of standard **calibration** files. However, as long as the tube wall is thin, reasonable results can also be obtained using just 2 sets of measurements (which produce a set of linear capacitance.permittivity relationships) in the **sensor file** (eg for air and polypropylene beads)

If it is not possible to generate a measured **sensor file** because of sensor size or access problems preclude filling the sensor with known dielectric(s), then a **theoretical model** of the **sensor capacitance/permittivity** characteristics can be used, together with data from a **standard sensor calibration file**. The most basic (and least accurate) recalibration option is to use a calibration file where the sensor has been calibrated containing air only (ie where the upper calibration point contains invalid data). In this case, any errors in the theoretical model can be corrected at the low calibration point only and the accuracy of measured permittivity values remote from the low permittivity calibration point will be limited. If a full valid calibration data file is available, a second (better option) allows the theoretical model to be corrected at each end of the calibration range. This will give good calibration accuracy over the full working range of permittivities. The detailed operation of the **Recal** program is described in the next section.

A8.4. THE RECAL PROGRAM

The **Recal** program software is installed as described in Appendix 7. **Recal is** a stand-alone executable program. When the program is run, by double-clicking on the **Recal** icon in the **ECT Software group** window, the following window appears:

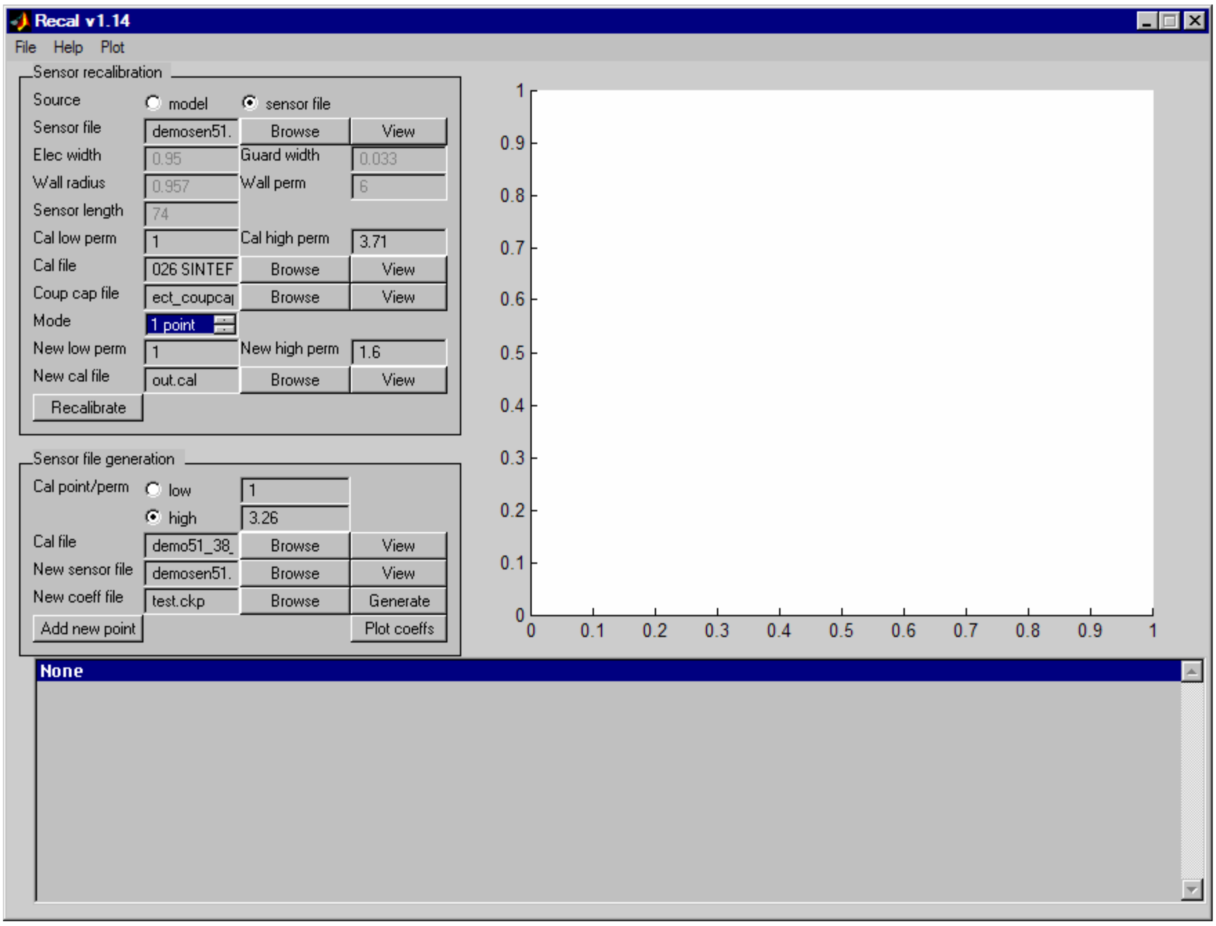


Figure A8.2. The Recal window at start-up

The **Recal** program can be operated in a number of different ways as described later. In all cases, a **measured calibration file,** with valid data for the **low calibration point** must be available before the program can be run.

The **Recal** window is divided into **3 regions.**

Region 1, occupying the **upper left-hand** part of the window contains a set of **control** and **input data buttons** and **data boxes**. The **upper group** of buttons/boxes (**Sensor recalibration)** relate to the generation of modified **calibration files,** while the **lower group** (**Sensor file generation**) are used to generate a **characteristic sensor permittivity file** which contains sets of **measured capacitance data** at **specified values of permittivity.**

Region 2, occupying the **upper right-hand** part of the window is the space where **graphical output data** is generated.

Region 3, occupying the **lower part of the window** is the space where **text output data** is generated.

Before describing some typical recalibration examples, the details of the required input data in **region 1** of the **Recal** window will be described briefly. Fuller descriptions are given in the subsequent text.

A8.4.1 SENSOR RECALIBRATION

The upper group of parameters (**Sensor recalibration**) in **Region 1** of the **Recal** window contains the data required to generate a new or modified calibration file

Source determines where the data for generating the sets of sensor **capacitance/permittivity polynomials** is obtained. This can be from either a **theoretical model** of the sensor, (**Model** option), which requires information about the sensor geometry to be entered, or from a **sensor permittivity file (sensor file** option**)** containing sets of measured capacitance/permittivity data for the sensor.

Sensor file is the experimental data file which holds the **capacitance/permittivity** measured data for the sensor and which is required for use with the **Sensor file** option. The **sensor file** can be generated from one or more calibration files as described in section 4.2.

The **Model** option generates the sets of sensor **capacitance/permittivity** characteristic polynomials from an analytical 2-dimensional sensor model of the sensor. It requires a number of sensor geometry parameters to be input before the file can be generated.

> **Elec width** is the normalised electrode width expressed as an equivalent electrode arc length (cm) , divided by (the circumference of the electrode plane/E) where E is the number of electrodes around the sensor circumference.
>
> **Guard width** is the normalised width of any earthed axial guards located between electrodes, expressed as an equivalent arc length (cm) , divided by (the circumference of the electrode plane/E) where E is the number of electrodes around the sensor circumference.
>
> **Wall radius** is the normalised radius of the the dielectric tube wall, defined as the tube inner diameter divided by its outer diameter.
>
> **Wall perm** is the permittivity of the tube wall. Typical values are 2.5 for perspex and 6 for glass.
>
> **Sensor length** is the axial length of the measurement electrodes in mm.

Note that, depending on the source option selected, data from either the theoretical sensor model or the sensor file are converted into sets of polynomials of the form C = f (K), where C is the set of inter-electrode capacitances and K is the permittivity of the material inside the sensor.

Cal file is the file name of the original measured calibration data file which is to be modified. Note that a valid value for **cal low perm** must be entered before the calibration file is selected using the **Browse** button, as this value is used together with data from the calibration file to derive the **high permittivity value** when the file is loaded (see below).

Cal low perm is the relative permittivity of the lower permittivity material used to create the calibration file to be modified. A suitable value must be entered (eg 1 for air).

Cal high perm is the permittivity of the higher permittivity material used to create the calibration file. A default value is derived automatically from the selected calibration file when the calibration file is selected using the **Browse** button by calculating the value of the ratio of the mean values of the high and low permittivity capacitances for opposite sensor electrodes and multiplying this ratio by the low calibration permittivity value (**cal low perm**). Alternatively, a new value can be entered manually directly into the box. It should be noted that for sensors of increasing wall thickness, the value derived automatically from the calibration file becomes increasing inaccurate and will tend to underestimate the permittivity.

The data in the calibration file can be viewed by clicking on the **View** button. A sample screen shot is given in appendix 2.

Coup cap file is the file containing the internal coupling capacitances within the capacitance measurement system. This file can be **generated or copied** to the **control PC** from the **embedded PC** inside the **DAM200E** unit using the **ECT Toolkit** software. It can be located and viewed in the same way as the calibration file and a sample screenshot is given in appendix 2.

Mode defines the method used to correct the new calibration file.

1 point generates a new calibration file from an existing (original) calibration file using a set of sensor polynomials obtained from either the theoretical sensor model or the sensor measured permittivity file. These polynomials are corrected (offset) so that the capacitances predicted by the polynomials fit the capacitances in the original calibration file at the original lower permittivity calibration point only. The new upper level calibration capacitances are derived directly from these corrected polynomials,

2 point works in a similar way but the polynomials are corrected so that the predicted capacitances fit the calibration capacitances at both the high and low permittivity calibration points of the original calibration file. The fitting is done by applying a combination of offsets at the lower calibration point and gain parameters at the higher calibration point (see section 4.4).

Note that as well as selecting the required mode using the scroll arrows, the selected option must be highlighted (blue) by left clicking on it to make it active.

New low perm is the required **low permittivity calibration point** for the new (modified) calibration file. If the **new low perm** is the same value as that in the original input calibration file **(cal low perm)**, the low permittivity parameters in the calibration file are unchanged.

New high perm is the required high permittivity calibration file point for the new (modified) calibration file.

New cal file is the name of the new calibration file to be generated. The **Browse** and **View** buttons work as before and allow the new file to be generated in the required folder.

Once a valid set of data has been entered, the **Recalibrate** button is used to generate the new calibration file.

A8.4.2 DETAILS OF PROGRAM OUTPUTS IN RECALIBRATION MODE

When the **Recalibrate** button is clicked, the program runs and two sets of output are generated. A set of text output data appears in Region 3 at the bottom of the Recal window and a set of graphical output data appears in Region 2 (the white space next to the input data boxes). A typical set of output data is shown in figure 3 below:

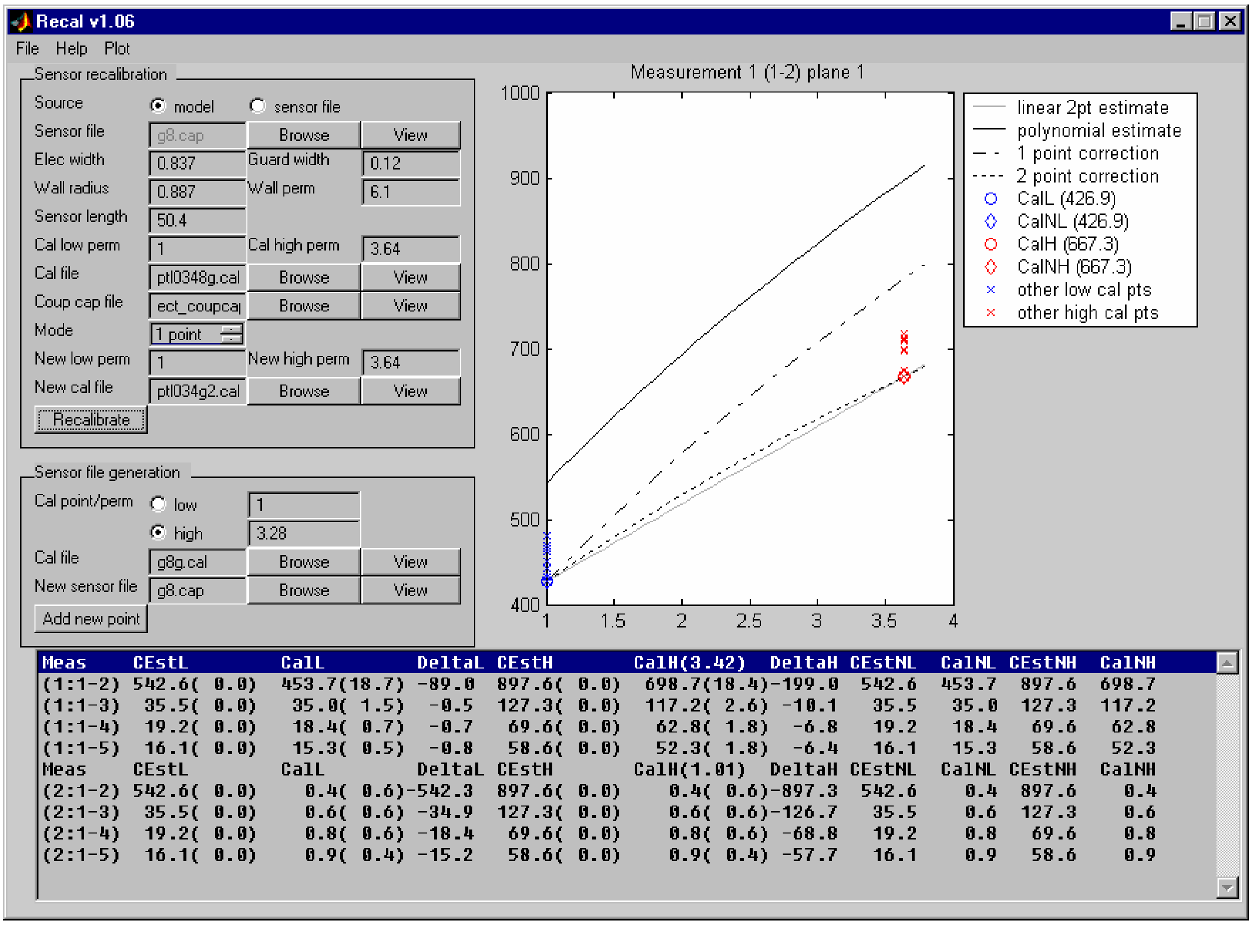


Figure A8.3. Recal window showing output data

A8.4.3 TEXT OUTPUTS

The text output is a set of data arranged in columns as follows:

Meas: The measurement plane, followed by the generic electrode-pair. The generic electrode pair means all measurements of the same combinations of electrode-types. For example, the generic pair C1-2 is the set of adjacent electrode measurements C1-2, C2-3, C3-4, C4-5, C5-6, C7-8, C8-1 etc. In principle, the capacitances for all of these electrode-pairs should be identical for a symmetrical sensor containing a homogeneous material. However, in practice, they will vary slightly because the sensor inter-electrode region is extremely sensitive to small dimensional variations.

CestL: The inter-electrode capacitances for the generic electrode-pair specified in the **Meas** column, calculated from either the t**heoretical sensor model** polynomial or the polynomial derived from the **sensor permittivity file**, at the low permittivity calibration point of the original calibration file (**Cal low perm**). The figure is the average of all similar generic electrode-pairs.

CalL: The averages of the sets of capacitance measurements for each generic electrode pair, derived from the low permittivity point of the original sensor calibration file.

DeltaL = CalL-CestL

CestH = The capacitances estimated from either the theoretical model or the sensor permittivity file polynomial at the high permittivity calibration point of the original calibration file (**Cal high perm**).

CalH = The averages of the sets of capacitance measurements for each generic electrode pair, derived from the high permittivity point of the sensor calibration file.

DeltaH = CalH-CestH

CEstNL = The capacitances estimated from either the theoretical model or the sensor file polynomials at the quoted new low permittivity calibration point (**New low perm**).

CalNL = A modified version of **CEstNL** obtained by forcing the model polynomials to fit the calibration capacitances at the original low calibration permittivity (**cal low perm**). That is, the original sensor model (**C = f (K)**) is modified to have the form: **C = f (K) + O**, where **O** is the set of offset capacitances (= **DeltaL**) required to match the estimated capacitances to the measured capacitances at the original low calibration point (**cal low perm**). This modification is carried out to the new calibration file data in both 1 point and 2 point modes. If the new low perm is not equal to the cal low perm then in 2pt mode a set of gain parameters is also used to fit the estimated curve to the calibration high point – see comments below.

CEstNH = The capacitance estimated from either the theoretical model or the sensor file polynomials at the new high permittivity calibration point (**New high perm**).

CalNH = A modified version of **CEstNH** obtained by forcing the model polynomial to fit the the calibration capacitances at the measured low calibration permittivity (**cal low perm**) as well as at the measured high permittivity calibration point **(cal high perm**). That is, the model is modified to have the form: **C = G. f (K) + O**, where O is the set of offset capacitances (= **DeltaL**) required to match the estimated capacitances to the measured capacitances at the original low calibration point (**cal low perm**) and **G** is a set of gain parameters chosen to match the estimated capacitances to the measured capacitances at the original high calibration point (**cal high perm**). This modification is only carried out to the new calibration file data when the **2-point mode** is selected. Note that the numbers in parentheses are the rms variation from the quoted mean to its left.

A8.4.4 GRAPHICAL OUTPUTS

The graphical output (Region 2) shows a number of calculated/measured capacitance/permittivity characteristic for a selected electrode-pair. The default plot set shows data for the capacitances between electrodes 1 and 2 (C1-2). Other capacitances can be viewed by selecting the required electrode-pair using the **Plot** menu in the **Recal** window.

In figure 3, the solid grey line (**linear 2 pt estimate**) is drawn between the high and low permittivity points for the specified electrode-pair capacitances, calculated from the **original input calibration file**. If the original calibration file has valid data at both the low and high permittivity calibration points, this line is a good yardstick against which other calculated calibration data can be compared.

The solid black line (**polynomial estimate**) is the capacitance/permittivity characteristic calculated from either the **theoretical sensor model** or the **sensor permittivity file**, depending on which **Source** mode is selected.

The plot composed of long and short dashes (**1 point correction)** is the **polynomial estimate**, adjusted to fit the values derived from the **lower permittivity point** in the **original calibration file**. The adjustment is a simple offset applied to the lower permittivity point capacitances.

The curve composed of short dashes (2 point correction) is again the **polynomial estimate**, but this time matched at both the **high and low permittivity points** of the **original input calibration file**.

A number of specific measurements are also plotted in the graphical output data as follows:

Blue circle: The value of **CalL** (the original low calibration point capacitances for the specified **electrode-pair,** calculated from the original calibration file.

Blue diamond: The value of **CalNL** (the new low calibration point capacitance for the specified electrode-pair) calculated from either the sensor model or sensor permittivity file polynomial, corrected as specified against the supplied calibration file. Note that this point will lie on the 1-point calibration (long and short dashes) line if the 1 point mode is selected and on the 2-point (short dashes) line if the 2-point mode is selected.

Blue crosses: The values of **CalNL** for the other similar generic electrode-pair capacitances at the original **low** calibration point.

Red circle: The average values of **CalH** (the original high calibration point capacitances) for the **generic electrode-pair,** calculated from the **original calibration file**.

Red diamond: The value of **CalNH** (the new high calibration point capacitances) calculated from either the sensor model or sensor permittivity file polynomial, corrected as specified against the supplied calibration file. Note that this point will lie on the 1-point calibration (long and short dashes) line if the 1 point mode is selected and on the 2-point (short dashes) line if the 2-point mode is selected.

Red crosses: The values of **CalNH** for the other similar generic electrode-pair capacitances at the **original high calibration point**.

Note that a full set of graphs are produced irrespective of whether the one-point or 2-point options are selected. However, the format of the new output calibration file is determined by whether the one or two-point option is selected.

A8.4.4.1 Comments on graphical outputs

Viewing figure 3 for the case of the C1-2 capacitance, it is clear that the capacitances derived from the theoretical model (polynomial estimate) are erroneous but that they can be progressively improved by adjusting the polynomial so that it matches the data to that in the calibration file at one permittivity value (1 point correction) or two permittivity values (2 point correction).

If other capacitance pairs are viewed using the **Plot** menu, the match between the model polynomial predictions and the measured data is seen to improve considerably. For example, figure 4 shows the outputs for C1-3.

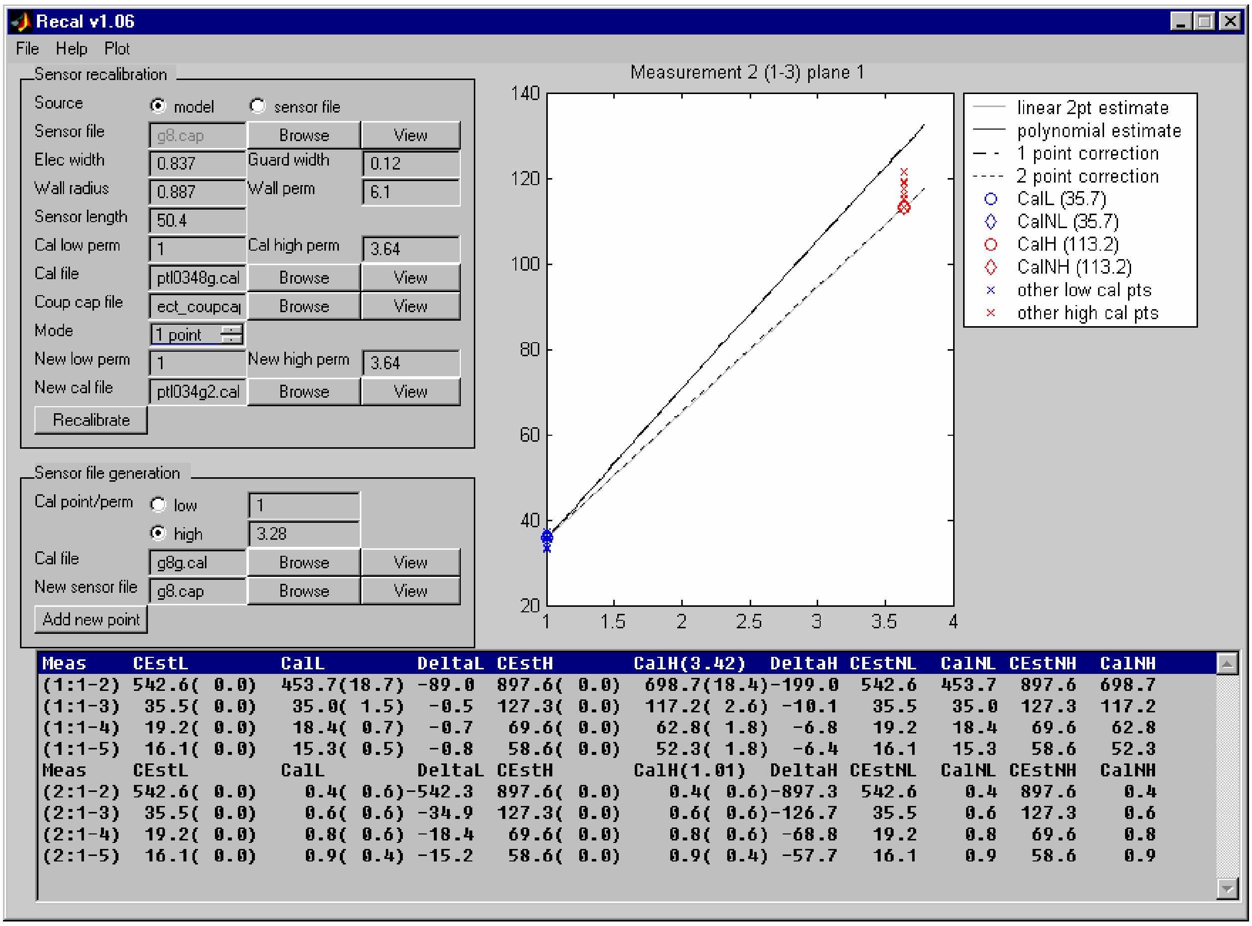


Figure A8.4. Plot outputs for C1-3

A8.4.5 CREATION OF A SENSOR PERMITTIVITY (C/K) FILE

A8.4.5.1 INPUT PARAMETERS AND CONTROL BUTTONS

A **sensor permittivity file**, containing information about the sensor capacitances measured with the sensor filled with materials of known permittivity, can be constructed from one or more standard calibration files for the sensor.

The process for generating the measured sensor permittivity file is to build up points (permittivities) in the file, by adding them one at a time from either the high or low permittivity calibration points from specified calibration files (in any order).

The process is carried out by inputting data into the **sensor file generation group** of boxes in the **Sensor File Generation region** of the **Recal** window. The function of the various controls and boxes in this region is as follows:

Cal point/perm. Selects the **low calibration point** and enters the **relative permittivity** of the **low permittivity calibration point** at which the cal file (see below) was generated. It is important that the specified permittivity is accurate for each set of input calibration data.

Cal file. Enters the name of the calibration file from which the capacitance/permittivity data is to be extracted.

> **Cal point/perm**. Once the required **input calibration file** has been selected, the **high permittivity** figure for the file will be calculated automatically and entered in the **Cal high perm** box from the opposite electrode capacitance ratios at the high and low permittivity points, assuming a low calibration permittivity of 1. For sensors of increasing wall thickness, this derived value becomes increasing inaccurate and will tend to underestimate the permittivity.

View Cal file button: Displays the **calibration data** as a file of both absolute capacitances and primary measurement data in integer format.

New sensor file. The name of the sensor permittivity file which is to hold the capacitance/permitititivity data for the sensor.

View New Sensor file button: Displays the **Sensor permittivity file** as sets of absolute capacitances and relative permittivities.

Add new point. This button adds a new data point to the **New sensor file**.

New Coeff. file. The name of the data file which is to hold the coefficients for the polynomial approximation to the sensor permittivity file.

Generate: This button generates the **sensor coefficients file** from the sensor permittivity file.

Plot Coeffs. This button plots the sensor capacitance/permittivity characteristics for an electrode-pair specified in the **Plot menu**.

A8.4.5.2. GENERATING THE SENSOR C/K AND COEFICIENTS FILES

The **sensor C/K file** and the related **coefficients file** are generated from a number of **individual sensor calibration files** measured with the sensor filled with a range of materials having differing permittivities.

The files can be generated as follows:

1. In the **Sensor Recalibration** section, use the **Browse** button to locate the Coupling capacitance file (Coup cap file) for the ECT system in use.

2. In the **Sensor File generation** section, use the **Browse** button to locate the **calibration file** from which the **low permittivity** capacitance/permittivity calibration data is to be extracted.

3. Select the **low calibration point** option in the **Cal point/perm** area of the **sensor file generation** section and enter the relative permittivity of the low permittivity calibration point at which the cal file to be input (see below) was generated (= 1 for air). It is important that the specified permittivity is accurate for each set of input calibration data.

4. Once the required input **calibration file** has been selected, the **high permittivity** figure for the file will be calculated automatically and entered in the **Cal high perm** box. This figure is calculated from the **ratio of the capacitances of opposing electrode-pairs** at the **high** and **low permittivity points**, assuming that the low permitivity has the value 1. However, it should be noted that for sensors of increasing wall thickness, this derived value becomes increasing inaccurate and will tend to underestimate the mixture permittivity.

5. Enter the name of the **sensor permittivity file** which is to hold the capacitance/permittivity data for the sensor in the **New sensor file** box.

6. Click on **Add new point**. This button will add a new data point at the **low permittivity calibration point** to the **New sensor file**.

7. Set the **Cal point/perm to s**elect the **high** calibration point and again click on **Add new point**. This will add the **high permittivity calibration data** from the calibration file to the **sensor file**.

8. Use the **Browse** button to locate another calibration file containing data from a calibration at a different high permittivity value.

9. Repeat steps 7 and 8 for as many calibration files as data is available to build up the sensor permittivity file.

10. The **sensor file** can be viewed in numerical format at any stage by clicking on the **New sensor file View** button.

10. Once all the points have been added, generate the **Coefficients file** by clicking on the **Generate** button.

11. To view the **sensor** and coefficients files in graphical format, click on the **Plot coeffs** button. The **capacitance/permittivity** characteristics for individual electrode-pairs will be plotted as shown in figure 5.

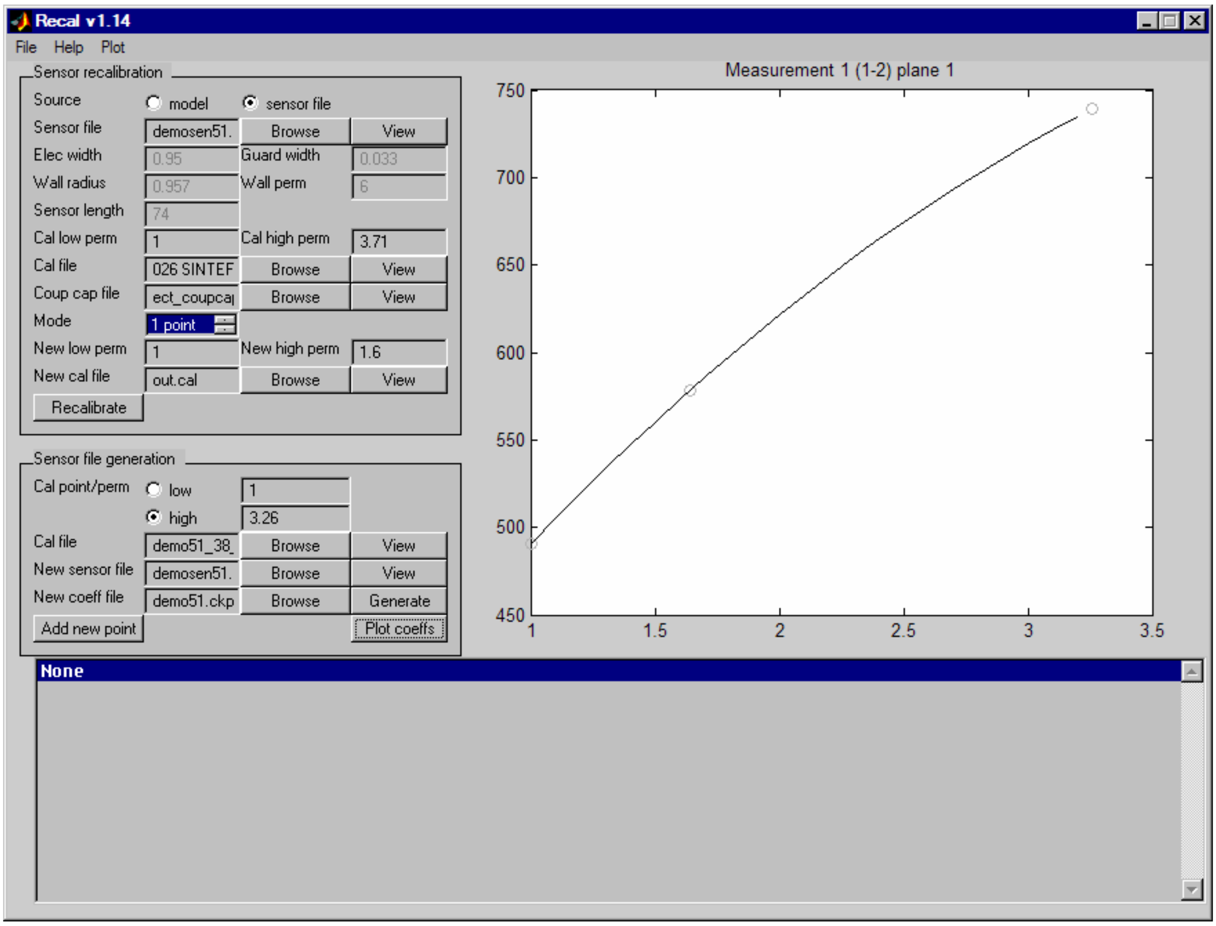


Figure A8.5. Recal window showing C/K curves for electrode-pair 1-2

In figure 5, the discrete points are the experimental values for the specified electrode-pair extracted from the **sensor file** (demosen51.cap) defined in the upper **Sensor Recalibration area** of the Recal window. The continuous curve is generated using a second order polynomial derived from the data in the **sensor coefficients** file. Note that the **Sensor file** defined in the upper part of the **Recal** window must be identical to the **New Sensor file** defined in the lower part of the **Recal** window.

The characteristics for the other electrode-pairs can be viewed by clicking on the **Plot menu** and selecting the required **electrode-pair** from the **drop-down list**. The **selected electrode-pair** is listed at the **top of the Recal window**.

A8.5. RECALIBRATION BASICS

For a normal ECT sensor caibration, the permittivity image pixels and the normalised inter-electrode capacitances will all have the value 1 when the sensor is filled with the higher permittivity material used to calibrate the sensor. This situation is shown in figure A.8.5 below for the case when the sensor is filled with glass beads and a standard calibration file for air/glass beads is used. Note that the volume ratio is correct (100%)

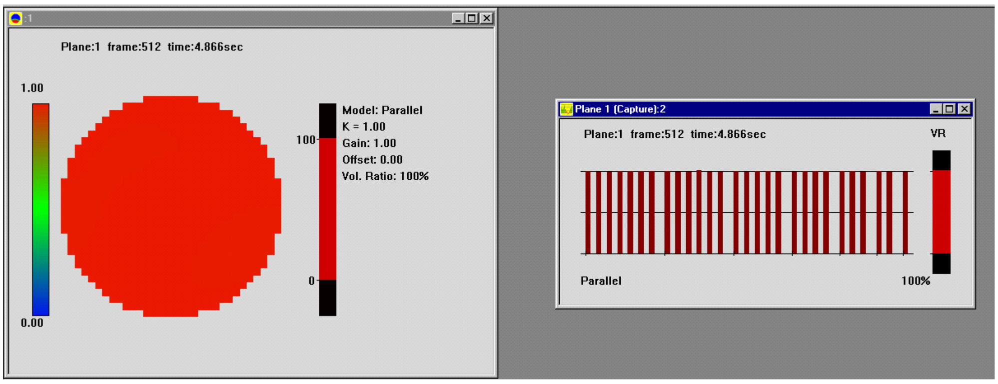


Figure A8.5. Permittivity image and normalised capacitances for sensor filed with glass beads

If we now remove the glass beads from the sensor and refill it with polypropylene beads without recalibrating the sensor, the resulting permittivity image and normalised capacitances are as shown in figure A.8.6.

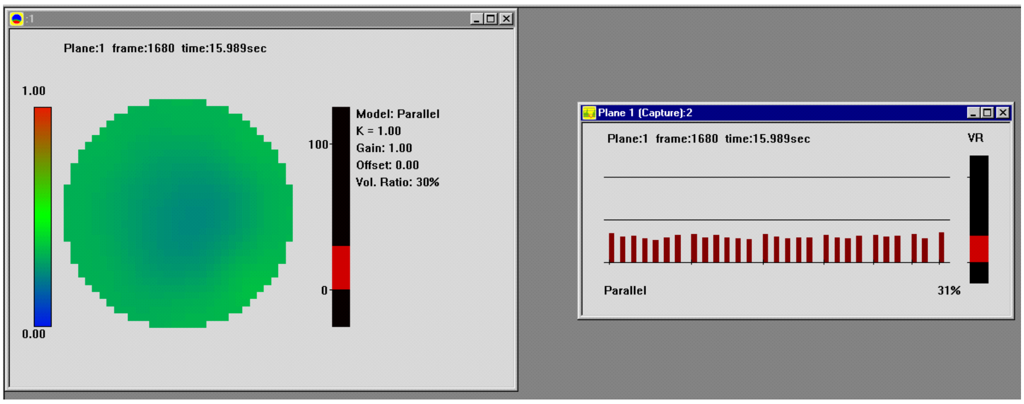


Figure A8.6. Image and capacitances when glass beads replaced by polyproylene beads.

Even though the sensor is filled with beads, the volume ratio (30%) is clearly wrong as we are not using a valid calibration file. However, suppose we are unable for some reason to completely fill the sensor with polypropylene beads but nevertheless wish to generate a calibration file which gives the correct volume ratio for polypropylene beads. It is possible to generate the correct calibration file from the data in the glass calibration file using the **Recal** program. There are a number of alternative ways in which **Recal** can generate the required calibration file and these will described in the next few sections.

A8.6. SENSOR RECALIBRATION USING RECAL

We will asume that we have an existing calibration file (PTL034g.cal) created for a low permittivity of air (K=1) and a high permittivity of a mixture of glass beads and air (K = 3.4), and that we wish to create a new calibration file suitable for use with air (K = 1) and a mixture of plastic beads and air (K=1.7).

A8.6.1. GENERATION OF A CALIBRATION FILE USING THE THEORETICAL SENSOR MODEL

A8.6.1.1 1-point correction

For almost all ECT sensors, it is possible to measure the capacitances with nothing but air inside the sensor and we can therefore always create a calibration file with valid data at the low (air) permittivity level.

The **Recal** screen for this situation is shown in figure A.8.7

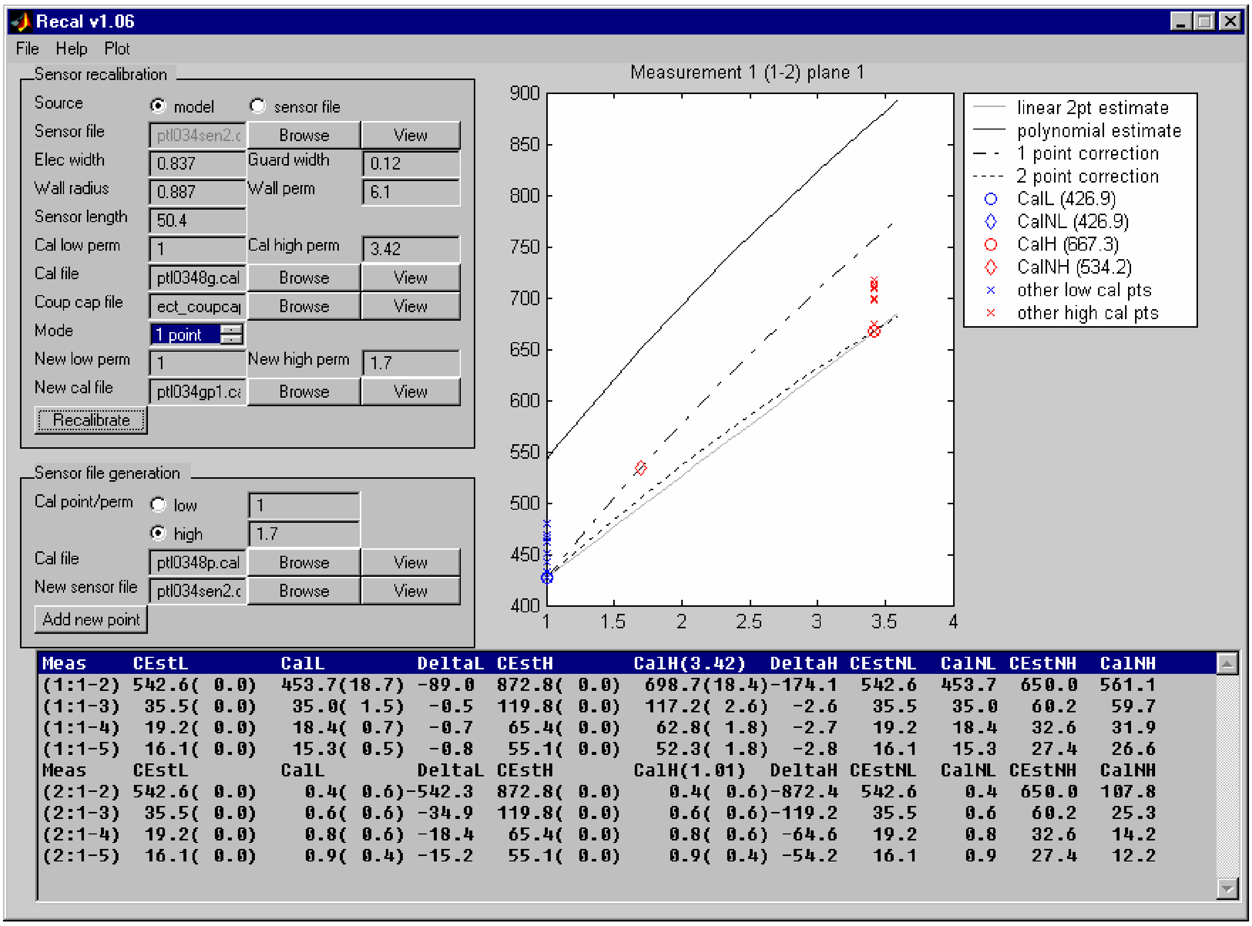


Figure A8.7. Recal screen for 1-point model recalibration

Note that the **1-point model option** has been selected and the sensor geometry parameters required for this model have been entered. The Cal high perm figure (3.42) has been derived automatically from the input calibration file for the glass beads (ptl0348g.cal). Note also that the New high perm figure has been set to the correct value for the polypropylene beads (1.7).

The output data in figure A8.7 shows the result of running **Recal** with this input data. The new output calibration file name is ptl034gp1.cal.

When this new calibration file is used in the ECT32 software with the sensor filled with polypropylene beads, the results are shown in figure A8.8.

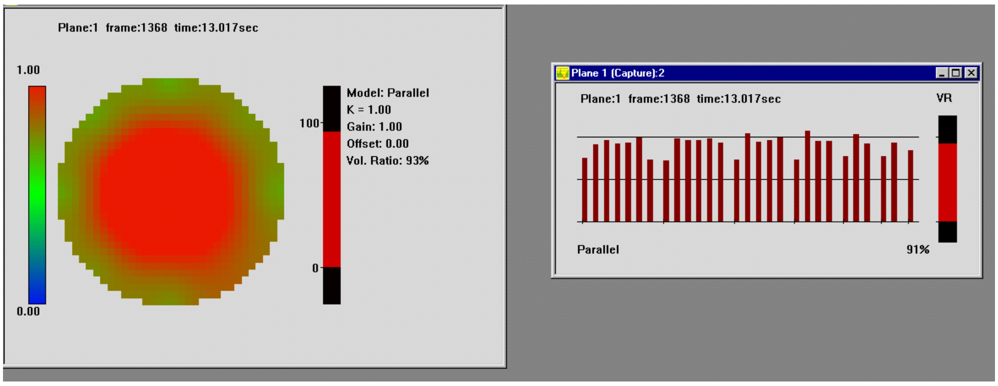


Figure A8.8. Sensor containing plastic beads using calibration file ptl034gp1.cal. (Theoretical model with 1-point correction)

These results show the limitations of using a 1-point recalibration method with the theoretical model. Although most of the normalised capacitances are approximately correct, the adjacent capacitances are too low and cause a low permittivity anulus (green ring) to appear around the outside of the permittivity image.

A8.6.1.2 2-point correction

Figure A.8.9 below shows the improvements which result if a 2-point correction method is used with the theoretical model. The high level capacitances and image are now almost correct.

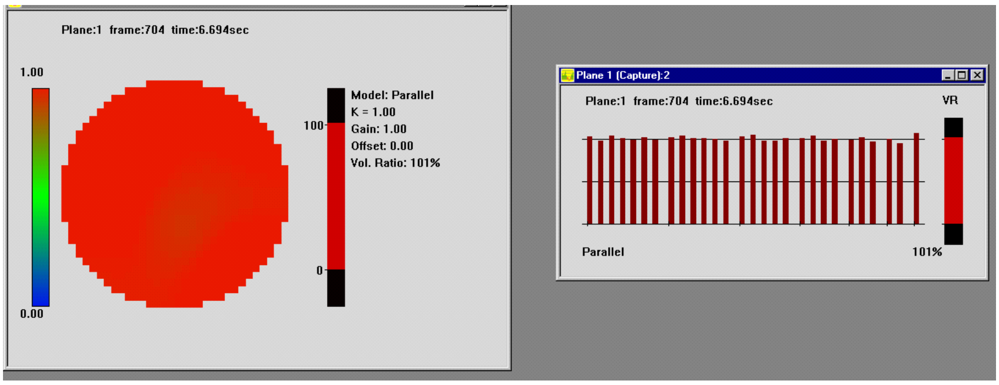


Figure A8.9. Sensor containing plastic beads using calibration file ptl034gp2.cal. (Theoretical model with 2-point corection)

The 2-point method is an effective method for recalibrating a sensor using an existing calibration file.

A8.6.2. GENERATION OF A CALIBRATION FILE USING A SENSOR PERMITTIVITY FILE

The next two sets of results were obtained uisng the **Sensor file** option. The sensor file contains 3 sets of calibration data measured at permittivities of 1 (air), 1.70 (polypropylene beads/air) and 3.42 (glass beads/air). This file was generated as described in section 4.2.

A8.6.2.1 1-point correction

Figure A.8.10 shows the results for the sensor filled with polypropylene beads using a calibration file generated from the **sensor permittvity file** and corrected at the low calibration point only. The voidage and normalised capacitances are almost correct.

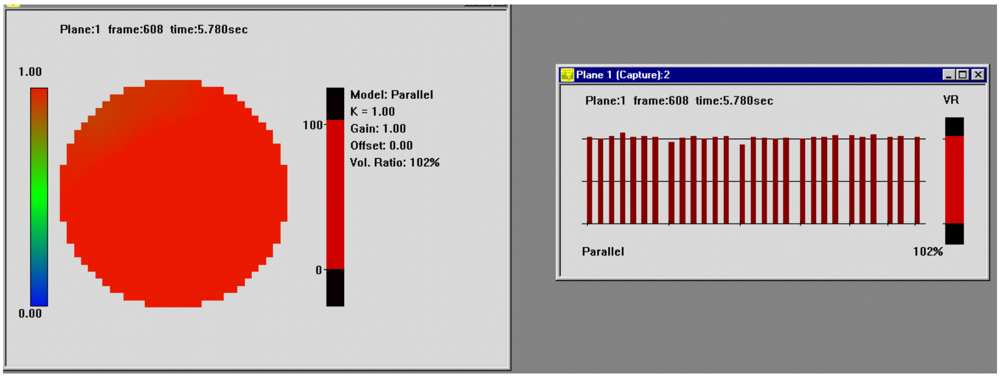


Figure A8.10. Sensor containing plastic beads using calibration file ptl034gps1.cal. (Sensor permittivity file with 1-point correction)

A8.6.2.2 2-point correction

Figure A.8.11 below shows the same results for a calibration file generated from the sensor permittivity file and corrected at the high and low calibration points for glass beads. There is a slight improvement in accuracy over the results obtained using the 1-point calibration file.

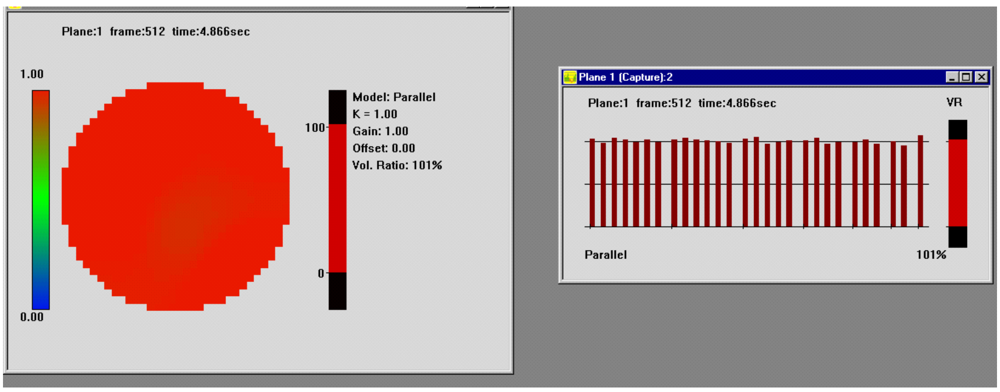


Figure A8.11. Sensor containing plastic beads using calibration file ptl034gps2.cal. (Sensor permittivity file with 2-point correction)

A8.6.3 OTHER RECALIBRATION OPTIONS

So far we have only modified the upper calibration permittivity. However, it is also possible to modify the lower calibration point. This can be useful, for example to produce a restricted permittivity measurement range, resulting in improved measurement resolution over a narrow range of permittivity variations.

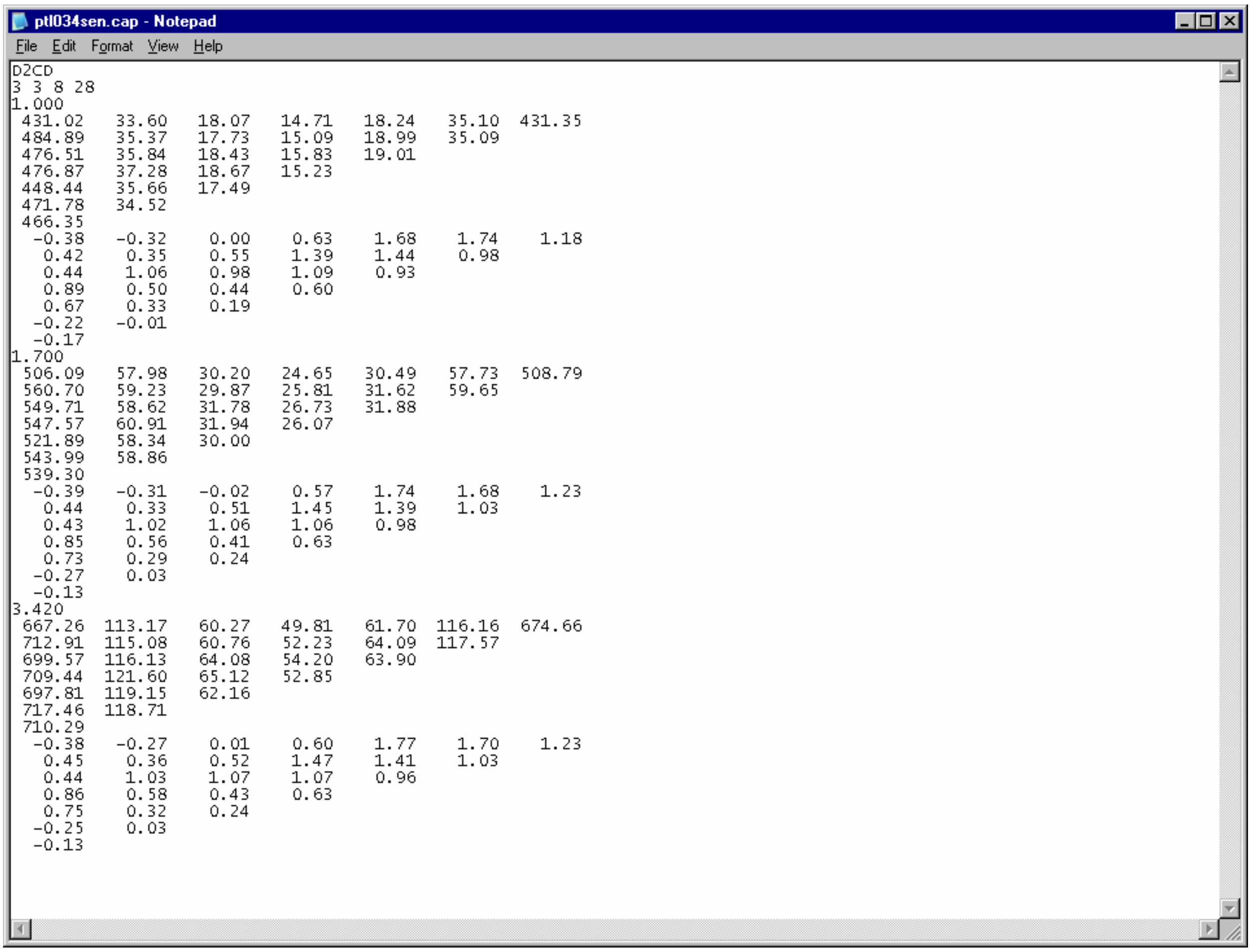
A8.7. CONCLUSIONS

The combination of recalibration options in the **ECT32** and **Recal** software allow ECT sensors to be recalibrated with relative ease. Generation of a new calibration file from a single set of calibration parameters in air is possible using a theoretical sensor model, although the results are of limited accuracy. If a sensor permittivity file is available, single-point calibration gives good results and 2-point correction further improves these results.

For all ECT sensors it is worth creating a sensor permititvity file wherever possible to assist in subsequent sensor calibrations.

A8.8 SCREENSHOTS OBTAINED USING RECAL VIEW BUTTONS

A8.8.1. Contents of a typical sensor Permittivity File as listed by View button




```
D2CD
3 3 8 28
1.000
 431.02   33.60   18.07   14.71   18.24   35.10  431.35
 484.89   35.37   17.73   15.09   18.99   35.09
 476.51   35.84   18.43   15.83   19.01
 476.87   37.28   18.67   15.23
 448.44   35.66   17.49
 471.78   34.52
 466.35
  -0.38   -0.32    0.00    0.63    1.68    1.74    1.18
   0.42    0.35    0.55    1.39    1.44    0.98
   0.44    1.06    0.98    1.09    0.93
   0.89    0.50    0.44    0.60
   0.67    0.33    0.19
  -0.22   -0.01
  -0.17
1.700
 506.09   57.98   30.20   24.65   30.49   57.73  508.79
 560.70   59.23   29.87   25.81   31.62   59.65
 549.71   58.62   31.78   26.73   31.88
 547.57   60.91   31.94   26.07
 521.89   58.34   30.00
 543.99   58.86
 539.30
  -0.39   -0.31   -0.02    0.57    1.74    1.68    1.23
   0.44    0.33    0.51    1.45    1.39    1.03
   0.43    1.02    1.06    1.06    0.98
   0.85    0.56    0.41    0.63
   0.73    0.29    0.24
  -0.27    0.03
  -0.13
3.420
 667.26  113.17   60.27   49.81   61.70  116.16  674.66
 712.91  115.08   60.76   52.23   64.09  117.57
 699.57  116.13   64.08   54.20   63.90
 709.44  121.60   65.12   52.85
 697.81  119.15   62.16
 717.46  118.71
 710.29
  -0.38   -0.27    0.01    0.60    1.77    1.70    1.23
   0.45    0.36    0.52    1.47    1.41    1.03
   0.44    1.03    1.07    1.07    0.96
   0.86    0.58    0.43    0.63
   0.75    0.32    0.24
  -0.25    0.03
  -0.13
```

Figure A8.12 Contents of the Sensor permittivity file

The file contents are as follows:

D2CD: Measurement system code

3 3 8 28: Number of permittivity values, Plane code (2 Planes), Number of electrodes, Number of inter-electrode measurements

1.000: Permittivity
Inter-electrode capacitances fF for plane 1.
Inter-electrode capacitances fF for plane 2

These blocks of data are repeated for each entered permittivity value.

A8.8.2 Calibration file as listed by View button

```
Absolute Capacitances in fF (using corrections from C:\ect32v2\Coupling
files\ect_coupcap_034_1.0.cap)

Plane 1
Charge injection capacitances
 -13.77   -0.77    5.45  -26.80  -10.95  -25.21    3.52
 -13.80   -0.77    5.33  -26.89  -10.57  -25.24    3.57
Low capacitances
 426.86   35.70   17.65   14.73   18.33   33.66  432.46
 480.26   35.42   17.50   15.07   18.64   37.25
 465.77   33.63   19.13   15.75   18.84
 468.66   34.77   19.27   15.59
 442.22   33.23   18.17
 462.19   36.56
 450.91
High capacitances
 667.26  113.17   60.27   49.81   61.70  116.16  674.66
 712.91  115.08   60.76   52.23   64.09  117.57
 699.57  116.13   64.08   54.20   63.90
 709.44  121.60   65.12   52.85
 697.81  119.15   62.16
 717.46  118.71
 710.29
Estimated Perm Ratio = 3.42

Plane 2
Charge injection capacitances
  -5.03    0.23   -0.74    6.10   -0.38    3.33   -1.94
  -5.00    0.22   -0.73    6.10   -0.41    3.37   -1.94
Low capacitances
  -0.35   -0.31   -0.00    0.59    1.74    1.73    1.21
   0.42    0.36    0.52    1.43    1.44    1.01
   0.44    1.00    1.03    1.10    0.96
   0.88    0.55    0.46    0.62
   0.73    0.32    0.22
  -0.22    0.02
  -0.13
High capacitances
  -0.38   -0.27    0.01    0.60    1.77    1.70    1.23
   0.45    0.36    0.52    1.47    1.41    1.03
   0.44    1.03    1.07    1.07    0.96
   0.86    0.58    0.43    0.63
   0.75    0.32    0.24
  -0.25    0.03
  -0.13
```

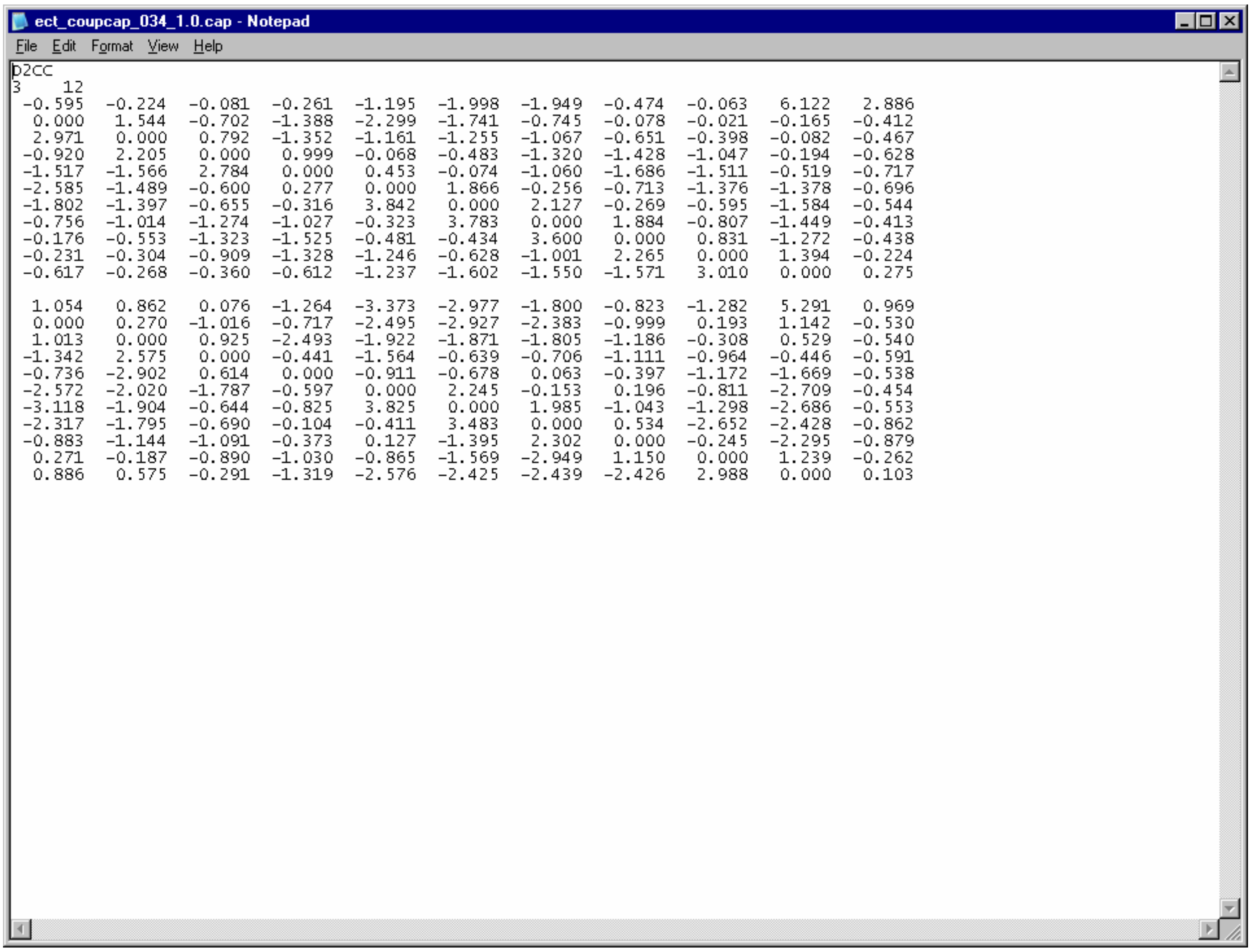
Estimated Perm Ratio = 1.01

Raw data (contents of C:\ect32v2\working\ptl0348g.cal)

```
D2CB
3   8   28
 971
 213   218   220   208   214   208   219
 916   897   958   932   927  1038  1006

 972
 213   218   220   208   214   208   219
 915   898   951   927   953  1037  1010
 375   230   227   213   220   221   381
 401   233   214   219   215   232
 400   221   220   214   226
 388   228   215   224
 383   222   225
 387   232
 396
 925  1043   903  1010   999   918   979
 918   971   949   897   936  1041
 851   889  1039   945   956
 883   885  1030  1061
 909   887  1057
 875  1034
 816
 149   444   859  1001   801   441   146
 155   447   831   975   793   429
 159   446   760   935   792
 152   422   755   915
 141   429   775
 144   420
 145
3251  3277  3274  3285  3275  3280  3276
3260  3280  3279  3274  3275  3280
3264  3276  3276  3278  3276
3259  3275  3277  3278
3253  3280  3273
3262  3276
3259
 971
 216   218   218   220   218   219   217
 967   964   899  1001   923   993   996

 972
 216   218   218   220   218   219   217
 969   963   900  1001   921   996   996

 216   218   218   220   217   219   217
 218   217   220   217   219   217
 218   220   217   219   217
 220   217   219   217
 218   219   217
 220   217
 218
1014  1001   904   956   991   910   957
1010  1032   988  1029   894   905
 990   902  1040   942   940
1030  1032   981   990
 911   969  1015
 951   987
 943
```

```
1023  1023  1023  1023  1023  1023  1023
1023  1023  1023  1023  1023  1023
1023  1023  1023  1023  1023
1023  1023  1023  1023
1023  1023  1023
1023  1023
1023
1014  1002   906   957   991   911   958
1011  1033   988  1029   895   906
 991   904  1041   943   940
1029  1032   982   991
 910   972  1016
 952   988
 943
```

A8.8.3 Coupling capacitance file as listed by View button



ect_coupcap_034_1.0.cap - Notepad

File Edit Format View Help

```
p2CC
3     12
 -0.595  -0.224  -0.081  -0.261  -1.195  -1.998  -1.949  -0.474  -0.063   6.122   2.886
  0.000   1.544  -0.702  -1.388  -2.299  -1.741  -0.745  -0.078  -0.021  -0.165  -0.412
  2.971   0.000   0.792  -1.352  -1.161  -1.255  -1.067  -0.651  -0.398  -0.082  -0.467
 -0.920   2.205   0.000   0.999  -0.068  -0.483  -1.320  -1.428  -1.047  -0.194  -0.628
 -1.517  -1.566   2.784   0.000   0.453  -0.074  -1.060  -1.686  -1.511  -0.519  -0.717
 -2.585  -1.489  -0.600   0.277   0.000   1.866  -0.256  -0.713  -1.376  -1.378  -0.696
 -1.802  -1.397  -0.655  -0.316   3.842   0.000   2.127  -0.269  -0.595  -1.584  -0.544
 -0.756  -1.014  -1.274  -1.027  -0.323   3.783   0.000   1.884  -0.807  -1.449  -0.413
 -0.176  -0.553  -1.323  -1.525  -0.481  -0.434   3.600   0.000   0.831  -1.272  -0.438
 -0.231  -0.304  -0.909  -1.328  -1.246  -0.628  -1.001   2.265   0.000   1.394  -0.224
 -0.617  -0.268  -0.360  -0.612  -1.237  -1.602  -1.550  -1.571   3.010   0.000   0.275

  1.054   0.862   0.076  -1.264  -3.373  -2.977  -1.800  -0.823  -1.282   5.291   0.969
  0.000   0.270  -1.016  -0.717  -2.495  -2.927  -2.383  -0.999   0.193   1.142  -0.530
  1.013   0.000   0.925  -2.493  -1.922  -1.871  -1.805  -1.186  -0.308   0.529  -0.540
 -1.342   2.575   0.000  -0.441  -1.564  -0.639  -0.706  -1.111  -0.964  -0.446  -0.591
 -0.736  -2.902   0.614   0.000  -0.911  -0.678   0.063  -0.397  -1.172  -1.669  -0.538
 -2.572  -2.020  -1.787  -0.597   0.000   2.245  -0.153   0.196  -0.811  -2.709  -0.454
 -3.118  -1.904  -0.644  -0.825   3.825   0.000   1.985  -1.043  -1.298  -2.686  -0.553
 -2.317  -1.795  -0.690  -0.104  -0.411   3.483   0.000   0.534  -2.652  -2.428  -0.862
 -0.883  -1.144  -1.091  -0.373   0.127  -1.395   2.302   0.000  -0.245  -2.295  -0.879
  0.271  -0.187  -0.890  -1.030  -0.865  -1.569  -2.949   1.150   0.000   1.239  -0.262
  0.886   0.575  -0.291  -1.319  -2.576  -2.425  -2.439  -2.426   2.988   0.000   0.103
```

Figure A8.13 Coupling capacitances window

A8.8.4 New calibration file as listed by View button (file generated for same K values as original calibration file)

Absolute Capacitances in fF (using corrections from C:\ect32v2\Coupling files\ect_coupcap_034_1.0.cap)

```
Plane 1
Charge injection capacitances
 -13.77   -0.77    5.45  -26.80  -10.95  -25.21    3.52
 -13.80   -0.77    5.33  -26.89  -10.57  -25.24    3.57
Low capacitances
 426.96   35.71   17.65   14.72   18.32   33.66  432.37
 480.29   35.45   17.51   15.07   18.65   37.26
 465.72   33.62   19.13   15.76   18.85
 468.71   34.75   19.26   15.60
 442.33   33.21   18.18
 462.23   36.56
 451.00
High capacitances
 667.26  113.19   60.27   49.80   61.69  116.16  674.59
 712.87  115.09   60.76   52.22   64.09  117.57
 699.55  116.13   64.09   54.21   63.89
 709.45  121.57   65.11   52.85
 697.92  119.15   62.15
 717.56  118.72
 710.24
Estimated Perm Ratio = 3.42

Plane 2
Charge injection capacitances
  -5.03    0.23   -0.74    6.10   -0.38    3.33   -1.94
  -5.02    0.20   -0.74    6.09   -0.43    3.36   -1.96
Low capacitances
  -0.33   -0.29    0.01    0.57    1.76    1.71    1.23
   0.41    0.38    0.54    1.42    1.45    1.00
   0.46    0.99    1.01    1.12    0.98
   0.89    0.53    0.44    0.60
   0.72    0.33    0.24
  -0.21    0.03
  -0.11
High capacitances
  -0.35   -0.24    0.04    0.60    1.80    1.70    1.26
   0.45    0.39    0.55    1.47    1.44    1.03
   0.47    1.03    1.07    1.10    0.99
   0.89    0.58    0.43    0.63
   0.75    0.35    0.27
  -0.22    0.06
  -0.10
Estimated Perm Ratio = 1.02

Raw data (contents of C:\ect32v2\working\ptl034g2.cal)
D2CB
3   8   28
 971
 213  218     220   208   214   208   219
 916  897     958   932   927  1038  1006

 972
 213  218     220   208   214   208   219
 915  898     951   927   953  1037  1010
```

375	230	227	213	220	221	381
401	233	214	219	215	232	
400	221	220	214	226		
388	228	215	224			
383	222	225				
387	232					
396						
926	1042	904	1009	998	919	978
919	972	950	898	937	1040	
852	890	1038	946	957		
884	886	1029	1060			
910	888	1056				
876	1033					
819						
149	437	847	981	789	433	143
153	439	817	961	781	421	
157	439	749	919	779		
151	415	743	899			
141	421	763				
143	413					
143						
3251	3241	3242	3238	3240	3238	3228
3230	3239	3240	3240	3240	3237	
3235	3240	3243	3239	3238		
3244	3236	3240	3238			
3254	3237	3237				
3247	3238					
3227						
971						
216	218	218	220	218	219	217
967	964	899	1001	923	993	996
972						
216	218	218	220	218	219	217
969	963	900	1001	921	996	996
216	218	218	220	217	219	217
218	217	220	217	219	217	
218	220	217	219	217		
220	217	219	217			
218	219	217				
220	217					
218						
1015	1002	905	955	992	909	958
1009	1033	989	1028	895	904	
991	901	1039	943	941		
1031	1031	980	989			
910	970	1016				
952	988					
944						
1023	1023	1023	1023	1023	1023	1023
1023	1023	1023	1023	1023	1023	
1023	1023	1023	1023	1023		
1023	1023	1023	1023			
1023	1023	1023				
1023	1023					
1023						
1015	1003	907	956	992	910	959
1010	1034	989	1028	896	905	
992	903	1040	944	941		
1030	1031	981	990			
909	973	1017				
953	989					
944						

APPENDIX 9

IU2000 IMAGE POST-PROCESSING SOFTWARE

The **PTL IU2000** software is an off-line utility for producing enhanced and improved 2-D images from capacitance data files created by **PTL ECT** systems.

CONTENTS

4. IMAGE VIEWER SOFTWARE

FIGURES

A9.1 INTRODUCTION

A9.1.0 OVERVIEW

The **PTL IU2000** image processing software complements the standard **operating software** and allows **captured ECT capacitance** data to be processed to produce **enhanced image** and other data files. **Images** can be constructed **from capacitance data files** (**single** or **twin-plane** and for all **previous** PTL data formats) using either **Linear Back-projection** or the **fast iterative algorithm** described **in PTL Application Note AN4**. **Output data files** for **capacitances**, **image**s or **volume ratio** can be generated in a number of **ASCII** and/or **binary** formats. A set of **sample capacitance data files** are included with the software.

For a simple introduction to the use of the **IU2000** software, please refer to **section 2 (Quickstart Instructions)**.

A9.1.1 SOFTWARE FEATURES

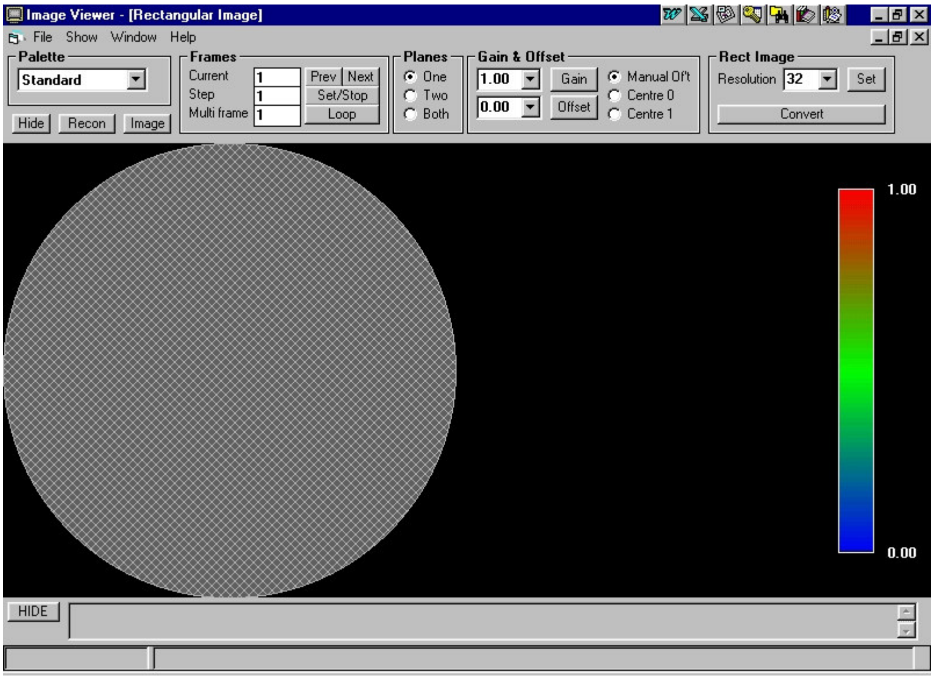


Figure A9.1. Opening Image Viewer Window

The **IU2000** software is based around an **Image Viewer** window (**Image**) (and an **Image reconstructor** window (**Recon**).

The **Image Viewer** window (shown in figure A9.1 above) provides **adjustable image permittivity gain** and **offsets**, choice of **display scales**, **multiple** or **sequential image display**, **image printing** and **file export facilities** etc.

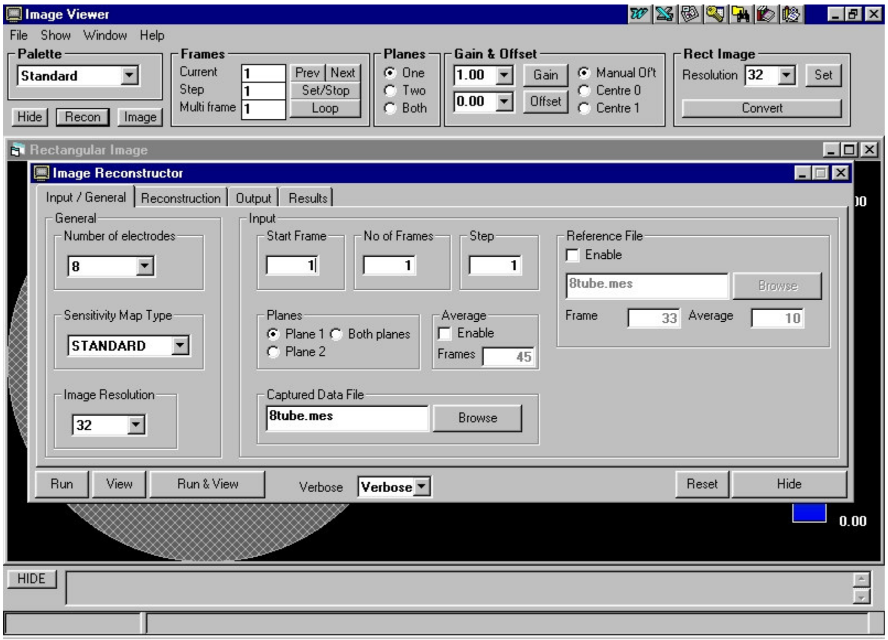


Figure A9.2 Image Reconstructor Window (Input/General Tab)

The **Image Reconstructor** Window, shown in figure A9.2 above, converts **captured capacitance data files** of the form **data.mes** (**PCECT** format), **data.cap** (**TransECT** format) or **data.bcp** (**ECT32** format) to a range of **output file formats,** using either a standard **Linear back-projection (LBP) algorithm** or a **fast iterative algorithm**. The **iterative algorithm** allows the image at each iterative step to be viewed.

Any number of **consecutive** or **non-consecutive captured capacitance data frames** can be converted into **image** or other **output data files,** starting with **any image frame** in the sequence of captured data. A choice of **concentration/permittivity** models is provided to give **improved accuracy and sensitivity** for many typical applications. Additional facilities include the provision for viewing **twin-plane image files,** the generation of **voidage (volume ratio) files, image averaging** and **correction** using a **reference file.**

The **Image Viewer** window can be accessed directly from the **Image Reconstructor** window, or vice versa, by the use of the **Image** and **Recon** buttons on the **toolbar** in the **Image Viewer** window.

The detailed facilities provided in the **Image Reconstructor** and **Image Viewer** windows are listed in the following two sections.

A9.1.2 IMAGE RECONSTRUCTOR WINDOW

The **Image Reconstructor** window includes the following options:

A9.1.2.1 Input parameter options:

Frame averaging for minimising the noise level of a stationary image.

Reference file selection: Selection of a capacitance data file frame which can be subtracted from all frames of the same or other data files. This facility can be used to compensate for the effects of drift, or to display changes from a set experimental condition. The reference frame can be an averaged data set.

A9.1.2.2 Reconstruction options:

Choice of **single-step (LBP)** or **iterative** algorithm for image reconstruction.
Optional use of **image pixel trunction** between user-defined limits.
Option to store and display **intermediate frames** in **iterative** algorithm.
Option to set **feedback gain** and **limit** parameters in **iterative** algorithm.
Choice of **parallel**, **series** or **Maxwell concentration/permittivity** models.

A9.1.2.3 Output file options:

Image file in (raw) ASCII format (**.img**)
Multiple image file in (raw) ASCII format (**.img**)
Image file in (new) ASCII format (**.aid**)
Image file in binary format (**.bid**)
PPM image (**.ppm**)
Volume ratio file (ASCII) (**.vr**)
Normalised capacitance file (ASCII) (**.ncp**)

A9.1.2.4 Function options

Reset button resets most input parameters to default values
Verbose option box determines detail level of output result data
Hide button hides **Image Reconstructor window**. (**Recon button** in **Image Viewer window** restores the **Image Reconstrucor** window).
Run and View button converts data files and displays images.
Run button carries out file conversion only.
View button allows converted files to be viewed.

A9.1.3 IMAGE VIEWER WINDOW

The **Image Viewer** window includes the following facilities:

A9.1.3.1 Menu options

A9.1.3.1.1. File Menu Options

Load and save Image files in rectangular pixel format.
Paste image to windows clipboard.
Print on-screen image

A9.1.3.1.2 Show Menu Options

Display normalised capacitance measurements in graphical or text format

A9.1.3.2 Toolbar options

A9.1.3.2.1 Colour palette options

Standard (RGB)
Red/blue
Grey
Electromagnetic (multi-colour)

A9.1.3.2.2 Frame Options
Single or multiple frame display
Continuous sequential frame display
Step/skip frames

A9.1.3.2.3 Planes Options
Display plane 1, plane 2 or both planes.

A9.1.3.2.4 Gain and Offset options
Set pixel gain
Set permittivity offset parameter
Manual offset mode
Automatic offset mode to display image centered on high permittivity value.
Automatic offset mode to display image centered on low permittivity value.

A9.1.3.2.5. Image Resolution
Select Image Resolution (Currently fixed at 32X32)

A9.1.3.3. Image Options
Display pixel permittivity using mouse cursor.
Display one frame of a multiple frame display using mouse cursor.
Display first and last frames using Home and End keys.
Display image parameters, including voidage, using mouse key.

A9.1.3.4 Scroll bar
Displays detailed information about image file

A9.1.3.5 Information bar
Displays information about currently selected image.

A9.1.4 SOFTWARE INSTALLATION

The installation procedure for the **IU2000** software is straightforward and is described in **Appendix 7.**

A9.1.5 LOCATION OF DATA FILES

As the IU2000 utilities operate on stored data files it is important to know where these files are located. The ECT file structure is described in **section 5**.

A9.1.6 SAMPLE DATA FILES

The following data files are included with the software and are located in the **c:\IU2000\imgfiles directory:**

All of the files were created using 8-electrode ECT sensors

8rod.mes	Single plane file for a 53mm diameter dielectric rod located inside a 100mm diameter sensor. (Static image)
8tube.mes	Single plane file for a 53mm diameter tube located inside a 100 mm diameter sensor. (Static image)
8fb.me2	Twin-plane file for a 60mm diameter fluidised bed containing sand. (dynamic image)

A9.2 QUICKSTART INSTRUCTIONS

These instructions provide a fast and simple introduction to the IU2000 software. They describe how to convert an input capacitance data file (eg one of the sample files provided as detailed in section A9.1.6) into an output image file using the Image Reconstructor window and to view the converted file using the Image Viewer window, using both the Linear Back-Projection and iterative algorithms.

New users should follow the instructions in this section to familiarise themselves with the general principles of the software and then read the rest of the manual in detail for information about the software features not covered in the **Quickstart** section.

2.1 IU2000 PROGRAM INITIALISATION

1. Double click on the **IU2000 icon** on the **Windows Desktop**. The **Image Viewer** window shown in **figure A9.1** will appear.

2. Click on the **Recon button** at the left hand end of the **Toolbar.** The **Image reconstructor** window will appear superimposed on top of the **Image Viewer** window as shown in **figure A9.3**.

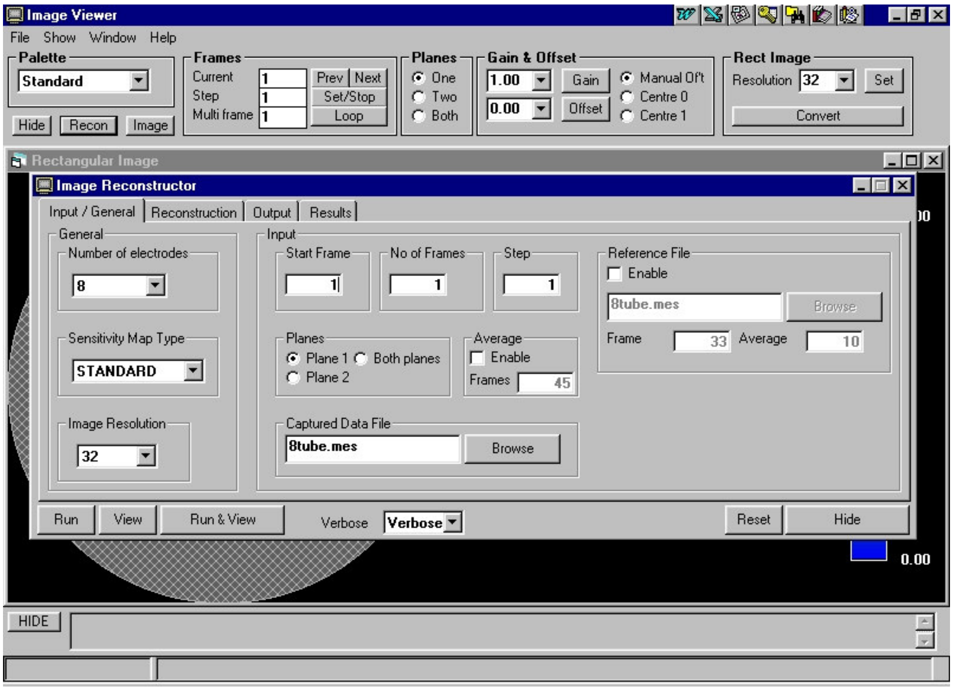


Figure A9.3 Image Reconstructor Window (Input/General Tab)

3. Click on the **Reset button** at the bottom RH of the **Image Reconstructor** window. This will reset many of the optional parameters to the **default** values

4. In the **General** data area, set the **Number of electrodes** to the value appropriate to **the captured data file** to be processed. The **sample data files** included with the **IU2000** software were generated using an **8-electrode** sensor.

5. In the **Input data area**, use the **Browse button** to select a **single plane capacitance data file** to be converted. (The **sample** data files are in the **C:\iu2000\imgfiles** directory).

A9.2.2 IMAGE GENERATION USING LINEAR BACK-PROJECTION (LBP)

This section explains how to use simple **linear back-projection** to obtain the image file.

1. Click on the **Run and View** button at the bottom of the **Image Reconstructor** window. The **first frame** of the **input data file** will be converted to an **image file** using **Linear Back-Projection** and displayed as shown in **figure A9.4 below**.

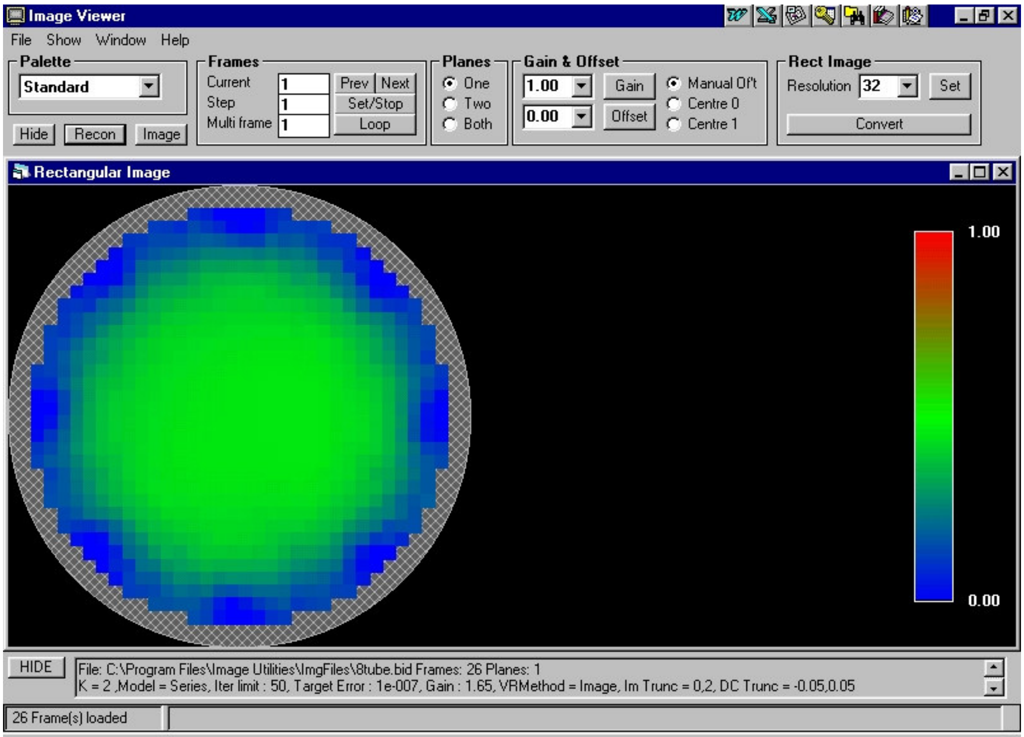


Figure A9.4. Image Viewer window. Single-Plane Image

2. Click the **mouse cursor** inside the **image**. The **permittivit**y of the **pixel** will be displayed on the **permittivity scale** at the **RHS** of the window.

3. Click on the **Recon** button on the **toolbar**. The **Image Reconstructor** window will again be displayed.

A9.2.3 IMAGE GENERATION USING AN ITERATIVE ALGORITHM

This section explains how to use the **same input data** to reconstruct an **improved image** using an **iterative algorithm**.

1. Click the **Reconstruction** tab. The window shown in **figure A9.5** will appear.

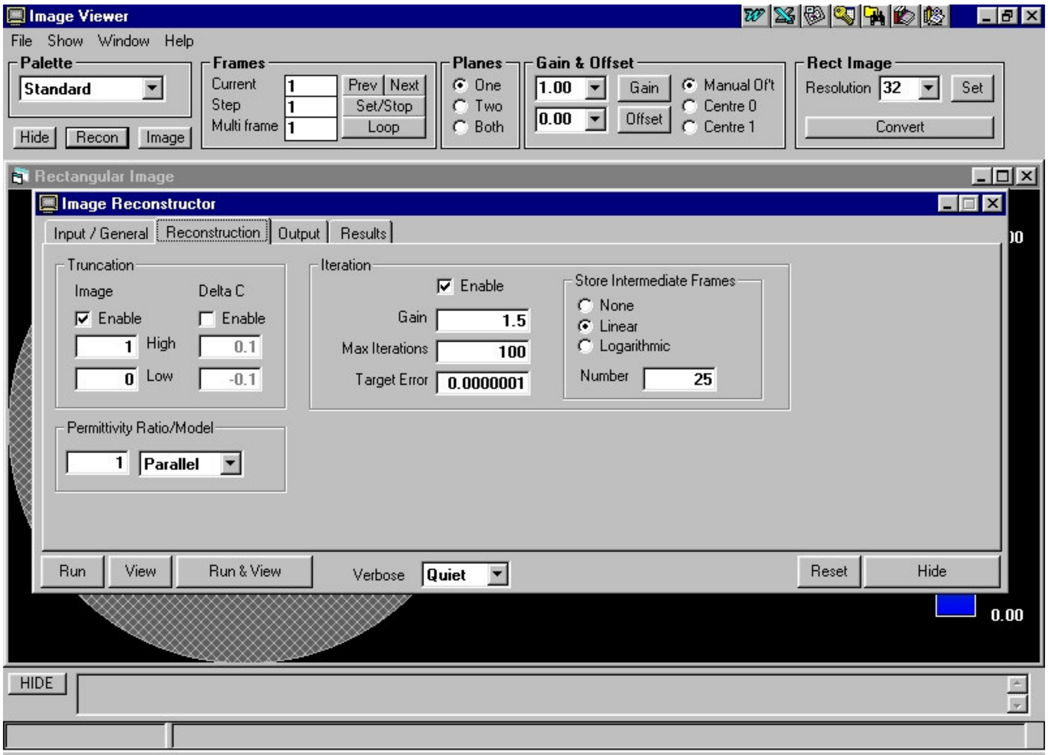


Figure A9.5 Image Reconstructor Window (Reconstruction Tab)

2. Enable the **Image** and **DeltaC** parameters in the **truncation** box.

3. Set the **image truncation high** and **low** limits to **1** and **0** respectively.

4. Set the **DeltaC** values to 0.05.

5. **Enable** the **iteration parameters** in the **iteration** box.

6. Set the **Max iterations** parameter to 100.

7. Select the **Linear** option in the **Store Intermediate Frames** box and set the **number** to 25.

8. Click the **Run and View** button. The file will be converted and the first frame (obtained using **LBP**) will be displayed once the output file has finished loading into the **Image Viewer** window.

9. Click the **Loop** button on the **toolbar**. The screen will show successive iterations (every fourth frame), up to the 100th iteration.

10 Click the **Set/Stop** button.

11. Click the **mouse cursor** inside the **image** and press the **Home** key on the **PC keyboard**. The **displayed frame** will now be the **first frame** in the **output file**.

12. Set the **Multi frame** box to **5** and click on the **Set/Stop** button in the **Frames** box. The 25 images will be displayed sequentially as a **5 X 5 block** as shown in **figure A9.6**.

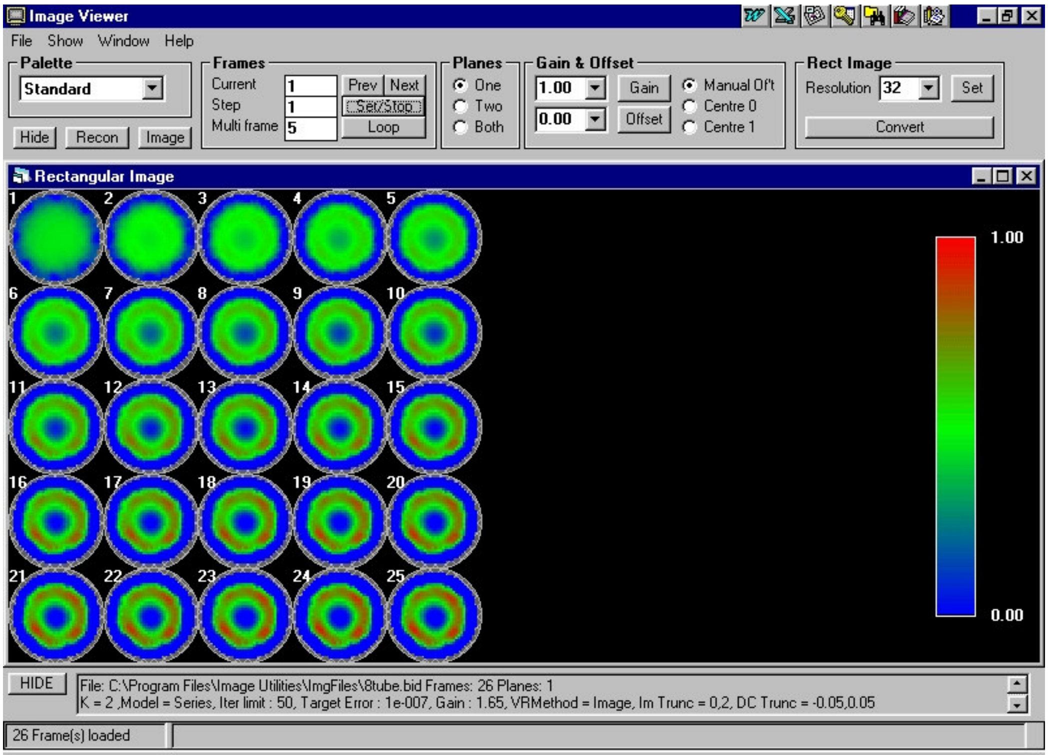


Figure A9.6 Image Viewer window. Multiple Single-plane Images

13. Double-click one of the image frames. The selected frame will be displayed as a **single** image.

14. Click the **Set/Stop** button. The window will revert to the **multi-frame display format.**

A9.2.4 SWITCHING BETWEEN IMAGE RECONSTRUCTOR AND IMAGE VIEWER WINDOWS

It is possible to switch between the **Image Viewer** and **Image Reconstructor** windows at any time by clicking the mouse cursor on the **Image** or **Recon** buttons on the Image Viewer **toolbar**.

A9.2.5 FURTHER INFORMATION

Having completed the above exercises, new users should read this section in detail from the start to familiarise themselves with the full range of features in the **IU2000** software. In particular, **section A9.3** gives detailed information about the operation of the **Image Reconstructor** window and **section A9.4** gives similar information about the **Image Viewer** window.

A9.3. IMAGE RECONSTRUCTOR SOFTWARE

This section gives full details of the operation of the **Image Reconstructor** Window, which uses either **Linear Back-projection** or an **iterative algorithm** to convert **normalised inter-electrode capacitance measurements** into **permittivity or concentration images**. Both methods are described in detail in a **PTL Application Note AN4**. In the following instructions it is assumed that a suitable **capacitance data file** has already been captured and saved using the standard ECT operating software. Otherwise, one of the **sample data files** supplied with the software (see section A9.1.6) can be used.

A9.3.1 INITIATING THE SOFTWARE

1. Double click on the **IU2000 icon** on the **Windows Desktop**. The **Image Viewer** window. shown in **figure A.9.1** will appear.

2. Click on the **Recon button** at the left hand end of the **Toolbar.** The **Image reconstructor** window will appear superimposed on top of the **Image Viewer** window as shown in **figure A9.2**.

3. Click on the **Reset button** at the bottom RH of the **Image Reconstructor** window. This will reset many of the optional parameters to the **default** values. These **default values** are indicated inside square [ ] brackets, where applicable in the following sections.

A9.3.2 SETTING THE INPUT PARAMETERS

The **Input parameters** specify the **captured data** which is to be converted into **image** or other data files. There are two sets of parameters, **compulsory** and **optional**:

A9.3.2.1 Compulsory parameters

The **Image Reconstructor** window should display the parameter screen for the **Input/General** tag as shown in **figure A9.2**. If this is not the case, click on the **Input/General** tab in the **Image Reconstructor** window to display the input parameters. Set the parameters as follows:

A9.3.2.1.1 General data area

1. **Number of electrodes:** Set to the value appropriate to **the captured data file** to be processed,
2. **Sensitivity map type:** Select the sensitivity map type to be used to construct the image (either **[STANDARD]** or **WATER**)
3. **Image resolution:** Set to match the **sensitivity map** [**32 pixels**].

A9.3.2.1.2 Input data area

1. Use the **Browse button** to select the required **captured capacitance data file.**
2. Set the **number** of the **first frame** to be converted in the **start frame** box **[1]**, set the total number of frames to be converted in the **No of Frames** box **[1]** and set the **step value** (=1 to convert every frame) in the **Ste**p box. [1]
3. Select the **planes to be converted** in the **Planes** box **[Plane 1].**

These are all the **compulsory parameters** which must be set in the **Input/General** data section of the **Image Reconstructor** window.

A9.3.2.2 Optional parameters

The function of the **optional parameters** is as follows:

A9.3.2.2.1 Averaging parameters

If the **Average box** is **enabled**, **[Not enabled]** the **input data** will be **averaged** over **the number of frames specified** in the **Frames** box on a rolling average basis. The averaging starts at the **start frame** number and so the **number of input frames** selected must be at least equal to the number of frames set in the **Average Frames** box. The averaging option can be a useful feature for minimising noise in a slowly-varying image.

A9.3.2.2.2 Reference file parameters

If the **Reference File** box is **enabled**, **[Not enabled]** the **data frame** specified in the **Reference file** box will be subtracted from all of the **captured data frames** specified in the Input box. This can be a useful feature, either for **minimising the effects of measurement drift**, or for **viewing changes from a given physical state**.

A9.3.3 SETTING THE RECONSTRUCTION PARAMETERS

The **Reconstruction** parameters determine how the **captured capacitance** data is converted into **image frames**. The image reconstruction method is described in detail in PTL **application note AN4**, which should be consulted for further information. The terminology used here follows that in the application note.

Click on the **Reconstruction tab** in the **Image Reconstructor** window, when the modified window shown in **figure A9.3** should appear.

A9.3.3.1 Choice of Image Reconstruction method

The **Image Reconstructor** window can convert **capacitance data** into **image pixel values** using either a very simple and approximate **Linear Back-Projection (LBP)** algorithm, or the more accurate **iterative** algorithm described in **AN4**.

The method used to reconstruct the image depends on whether the **Iteration** box is **enabled**. If it is **not enabled**, (the default situation following operation of the **Reset** button), the method used is **LBP**. If it is **enabled**, the **iterative** method is used.

To carry out **image reconstruction** using **LBP without image truncation**, ensure that the **Iteration** box is **not enabled**, then proceed directly to section A9.3.3.4.

To carry out iterative reconstruction, proceed as follows:

A9.3.3.2 Truncation parameters

Input the required image **truncation** and **Delta C** values in the **Truncation** boxes. If **Image truncation** is **enabled**, **[not enabled]** the **image pixels** will be truncated to lie between **the normalised pixel limits** set in the **high** and **low image truncation boxes**. Typical values for situations where the pixels are known to lie between the values of 0 and 1 would be (High = 1) and (Low = 0).

If the **Delta C option** is **enabled**, **[not enabled]** the **capacitance values** used to correct the image (**ΔC**) will be limited to the values set in the **Delta C boxes**. Typical values are (**±0.05**).

A9.3.3.3 Iteration parameters

1. To reconstruct images using simple **Linear Back-projection (LBP)**, set **the Iteration Enable** tick box to **not enabled**.

2. To construct images using the **iterative** method, **enable** the parameters in **the iteration data area** and set them as follows:

> **Gain**: This is the feedback gain used in the iterative algorithm. Typical "safe values" are between 1 and 1.6. Values greater than 1.6 may cause the image reconstruction solution to **diverge** rather than **converge**. The suggested (default) value is **[1.5**].
>
> **Max iterations**: This is the number of iterations used to produce the final image. Note that [0] means that **no iterations** are used and the image will be obtained **using simple Linear Back-projection**. Suggested values range from 1 to 500 depending on the rate of convergence.
>
> **Target error**: This is the **limit** set for the **rms** value of the **capacitance errors** (**caperror)** in the **iterative algorithm**. Further iteration will cease if the calculated value of **caperror** is less than the **set value of target error**. This can be a useful method for terminating the algorithm when it has effectively ceased to converge. However, if early termination of the algorithm is not desired, a value of 0.0000001 should be used.
>
> **Store Intermediate Frames parameters**
> **The Store intermediate Frames parameter box** allows **intermediate image frames** to be generated and stored for subsequent **display** in the **output data file**. This facility is [**disabled]** if the [**None]** option is selected.
>
> If either the **Linear** or **Logarithmic** options are selected, **intermediate image frames** will be generated.The **number of intermediate frames stored** is set in the **Number** box and must not exceed the number of frames set in the **Max Iterations** box. As an example, if the **Max Iterations** box is set to **100** and the **Store Intermediate Frames number** box is set to **25**, every fourth frame (**100/25**) will be stored in **the output data file** if the **Linear** option is selected. If the **Logarithmic** option is selected, the number of each stored frame is determined on a logarithmic basis.

A9.3.3.4 Permittivity Ratio/Model

In the **Permittivity Ratio/Model box**, select the required **Permittivity Ratio and Permittivity Model**. Details of permittivity models are given in **PTL Application notes AN2** and **AN4**. The **default** model option is [**Parallel],** in which case the **permittivity ratio** is **ignored**. For the **Series** and **Maxwell** models, enter the **permittivity ratio of the two materials with which the sensor was calibrated.** Typical values are 1.6 for plastic beads/air and 3.1 for glass beads/air.

A9.3.4 OUTPUT PARAMETERS

The output parameters define the **output file name** and **type**.

Click on the **Output tab** in the **Image Reconstructor** window. The window shown **in figure A9.7** will appear.

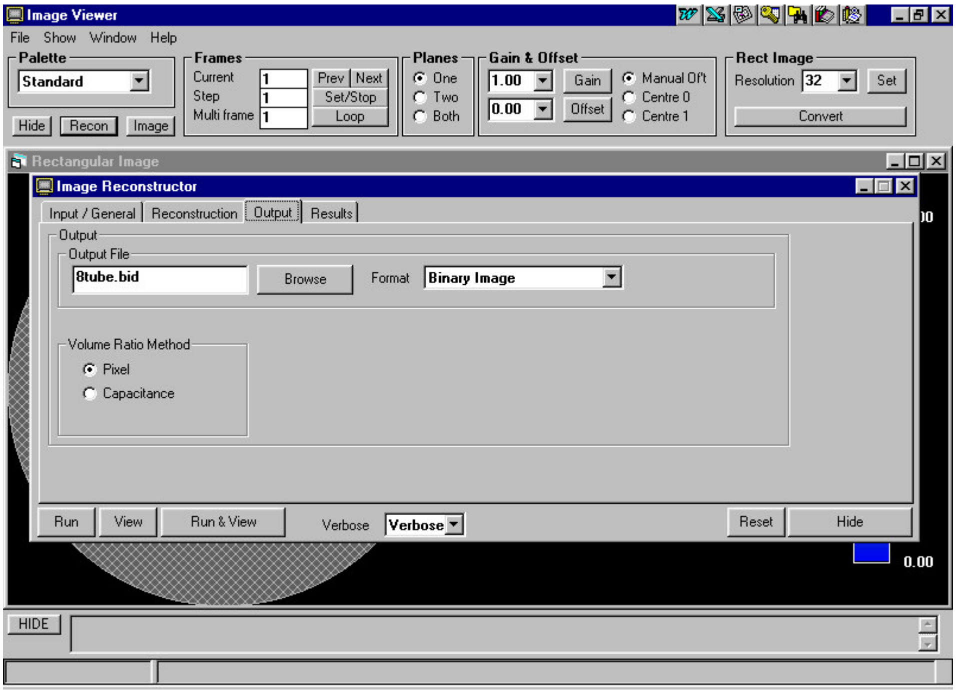


Figure A9.7 Image Reconstructor Window (Output Tab)

A9.3.4.1 Output file name and type

The **output file name** defaults to the [**input filename]** with a **new file extension** unless this is deliberately changed in the **Output File** box.

The **output file type** should be selected using the **Format** box. A list of **available file types** is given in section A9.3.4.2 below. The default file type is **Binary image (.bid)** as this stores the image data in a compact format.

A9.3.4.2 Output file types

The following **output file formats** are available:

A9.3.4.2.1 Image file in (raw) ASCII format (.img)
This option will generate image files in ASCII format, similar to those generated by the previous **PTL Image Utilities Software**. These files can be viewed using the **Image Viewer** software.

A9.3.4.2.2 Multiple image file in (raw) ASCII format (.img)
This option will generate image files in ASCII format, similar to those generated by the previous **Image Utilities Software**, **but with a unique filename** for each **image,** of the form **filename_imagenumber.** These image files **can be viewed** individually using the **Image Viewer** software.

A9.3.4.2.3 Image file in (new) ASCII format (.aid)
This option will generate image files in ASCII format, but in a format similar to that defined for **binary** image files**.** These files **cannot be viewed** using the **Image Viewer** software.

A9.3.4.2.4 Image file in binary format [.bid]
This (**default**) option produces image files in **binary** format. These files can be viewed using the **Image Viewer** software.

A9.3.4.2.5 Image file in PPM format (.ppm)
This option produces ASCII image files encoded in **RGB** format as defined in the **P**ortable **P**ix **M**ap standard. These files **cannot be viewed** using the **Image Viewer** software.

A9.3.4.2.6 Volume ratio file (ASCII) (.vr)
This option produces **Volume ratio** files in ASCII format**.** These files **cannot be viewed** using the **Image Viewer** software.

A9.3.4.2.7 Normalised capacitance file (ASCII) (.ncp)
This option converts the **normalised binary capacitance** data into normalised capacitance data in **ASCII** format. These files **cannot be viewed** using the **Image Viewer** software.

Details of all of these **file formats** are given in section A9.7.

A9.3.4.3 Volume Ratio Method
The option selected in this box determines whether the **volume ratio** is calculated from the image **[Pixels]** or from the **Capacitance** measurements.

A9.3.5 RESULTS TAB

Detailed information about the converted data can be viewed in this screen once the data has been converted. Click on the **Results tab** to view this data.

A9.3.6 OTHER CONTROLS

Once the parameters defined in sections 3.2 to 3.4 have been defined, the output file can be generated using the buttons at the bottom of the **Image Reconstructor** window as follows:

A9.3.6.1 Run button

Clicking on the **Run** button will generate the required **output file**. While the data is being generated, information is displayed on the PC screen and this information can be viewed in summary form by clicking on the **Results** tab.

A9.3.6.2 View button

If the output file generated by the **Run** button is compatible with the **Image Viewer** software, it can be viewed by clicking on the **View** button. Details of the **Image Viewer** software are given in section A9.4.

A9.3.6.3 Run and View Button

This button combines the functions of both the **Run** and **View** buttons. Clicking this button will automatically generate the **output data file** and load this data into the **Image Viewer** window.

A9.3.6.4 Verbosity level box

The option selected in the **Verbose** box determines the amount of incidental data which is generated to accompany the output file. The normal (default) setting is **Verbose**.

A9.3.6.5 Reset button

Clicking this button resets all of the input data to the default values where these are defined.

A9.3.6.6 Hide Button

Clicking this button hides the **Image Reconstructor** window. It can be retrieved by clicking on the **Recon** button in the **Image Viewer** window.

A9.3.7 OUTPUT MESSAGES

While file conversion is in progress following use of the **Run** or **Run and View** buttons, detailed **output messages** are generated. One **important message** which can occur **when iterative reconstruction** is being used concerns the **convergence** or otherwise of the **image reconstruction** process.

If the **rms capacitance error** between **consecutive image frames increases** rather than d**ecreases**, the message **"Warning - Divergence"** is displayed during file conversion. At the end of the file conversion process, the summary message **"One or more frames diverged, view image?"** appears. This is a **warning** that the **image reconstruction process has failed**, usually because an **excessively large gain figure** has been set in the **Reconstruction parameters** window. The **cure** is to **re-run the software** using a **lower gain figure** (eg 1.5) or to lower the limits set in the **ΔC** boxes.

A9.4 IMAGE VIEWER SOFTWARE

Once an output file has been generated using either the **Run** or **Run and View** buttons in the **Image Reconstructor** window, the output data can be viewed and modified using the **Image Viewer** window, provided that the output file format is compatible with the **Image Viewer** (see section A9.3.4). The **Image Viewer** window can also be used to view **image files** created by other software, provided that they are in a compatible format (see section A9.4.2).

The **Image Viewer** window can be used to view images in either a rectangular or triangular grid format. All of the data files produced by **PTL ECT** systems are in rectangular format and this option should normally be chosen when viewing files. Facilities included include **adjustable image gain, offset, choice of display scales, number of pixels, multiple or sequential image display, image printing and file export facilities**, etc.

If the **Run and View** button was used, the **Image Viewer** window is activated automatically. If the **Run** button was used to generate the output data file, the **View** button should be used to activate the **Image Viewer** window.

To view output files in ASCII format which are not compatible with the **Image Viewer** software, a suitable **ASCII text editor** such as **Notepad** or **Word** should be used as described in section A9.4.2.

A9.4.1 PROGRAM INITIALISATION

The **Image Viewer** window is the default window when the **IU200** software is initiated. Previously converted data files can be viewed immediately by opening them using the **File menu** (see section A9.4.2).

If the **Image Reconstructor** window has been used, the output data file will be loaded automatically into the **Image Viewer** window if the **Run and View** button was used to generate the output file.

If the **Run** button was used, the output file can be viewed using the **Image Viewer** window by clicking on the **View** button.

It is possible to alternate between the **Image Viewer** and **Image Reconstructor** windows at any time simply by clicking on the **Recon** or **Imag**e buttons on the **toolbar** in the **Image Viewer** window.

If the **Run and View** button has been used in the **Image Reconstructor** window, please skip the next section and go directly to section A9.4.3

A9.4.2 LOADING IMAGE FILES IN RECTANGULAR PIXEL FORMAT

As well as displaying **image files** generated by the **Image Reconstructor** window, the **Image Viewer** window can display **image files** in ASCII format produced by **PTL operating software, including PCECT, TransECT** and **ECT32** software. However, image files produced by most of these programs contain a **frame time stamp** at the start of each frame. This time stamp must be removed before the image data can be read correctly by the **Image Viewer** window. This can be done manually using a text editor.

Before loading an image file, the **image resolution** (number of pixels across the screen) must be set correctly in the **Rect Image Resolution** box. Once the correct value has been selected, it must be confirmed using the **SET** button. The resolution must be set to the same value as that used when the original file was saved (normally 32 in the standard ECT software).

To load a rectangular format file:

1. Set the **pixel resolution as described above.**

2. Click on **File** in the **Menu Bar**.

3. Click on **Load Rectangular Image(s)** in the **drop down menu**. A file menu will appear.

4. Find the file to be loaded and click on the **OK** button. The image file will be loaded and the first image in the sequence will be displayed on the screen.

A9.4.3 MANIPULATING IMAGE FILES

The Image Viewer window provides a number of facilities for viewing and modifying images. A typical **single frame single-plane image** is shown in **figure A9.4**. If **a single-frame twin-plane image file** is displayed, the images for the **two planes** will appear **side-by-side** as shown in **figure A9.8**.

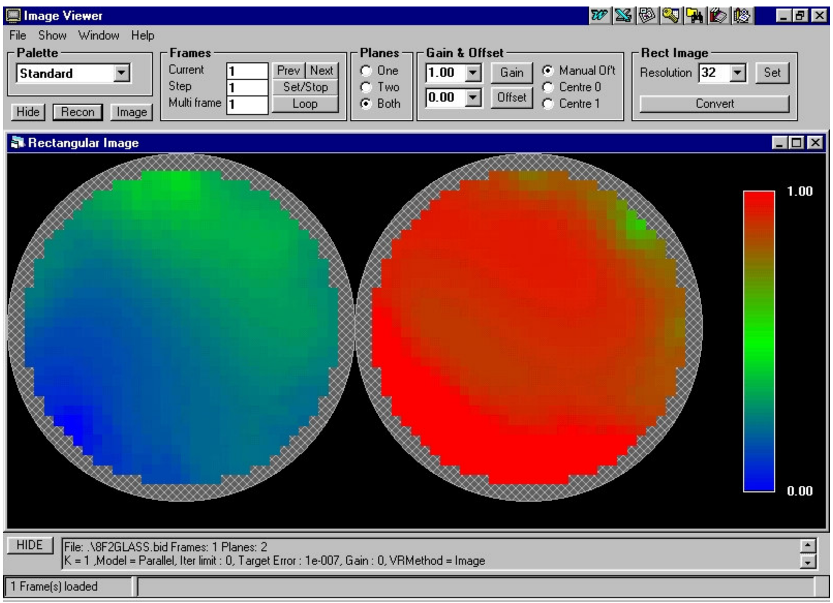


Figure A9.8. Image Viewer window. Twin-Plane Image

A9.4.3.1 Image gain

The Image Gain factor G applied to each of the image pixels has a default value of 1. However, this can be changed by adjusting the figure in the GAIN box. However, if a value of G other than 1 is entered, the permittivity value of each pixel is multiplied by the Image gain factor G. The effect is to change the overall gain of all of the image pixels. The values on the permittivity colour scale change automatically to reflect the effect of the new gain setting.

A9.4.3.2 Permittivity offset

The **Permittivity offset** parameter **OS** is normally set to zero, in which case the **displayed image** covers the **normalised permittivity range** from 0 to 1. If the **Manual Of't** (Offset) option is enabled in the **Gain and Offset** box, an offset value other than zero can be selected or entered (eg **OS**), in the range 0 to 1. In this case, **the permittivity range displayed** will be from **OS to (1 + OS)**. The effect of this is to **offset** the displayed **measurement range**. This facility can be used, for example, to display permittivity values which exceed the nominal maximum value of 1.

A9.4.3.3 Combined gain and offset

If both **Image gain** (**G**) and **Permittivity offset** parameters (**OS**) are in use and the **Manual Of't** option is **enabled**, the **permittivity scale** will be modified to run from **OS** to (**OS + 1/G).** This facility can be very useful for monitoring small permittivity changes.

If the **Centre 0** option is enabled, the **offset parameter** will be automatically adjusted, taking into account the **Gain** figure, so that the **permittvity scale is centered around the low level normalised permittivity values (0)**. Conversely, if the **Centre 1** option is enabled, the **offset parameter** will be automatically adjusted so that **the permittivity scale is centered around the high level normalised permittvity value (1)**.

A9.4.4. IMAGE DISPLAY FORMAT

A9.4.4.1 Information and scroll bars

The **number of images** in the selected image file is shown in a box on the **information bar** at the **LHS bottom** of the **Image Viewer** window.

The **volume ratio** of the current displayed image (calculated from the image pixels) can be dislayed in a **Frame data box** at the **RHS** of the **information bar** by **clicking the mouse pointer inside the image**.

A **scroll bar** immediately above the **information bar** displays detailed information about the **current data file**. This information can be viewed in full by the use of the **scroll arrows** at the **RHS** of the **scroll bar**.

A9.4.4.2 Frames display format

The parameters in the **Frames box** on the **toolba**r determine how the images are displayed**.**

Images can be displayed in sets of **P X P** images on the screen where **P** can be any integer. The default value is 1 but can be changed by setting a new value in the **Multiframe** box.

The **number** of the first image to be displayed (counting the first frame to be loaded as 1) is set in the **Current frame** box.

The **images** are displayed **consecutively** from the **initial frame** in steps of **Q** images, depending on the value of **Q** set in the **STEP** box. An example of a multiple image display screen is shown in **figure A9.6** for a **5 X 5** image format. A set of images for a **twin-plane** file in **3 X 3** format are shown in **figure A9.9**.

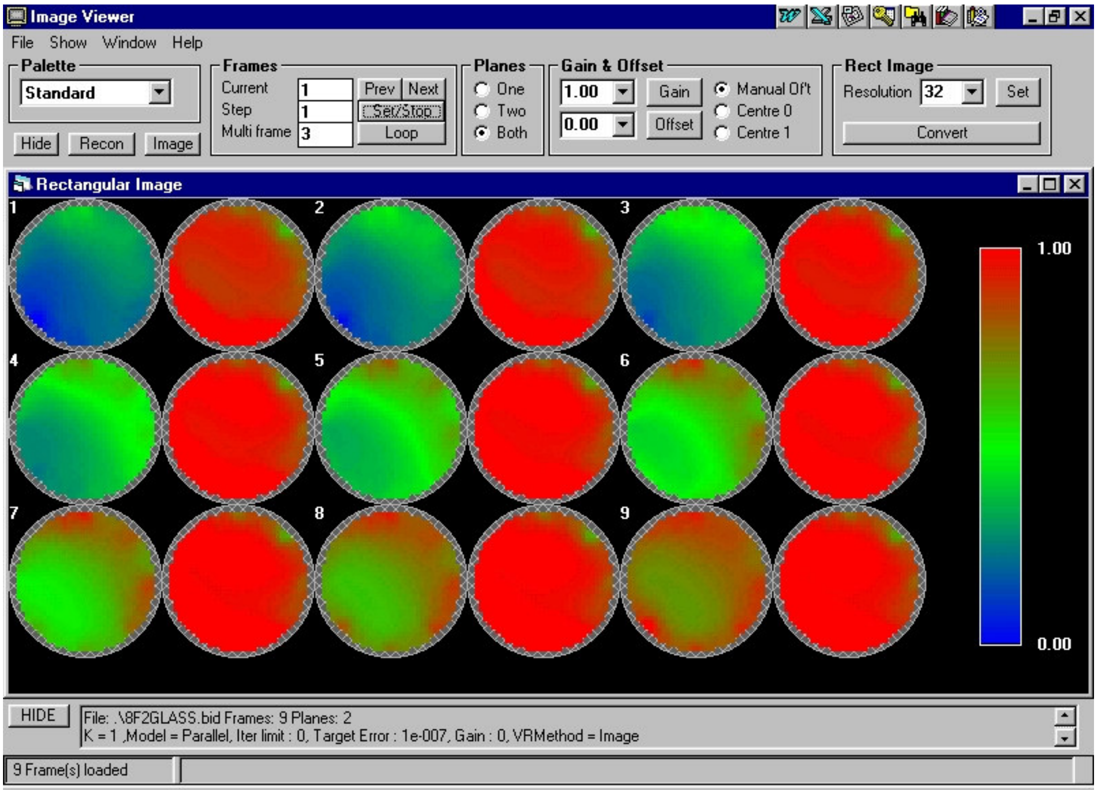


Figure A9.9 Image Viewer window. Multiple Twin-plane Images

If the **Loop** button is clicked, images will be displayed **sequentially**, starting at the current frame, on a cyclical basis until the **Set/stop** button is clicked.

A9.4.4.3 Image colour scales

Images can be displayed in a number of different colour scales depending on the scheme chosen from the **Palette** box. The standard colour scheme matches that used in the original **PCECT** and **TransECT** software. Other options include **Red/Blue**, **Grey** and **Electromagnetic**.

A9.4.4.4. Mouse Cursor functions

The **Mouse cursor** has a number of functions in the **Image Viewer** window.

If the **mouse cursor** is clicked **inside a single image**, the **normalised permittivity** of the pixel is displayed on the **permittivity scale** at the RHS of the window. Information about the image frame (including the **Volume Ratio**) is also displayed on the **information bar** at the bottom of the window when the mouse cursor is clicked inside the image.

If **multiple images** are displayed, clicking the mouse pointer inside one of the images will display the parameters for that image, including the pixel permittivity.

If the **mouse pointer** is **double-clicked** inside one image frame, the frame will be re-displayed as a **single image**. The **multiple image display** can be **restored** by clicking on the **Set/Stop** button in the **Frames** box on the **toolbar**.

A9.4.4.5 Use of Home, End, Page Up and Page Down keys

With the **Image Viewer** window displaying a **single image**, and with the mouse cursor clicked inside the image, pressing the **Home** key will display the **first image in the output file** and similarly pressing the **End** key will display the **last image in the file**.

Similarly, when a **multiple image** display is shown, and with the mouse cursor clicked inside one image, pressing the **Home** key will display **multiple images** starting with the **first image in the output file** and similarly pressing the **End** key will display **multiple images** starting with the **last image in the file**.

If the **Page Up** key is pressed, the **previous frame** is displayed. Similarly, if the **Page Down** key is pressed, the **following (next) frame** in the file is displayed.

A9.4.5 FILE MENU OPTIONS

The **File menu** 4 provides a number of options for saving data. Converted rectangular image files can be saved in conventional format. In addition, images can be saved in **bitmap** format as well as copied to the **Windows clipboard** for transfer to other programs.

Note that the **Load mesh** and **Load triangular images** options are not usable with **Rectangular** format data files.

The **image** on the screen can be **printed** directly by clicking on the **Print** option in **the File menu.** Options are available for **re-sizing** and **centering** the image and setting the **background colour**.

A9.4.6 SHOW MENU OPTIONS

A9.4.6.1 Menu Options

Clicking on the **Show** menu in the **menu bar** will produce a **drop down menu** with the following options:

Rect Image This is the default image display format and should not be changed.
Mesh Not applicable in IU2000.

Tri Image Not applicable in IU2000.

Tool Bar This option recalls the **tool bar** if it has been removed using the **HIDE** button.

File Info This option recalls the **scroll bar** file information if it has been removed using the **HIDE** button.

Measurements This displays the **normalised capacitance measurements** for the displayed data frame in either **graphical** or **text** format as described in section 4.6.2.

Reconstruction Window This displays the **Image Reconstructor** window.

A9.4.6.2 Show Measurements option

The **show measurements** option allows the **individual capacitance measurements** associated with the **selected image frame** to be viewed in a separate **Measurements** window. **Figure A9.10** shows the **Measurements** window which is displayed after selecting the **show measurements** option for a **single image frame**. The window can be re-sized to any convenient size but this may upset the data format when the **text tab** is used (see **Text tab** below).

The **Measurements window** has 3 Tab options:

Graphic tab. This displays the **normalised inter-electrode capacitance measurements** as a series of **vertical red lines** in the **conventional format** used in **all PTL operating software.** That is, the first line is C12, the second C13 etc, where 1 is the source electrode. The lines are in separated groups, determined by the number of the source electrode, so the second group contains C23, C24 etc. The lower horizontal black line is the 0 capacitance level and the upper line is the 1 level. The **displayed capacitances** are scaled (multiplied) by **the image gain** factor. The **identity** and the **unscaled value** of each individual measurement line can be checked by placing the **mouse cursor** on the required **red line**.

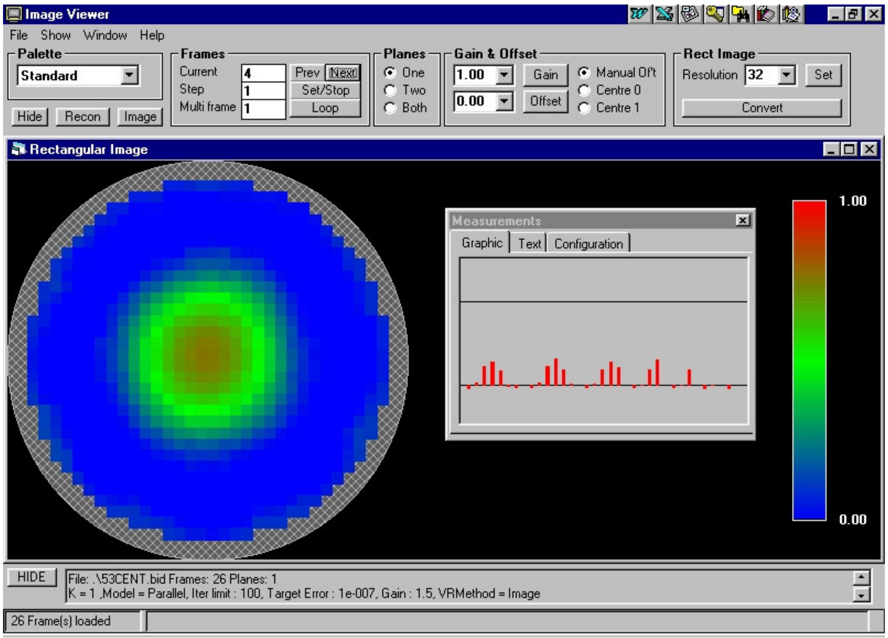


Figure A9.10 Image Viewer showing capacitances in graphical format

Text tab. This displays the (unscaled) **numerical** values of each normalised inter-electrode capacitance in ASCII format. The display mode conforms to the standard triangular format, so that the top row shows C12, C13 ..... C1N, the second row C23, C24, ...C2N etc.

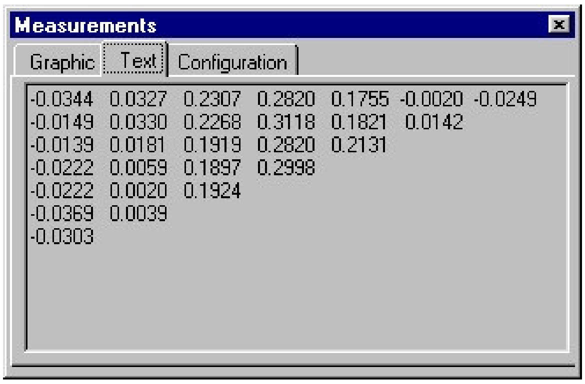


Figure A9.11 Capacitances displayed in text format

Configuration tab. This tab allows the **DeltaC feedback capacitance** values to be displayed for an image obtained using the **iterative algorithm.** These **capacitances** can be viewed, by setting the **DeltaC box** to be **enabled** in the **Configuration tab**. A scaling factor (**Graphic gain**) can also be applied to the **feedback capacitances**. The **DeltaC values** are the **differences** between the **capacitances back-calculated from the current image** and the **set of measured normalised capacitances** for the **current frame**. Note that **DeltaC** values will only be produced if **iterative** image reconstruction has been used.

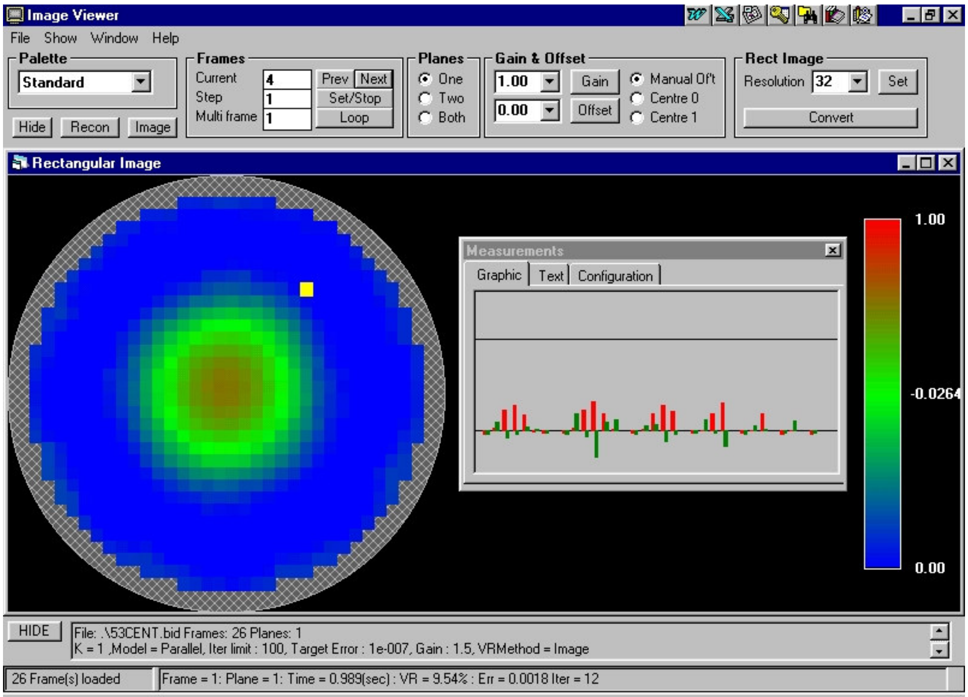


Figure A9.12 Capacitance and Feedback Measurements in graphical format

Figure A9.12 shows the **feedback capacitances** displayed as **green lines** for the same image as shown in figure 10. A **gain factor** of **10** has been applied in the **configuration tab** to the **feedback capacitances** to make them more visible. The **text tab** displays each **normalised capacitance measurement**, followed by the corresponding **DeltaC value in brackets**.

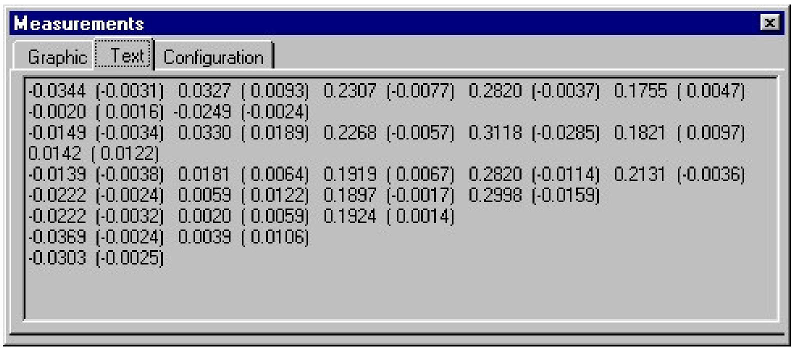


Figure A9.13 Capacitance and Feedback Measurements in text format

The **show measurements** option can also be used in **multiple image display mode** as shown in **figure A9.14**. The **capacitance** and **DeltaC** values can be viewed for **each image frame** by clicking the mouse cursor on **each frame in turn**. If the **iterative algorithm** has worked correctly, the **DeltaC** values will be **relatively large** for the first few frames and then will **reduce in value** with increasing iterations.

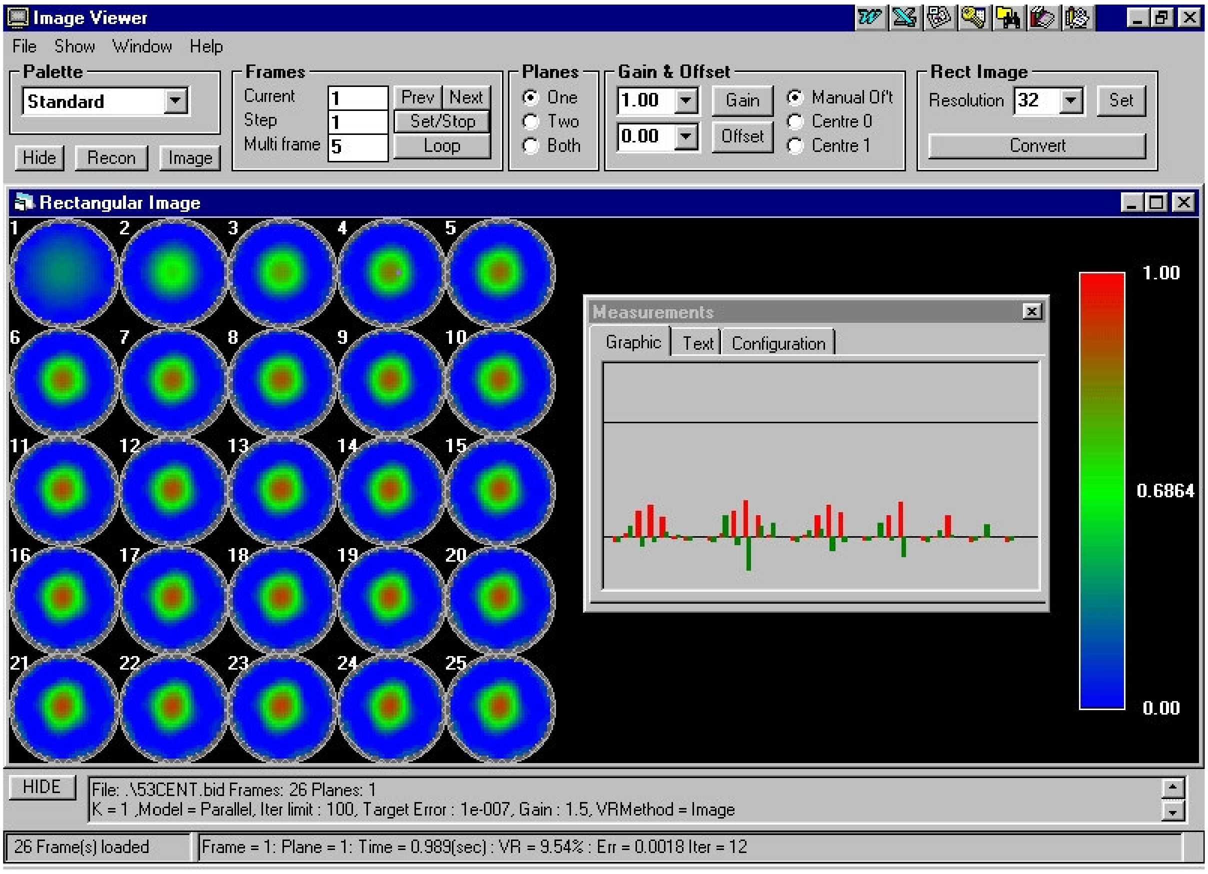


Figure A9.14. Multiple Images with Capacitance and feedback Measurements

A9.4.7 TOOLBAR BUTTONS

A9.4.7.1 Hide button

Clicking on the **HIDE** button will remove the tool bar from the screen and allows a larger image to be displayed. The tool bar can be restored by clicking on the Tool Bar heading in the **SHOW** drop down menu (see section 4.6)

A9.4.7.2 Recon Button

Clicking on this button will display the **Image Reconstructor** window

A9.4.7.3 Image button

Clicking on this button will display the **Image Viewer** window.

A9.5 DATA FILE LOCATIONS

A9.5.1 ECT32v2 FOLDER STRUCTURES

When the standard **PTL ECT software** is installed, the installation procedure sets up a set of directories and subdirectories for program and data files. The **ECT32v2 software** is installed in the directory **C:\ECT32v2. Capacitance data files** are normally stored in the directory **C:\ECT32v2\working**.

A9.5.2 SELECTING THE REQUIRED CAPTURED CAPACITANCE DATA FILE

When using the **IU2000** software, the required data file must be selected using the **Browse button.**

A9.5.3 LOCATION OF FILES CREATED BY IMAGE RECONSTRUCTOR WINDOW

The Image Reconstructor window automatically stores the image files in the same directory as the input data file unless a different directory is set by the user.

A9.6 SOFTWARE INSTALLATION

See Appendix 7 for software installation details.

A9.7 IU2000 OUTPUT FILE FORMATS

A9.7.1 VIEWING CONVERTED IMAGE FILES IN ASCII FORMAT

The simplest method for viewing the **converted image files** in ASCII format is to use a suitable W**ord Processor** or the **Notepad** or **Wordpad editors** included in the **Windows** operating system.

A9.7.2 OUTPUT FILE FORMATS

A.9.7.2.1 Image file in (raw) ASCII format (.img)

These are image files in ASCII format and can be viewed directly using the **Image Viewer** software. The files contain ASCII characters separated by **TAB** characters.

The image file format is is a grid of **32 X 32** numbers in **ASCII** format corresponding to the 32 X 32 pixels in the image. The numbers are separated by **TABS** to facilitate viewing. The numbers are normalised pixel values in the nominal range between 0 and 1. However, as the images are constructed using a simple linear back-projection algorithm, actual values may lie outside these limits because of inaccuracies in the image reconstruction algorithm.

A9.7.2.2 Multiple image file in (raw) ASCII format (.img)

These are image files in ASCII format, similar to those generated by the previous **Image Utilities Software**, but with a **unique filename** for each **image,** of the form **filename_imagenumber.** These image files **can be viewed** individually using the **Image Viewer** software.

A9.7.2.3 Image file in (new) ASCII format (.aid)

These are ASCII image files in a format similar to that defined for **binary** image files**.** These files **cannot be viewed** using the **Image Viewer** software.

Details of the file format are as follows:

File extension '**.aid**'.

<u>File Header</u>

- File Identification code `AIDF'.
- Parameters Vector = [9, sensmapcode, time_resolution, value_type, planes, pixels, frames_per_second, frames, measurements,frame_description_length]
- The interpretation of elements of the parameters vector is as defined for the binary image file format defined below.

<u>Data Segment</u>

Image frames have the same contents as for '**.bid**' files. The integer plane number, frame number and frame time occupy one line and are followed by a CRLF sequence. The pixel values are presented as a whitespace separated sequence of pixels floating point values, terminated by a CRLF sequence. The whitespace may include CRLF sequences to aid presentation.

A9.7.2.4 Image file in binary format [.bid]

These are image files in **binary** format and can be viewed using the **Image Viewer** software.

Details of the file format are as follows:

File extension '**.bid**'

<u>File Header</u>

- File Magic Number = ECBIDF00HEX
- Parameters Vector = [9, sensmapcode, time_resolution, value_type, planes, pixels, frames_per_second, frames, measurements,frame_description_length]

The interpretation of the members of the parameters vector for capacitance data files is retained. The number value_type identifies the type of data stored in frames. The only currently supported value for this field is `0', indicating that image values are all normalised permittivity values in the nominal range [0, 1]. Due to the non-linearities of the back projection algorithm and the 2-point calibration, this range is extended to [-0.5, 1.5]. The number pixels identifies the number of pixel values in each frame of image data. The number frame_description_length indicates the length of the free format string description (including the terminating zero), this facilitates random access of frame data.

Data Segment

The data segment consists of a contiguous sequence of frames. Each frame consists of the following data:

- 32-bit Integer plane number (always zero for single plane systems)
- 32-bit Integer frame number in sequence (for this plane)
- 32-bit Frame Time in milliseconds (since nominal start time, in milliseconds.)
- IEEE-754-1985 32-bit floating point volume ratio (normalised to unity).
- Null ((char) 0) terminated ASCII string frame description. The contents will be free format to be populated as required by the reconstruction engine and indicate supporting frame related reconstruction information (e.g. number of iterations).
- measurements x IEEE-754-1985 32-bit encoded floating-point values, in the measurement sequence specified in the corresponding sensitivity map file (sensmapcode). These are the values used to drive the reconstruction engine and may have had transformations applied (eg sensor modelling).
- pixels x IEEE-754-1985 32-bit floating-point values representing the pixel values in the measurement order specified in the corresponding sensitivity map file (sensmapcode).

A.9.7.2.5 Image file in PPM format (.ppm)

These are ASCII image files encoded in **RGB** format as defined in the **P**ortable **P**ix**M**ap standard. These files **cannot be viewed** using the **Image Viewer** software. The following paragraphs describe the structure of these files.

The Portable pixmap file format is part of the Extended Portable Bitmap Utilities (PBMPLUS). PPM is used as an intermediate format for storing color bitmap information generated by the PBMPLUS toolkit. PPM files may be either binary or ASCII and store pixel values up to 24 bits in size.

ppm - portable pixmap file format
The portable pixmap format is a lowest common denominator color image file format.

The definition is as follows:
A "magic number" for identifying the file type.

A ppm file's magic number is the two characters "P3".
Whitespace (blanks, TABs, CRs, LFs).
A width, formatted as ASCII characters in decimal.
Whitespace.
A height, again in ASCII decimal.
Whitespace.
The maximum color-component value, again in ASCII decimal.
Whitespace.
Width * height pixels, each three ASCII decimal values between 0 and the
specified maximum value, starting at the top-left corner of the pixmap, proceeding in normal English reading order.
The three values for each pixel represent red, green, and blue, respectively;
a value of 0 means that color is off, and the maximum value means that color is maxxed out.
Characters from a "#" to the next end-of-line are ignored (comments).
No line should be longer than 70 characters.

Here is an example of a small pixmap in this format:

```
P3
# feep.ppm
4 4
15
 0  0  0    0  0  0    0  0  0   15  0 15
 0  0  0    0 15  7    0  0  0    0  0  0
 0  0  0    0  0  0    0 15  7    0  0  0
15  0 15    0  0  0    0  0  0    0  0  0
```


A9.7.2.6 Volume Ratio files

A sample twin-plane volume ratio file is shown below. The files give the frame time in seconds and the voidage for each plane. The format for a single-plane file is similar except that the data for the second plane is missing.

Time	VR1	VR2
3.152	0.990	0.966
3.166	0.991	0.964
3.195	0.990	0.963
3.209	0.992	0.965
3.226	0.991	0.963
3.255	0.991	0.963
3.269	0.993	0.963
3.283	0.994	0.964
3.314	0.992	0.960

A9.7.2.7 Normalised capacitance file (ASCII) (.ncp)

These are normalised capacitance data in **ASCII** format. These files **cannot be viewed** using the **Image Viewer** software.

APPENDIX 10

3-D PLOTTING PROGRAM PLOT3D FOR ECT DATA

CONTENTS

A10.1 INTRODUCTION

PLOT3D is a stand-alone compiled Matlab program which allows **capacitance data** captured using the **PTL ECT32 software** to be displayed as a set of **multiple 2-D image frames** and viewed in **2 or 3-D**. The processed data can be converted into a **video file** which can be viewed using standard video playback software such as Windows Media Player.

A10.2 THE CONTROL PANEL

When the program is started, the **Control Panel,** shown in figure A10.1 is displayed in an otherwise empty plotting window. The **Control Panel** consists of an upper (**Plotting controls**) area and a lower (**Data Source controls**) area.

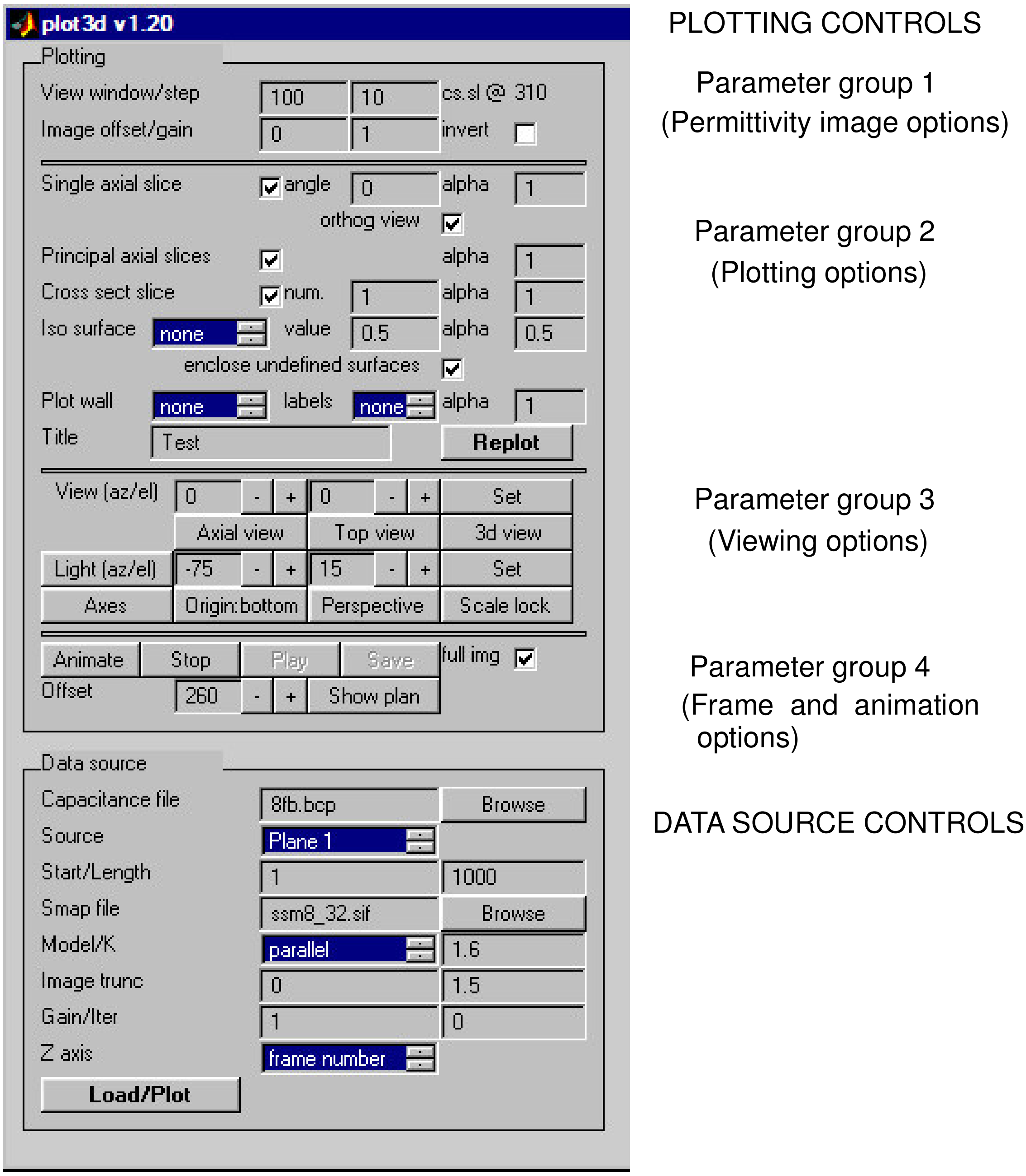


Figure A10.1 Plot3d Control Panel

The **Data Source controls** are used to extract, modify and load capacitance data from a captured capacitance data file. Once the selected data has been loaded, the **Plotting controls** are used to display the image data in the required format. In general, it is possible to display all or part of the captured data file in the plotting window and to view and scroll through this file to examine the captured data.

It should be noted that some of the operations of the buttons in the **Control Panel** under Matlab differ from the standard Windows conventions. For example, **it is necessary to click on options selected using Matlab scroll boxes to set them to be active (Blue). Failure to do this will prevent the selected option from being activated.**

A10.3 LOADING DATA FROM A CAPTURED DATA FILE

A10.3.1 DATA SOURCE CONTROLS

Data from a **captured data file** is extracted and loaded using the controls in the **Data Source area**. The details are as follows:

Capacitance file:	Input the filename of the input capacitance data
Source:	Select the plane to plot (Plane1 or Plane 2)
Start/ Length:	**Start**: Select the **first frame number** to be extracted from the input data file. **Length:** Set the **number of frames** to be extracted.
Smap file:	The **sensitivity map** file to be used to generate the permittivity images.
Model/K:	**Model:** Select the capacitance/permittivity model to be used to construct the permittivity image (Parallel/Series/Maxwell). **K**: Select the permittivity ratio to be used with the series and Maxwell models
Image Truncation:	Insert the upper and lower values of normalised permittivity to be used for image truncation at each iteration when using iterative image reconstruction.
Gain/Iter:	**Gain**: The gain to be used for iterative image reconstruction **Iter**: The number of iterations to be used
Z axis:	Select from **Frame number/Time stamp/Time delay** **Time stamp** is the frame time from the start of data capture. **Time delay** is the time of the frame relative to the last frame in the extracted data set.
Load/Plot	This loads the selected data and displays it in the format selected in the **Plotting options** area. The **load button** stays depressed until the data has loaded. This operation can take some time due to reconstruction effort, especially if iterations are specified.

Note: For simple **LBP image reconstruction**, set the **Iter** value = 0. In this case, the **Image truncation** and **Gain/Iter** parameters are ignored.

A10.4 PLOTTING THE CAPTURED DATA

Once the **input data** has been loaded using the options set in the **Data source** area, the **PLOTTING CONTROLS** can be used to visualise the **image data** in a wide variety of ways.

There are **four principle groups of parameters,** arranged vertically within the **Plotting controls**.

Note that when any parameters in **Parameter groups 1 and 2** are changed, the **Replot** button must be used **to generate an updated new image.**

A10.4.1 PERMITTIVITY IMAGE OPTIONS (PARAMETER GROUP 1)

The first group of parameters modify the set of images reconstructed by the **Load/plot** button in the **Data Source** group. The functions of the parameters in this group are as follows:

View window	**Window** is the total number of frames to be plotted within the **range of total data frames loaded** .
/step:	**Step** is the number of frames to be incremented when stepping through the **set of image frames** using the **Offset buttons**. It also determines the **Frame increment steps** to be used in the **Animate** option. (see section 4.4
Image offset /gain	**Image offset:** The **permittivity offset** (use 0 for no offset) applied to the image pixels **Gain:** The **gain factor (Use** 1 for normal gain) applied to the image pixels.
Invert	If this option is ticked, an inverted image is plotted. This facility is useful for plotting **iso-surfaces (section 4.3)** of lower permittvity objects inside a higher permittivity medium (for example, gas bubbles in oil).
cs.sl@xx	This parameter is normally "greyed out" unless the **Cross sectional slice** option (see **parameter group 2**) is selected, in which case it displays the frame number (xx) of the displayed cross sectional image.

A10.4.2 PLOTTING FORMAT OPTIONS (PARAMETER GROUP 2)

The parameters in this group determine the **format** of the plotted images. **It is possible to plot images in one or more formats simultaneously**. In general, the **Replot** button must be used to plot a **new or updated image** after any of the parameters in the **Plot format options group** are changed.

A10.4.2.1 Single axial slice option.

If ticked, this option shows the **permittivity distribution** across a **2-D diametric axial plane** (**axial slice)** oriented at a **selected angle** from the **cross-sectional axis origin**.

Angle: This parameter sets the angle of the **axial image slice** relative to the origin (the horizontal X axis which lies in the plane of the computer monitor screen). A **cross-sectional plan view** of the image plane (**axial slice**) and the angle from which it is viewed (**viewing angle**) can be seen by clicking on the **Show Plan** button (see parameter group 3 below) which produces a plan view of the sensor looking from the **top** of the sensor. The two primary X and Y axes (at azimuth angles of 0 and 90 degrees respectively) are shown as dotted lines on the **Plan view** diagram.

Alpha: This sets the **transparency** of the image. 1 is opaque and 0 is transparent.

Orthog view: If this option is selected, the **permittivity image** is viewed from an angle of (-90) degrees relative to the **angle of the diameter which defines the axial slice plane.** If any other views are subsequently selected using the **view buttons** in parameter group 3 (section 4.3), the **view angle** may be temporarily modified, but the view will revert to 90 degrees to the **axial slice** when the **Replot** button is operated.

When the **orthogonal view** option is selected and the angle is changed there will be a slight varition in the width of the image slice. This is due to the variation in the radius of the pixelated circle representation of the sensitivity map.

A typical **Single axial slice image** (for 100 frames starting at frame 1) is shown in figure A10.2 below for an axial permittivity slice across a single axial plane plotted for a diameter of 45 degrees to the reference axis. The LHS axis is the frame number and alpha has been set to 1.

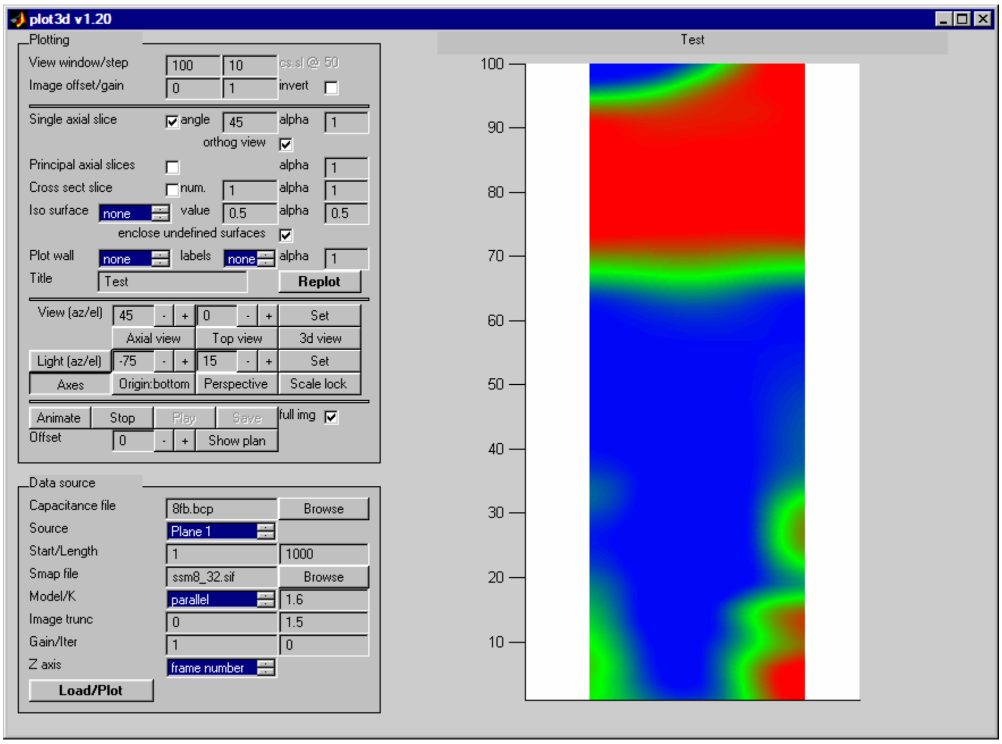


Figure A10.2 Image display in Single axial slice mode

A plan view from the top of the sensor can be seen by clicking on the **Show Plan** button in the 4th parameter group as shown in figure A10.3 below.

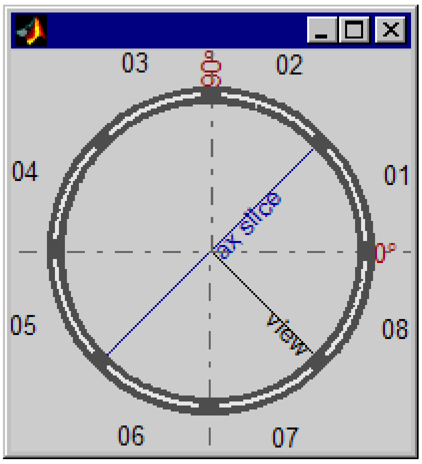


Figure A10.3 Plan view of image plane from top of image set.

Figure A10.3 shows the axial slice oriented at 45 degrees to the origin (0 degree axis) with the view direction clearly indicated. The electrode positions are also shown.

A10.4.2.2 Primary axial slices option

This option plots permittivity profile images in the two diametric planes which coincide with the primary X and Y axes (ie at azimuth angle = 0 and azimuth angle = 90 degrees). The **3_D view option** must be enabled to view the two image axes and the image can be rotated in 15 degree increments by clicking on the **View (az) button**. (see parameter group 3 below).

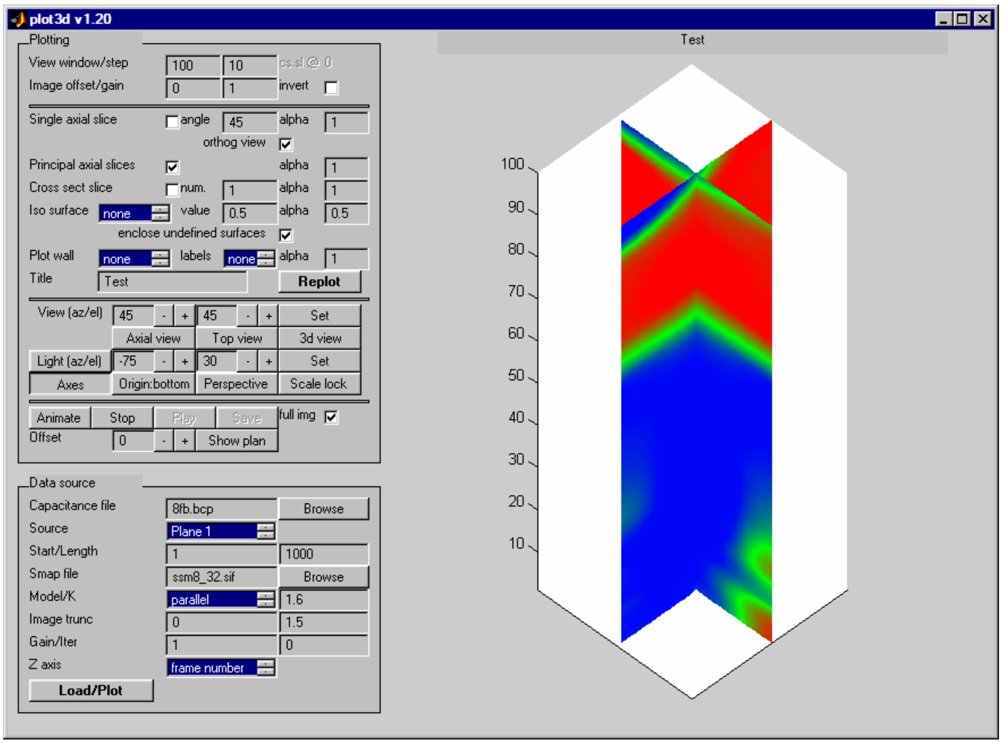


Figure A10.4 Principal axial slices in 3-d view

A10.4.2.3 Cross sectional slices option

This option plots one or more (num) **cross-sectional image frames** at specified times or frame numbers. A typical plot (for num = 3) is shown in figure A10.5

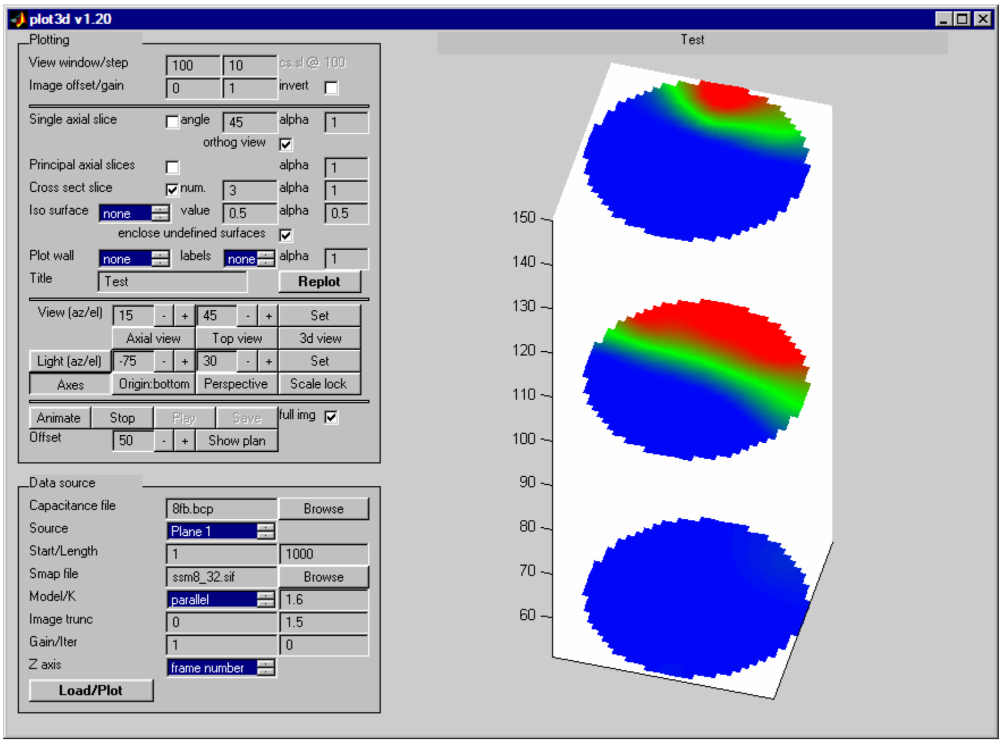


Figure A10.5 Cross sectional image display

The number of images plotted is determined by the number entered in the **num box** as follows:

If **num** = 1, the **middle frame** in the displayed sequence is plotted and the frame number is indicated in the **cs.sl@ (cross-section slice)** parameter in **Parameter group 1.**

If **num** = 2 the **first** and last **frames** in the displayed sequence are plotted and if n = 3 the **centre frame** is also plotted. etc.

The images can be rotated in both azimuth and elevation using the **View (az/el) +/-** buttons in **Parameter group 3.**

.

A10.4.2.4 Iso surface option.

This option plots a **3-D surface contour** of the permittivity images at a **specified permittivity value** (typically in the range 0 - 1) which is set in the **value box.** The **colour** chosen to plot the surface can be chosen from a **scroll box** list. Note that when the **map** option is selected from this list, the colour corresponds to that defined by the standard **PTL RGB permittivity colour scale**. The **surface transparency (alpha)** is set in the **alpha** box**.**

Note also that the **iso-surface image** needs to be viewed in **3-D** with the **Lighting** activated (see **Viewing options**) to allow the **iso-surface** to be viewed effectively. A typical image is shown in figure A10.6 below. This image was produced with the **Enclose undefined surfaces** and **Plot wall** options enabled.

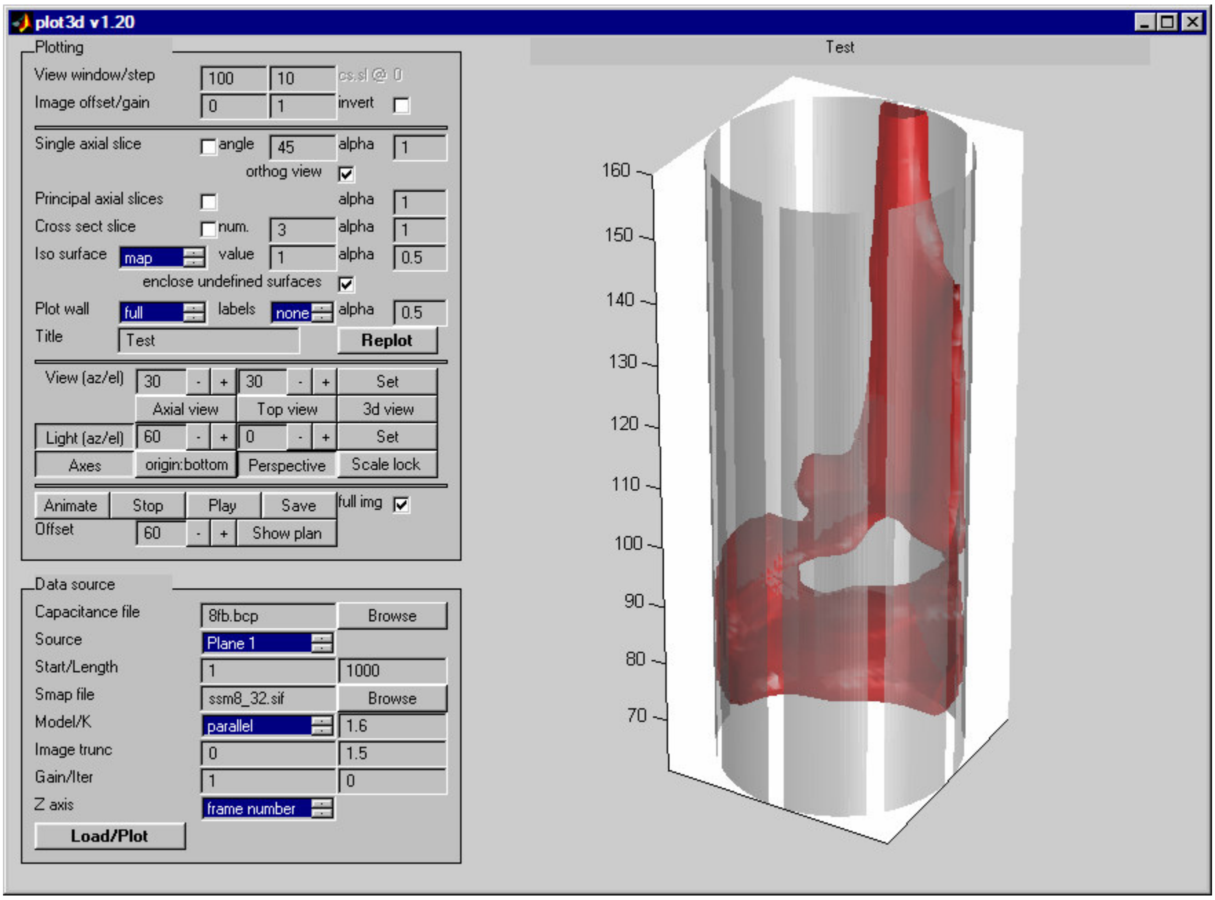


Figure A10.6. Iso surface image display

There are so many possibilities with the different parameter settings in **iso-surface mode** that we suggest that users experiment with the effects of the various viewing and lighting controls as described in the **Viewing Options** parameter group below.

The **Enclose undefined surfaces** button completes the iso surface volume at the tube boundaries.

A10.4.2.5 Wall Plotting Controls

The **Plot wall** option allows the sensor wall to be plotted. **Electrode gaps** are indicated by gaps in the plotted wall. The options are **None, half** and **full.**

If **None** is selected, no wall is plotted. if **Half** is selected, the half sector of the wall behind the pipe contents view is plotted (this is view dependent) and if **Full** is selected, all of the wall is plotted.

Note that the **wall transparency** (**alpha**) can be set in the range 0 to 1 using the **alpha box**. A value of **alpha** = 0.5 was used to produce the image of figure A10.6.

The **Title box** allows an **image title** to be inserted at the top of the image.

Notes:

1. When any parameters in the **Plotting options** parameter group are changed, the **Replot** button must be used to plot an updated new image. This replots the images with the specifed parameters

2. It is possible to use more than one of the plotting options simultaneously.

A10.4.3 VIEWING OPTIONS (PARAMETER GROUP 3)

The controls in this group determine how the image formats selected in the **Plotting options** group are viewed. In general, the controls in this group have an immediate effect on the image and there is no need to use the **Replot** button. Note that **Azimuth** and **Elevation** views can be difficult to comprehend initially because of the changes in perspective.

The **View Az/El** buttons allow the **image viewing angle** to be changed in **azimuth** and **elevation** and pressing them increments or decrements the viewing angle in 15 degree steps. Alternatively, a specified number (degrees) can be entered directly and applied using the **Set button.**

An **azimuth view angle** of **0 degrees** corresponds to a normal view of the PC screen. from the front, which corresponds to a view along the -Y axis in the direction of the origin at (ie at -90 degrees to the X axis).

An **azimuth angle** of **90 degrees** shows a view of the image from the **RHS** of the PC screen. Similarly, an a**zimuth angle** of **180** degrees corresponds to a **rear view** of the image and **270 degrees** corresponds to a view from the **LHS** of the screen.

An **elevation view angle** of **0 degrees** corresponds to a view in the **X-Yplane**. A **view angle** of **90 degrees** views the image from the **top** and a **view angle** of **-90 degrees** corresponds to a view from the **bottom** of the image.

The function of the **3 View buttons** is to reset the **view parameters** to commonly useful values as follows:

Axial view: Displays the selected image plots as axial slices in the two principal orthogonal X-Y axis planes (at angle = 0 and angle = 90 degrees).

Top view: Views the selected image plots from the top.

3-D view: Views the selected image plots in 3-D.

Light (az/el): This button toggles turns on lighting when depressed. The angle from which the illumination is cast is determined by the azimuth and elevation angles selected and works in a similar way to the View angles .
The lighting option should be turned on and the angle adjusted to suit to view iso-surface images.
The lighting angle can be seen by clicking on the **Show Plan** button in the 4th Parameter group.

Axes This button displays the **vertical scale axis** when it is depressed. The scale format (frame number/elapsed time/time delay) depends on the **Z axis option** chosen when the data was loaded.

Origin This button cycles around 4 options as follows:

Top: frame and time origin (frame 1, time 0)are at top of image.
Left: frame and time origin (frame 1, time 0) are at LHS of image.
Bottom: frame and time origin (frame 1, time 0) are at bottom of image.
Right: frame and time origin (frame 1, time 0) are at RHS of image.

The use of these options allows the view to be set to match the direction of flow

Perspective When depressed, this button produces a perspective view of the image.

Scale Lock: When images are rotated, Matlab automatically adjusts the image scaling to fit the images (including the projected axes) to the available window size. This auto-scaling has the effect of changing the apparent image size. It can be disabled and the scaling fixed to that of the current image by depressing the **Scale lock** button.

A10.4.4 FRAME AND ANIMATION OPTIONS (PARAMETER GROUP 4)

The controls in this group allow:

1. The frame number range to be incremented and re-plotted. This is carried out using the **Offset +/-** buttons which increment the range of frames plotted by the **step** interval defined in the **Permittivity Image parameter group 1**. The frame number of the lowest frame number plotted is displayed in the frame number offset box. Alternatively, a **frame number** can be entered directly in the offset box and the **Replot** button used to dsisplay the new image set.

2. A Plan view of the **image slice angle**, **viewing angle** and **lighting angle** to be displayed by clicking on the **Show plan button**.

3. A **video file** to be created showing the full range of frames loaded, incremented by the **step** interval. The file is created using the **Animate** button and once it has been created, can be viewed by pressing the **Play** button. Alternatively, the file can be saved as a **.avi file** using the **Save** button and can then be viewed by other software, eg **Windows Media Player.** The **save** operation can take some time to complete. If the **Stop** button is pressed during the animation process, the animation creation is abandoned. It may be necessary to press this button a few times to interrupt the animation process.

APPENDIX 11

MAKEMAP SENSITIVITY MAP GENERATION SOFTWARE

The **Makemap** program calculates **sensitivity matrices** for circular ECT sensors with either internal or external electrodes and for different excitation protocols. It is supplied as compiled software and, following the installation of the software, can be run by clicking on the Makemap icon in the ECT program group window. The sensitivity matrices generated by this software will be written to the c:\makemap folder and must be copied to a suitable folder for use with the required image reconstruction program.

A11.1 INTRODUCTION

PTL ECT systems use a basic **linear back-projection** (**LBP**) algorithm to generate approximate ECT images. This algorithm, which assumes that the relationships between the permittivity distribution inside the capacitance sensor and the measured inter-electrode capacitances can be expressed as sets of linear equations, consists of a pair of **matrix equations** known as the **forward** and **inverse** problems**.** The **forward problem** calculates **t**he inter-electrode capacitance values from a given permittivity distribution. The **inverse problem** calculates the permittivity distribution inside the sensor from the inter-electrode capacitance measurements. The objective in ECT is usually to solve the **inverse problem**, that is to obtain the distribution of permittivity inside the capacitance sensor from measurements of the capacitances between sets of electrode-pairs.

A11.1.1 The Forward Problem

In the **forward problem** the **the measured values of inter-electrode capacitance C** and the **internal permittivity distribution K** inside the sensor are assumed to be related by a matrix transform **S,** which is known as the sensor **Sensitivity Matrix**. That is:

$$\mathbf{C} = \mathbf{S}.\mathbf{K} \qquad (1.1)$$

In this equation, **C** is a column vector containing the set of **m** normalised inter-electrode capacitance measurements **c** for one image frame, **K** is a column vector containing the set of **n** normalised pixel permittivities **k,** and **S** is a normalised **m x n** matrix (the **sensitivity matrix).** The **sensitivity matrix S** can be considered to consist of a set of **m** sub-matrices, each containing **n** elements, and these **sub-matrices** are often referred to as **sensitivity maps** (or sometimes sub-maps, see below) as they define the sensitivity of a specific electrode-pair capacitance (1 of **m**) to a change in the permittivity of each pixel in the image. In current PTL software, all of the values in these matrices and column vectors are normalised so that the values in the **C** and **K** column vectors lie within the nominal range 0 to 1. The sensitivity matrix for the ECT sensor must be either calculated or measured before any ECT images can be constructed.

We will try to stick to the convention that the term **"Sensitivity matrix"** means the matrix **S** as defined above, whereas the term **"Sensitivity map"** refers to one of the m sub-matrices. However, the term **"Sensitivity map"** is often used instead of "**Sensitivity Matrix**", in which case the sensitivity maps are usually referred to as sub-maps. Readers should be aware of this potential confusion. Fortunately, the context will often clarify which matrix is being referred to.

A11.1.2 The Inverse Problem

The **inverse problem** calculates the **permittivity distribution K** inside the sensor from the measured **inter-electrode capacitances C** and is defined by equation 1.2.

$$\mathbf{K} = \mathbf{Q}.\mathbf{C} \qquad (1.2)$$

In principle, **Q** is simply the inverse of the sensitivity matrix **S**. Unfortunately, in most cases it is not possible to obtain an exact inverse of the sensitivity matrix as this requires **S** to be a square matrix, which can only be the case if the number of pixels in the image equals the number of independent electrode-pair capacitance measurements. In reality there are usually less than 100 independent capacitance measurements available while 1000 pixels is often the minimum desirable image resolution. Consequently, some form of approximate inverse must be used. One option, which is adopted in the **LBP** algorithm and can be justified by a simple physical explanation, is to use the transpose of the sensitivity matrix, $\mathbf{S}^{\mathbf{T}}$ for **Q.**

A11.1.3. Sensitivity Matrix formats

There are currently two standards for sensitivity matrices for use with **PTL** software and these are defined in section A11.6. Matrices for use with the original **PCECT** software contain a restricted amount of information, whereas matrices for the newer **ECT32** software contain additional data which allows the electrode positions to be plotted on the permittivity image. The **PTL Makemap** software allows sensitivity matrices to be calculated for any circular-cross-section ECT sensor, using an analytical field calculation method.

A11.2 PRINCIPLE OF OPERATION OF MAKEMAP SOFTWARE

The **sensitivity matrix** for an **ECT sensor** defines, for each **electrode-pair** combination, how much a **change in the permittivity** of each **pixel** inside the sensor changes the **capacitance measured between the specified electrode pair,** when **one electrode** is excited at a **potential V** and the **other electrode** is held at **ground potential**. The **permittivity** of the pixel whose **sensitivity coefficient** is being measured is assumed to change from its **minimum** value at **calibration** to its **maximum value** at **calibration**, while all of the other pixels are held at their minimum values.

The **Makemap** software calculates **sensitivity matrices** for sensors with a **circular cross-section and** uses an **analytical method**, based on expressing the **potential distribution inside the sensor** as **a mathematical series**. The advantage of this method is that it can produce high-accuracy results with minimal computational effort compared with the finite-element approach. Figure A11.1 shows the cross-section of a typical ECT sensor, consisting of a set of measurement and axial guard electrodes located on the outside wall of a dielectric cylinder.

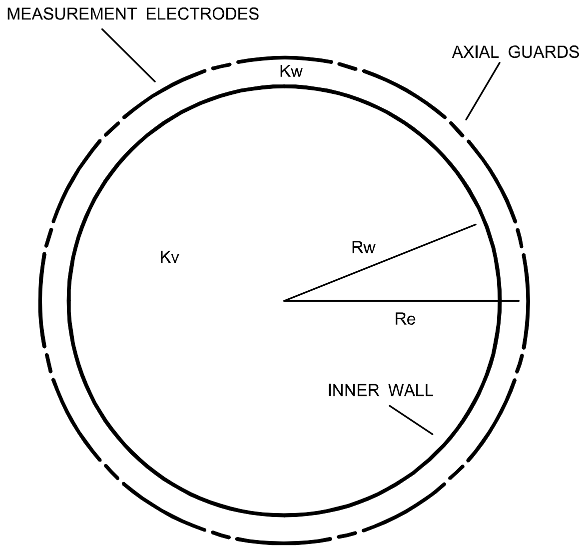


Figure A11.1 Cross-section of circular ECT sensor

The basis of the method for calculating the sensitivity matrix can be summarised in the following steps:

1. Define the **potential distribution** around the **perimiter** of the **capacitance sensor** for the case where **one electrode,** whose centre is located at **$\theta = 0$**, is excited at a **constant potential V** and all of the other electrodes are **grounded** (held at zero potential) as shown in figure A11.2 below:

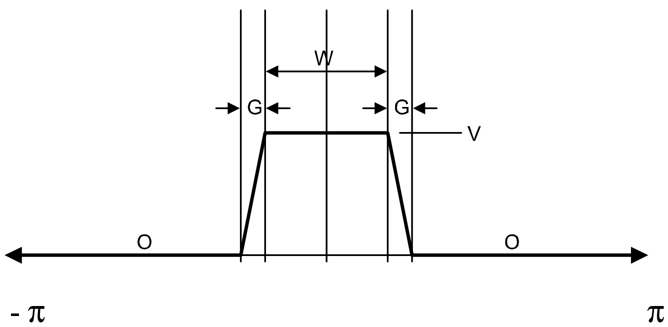


Figure A11 .2 Potential distribution around sensor in circular electrode plane

Figure A11.2 shows the potential distribution around the perimeter of the sensor, which has been represented, for convenience, as a straight line extending from - π to π. All electrodes are maintained at ground (0V) potential apart from the source electrode (of width W and located at θ = 0), which is excited with an alternating voltage V. The electric field is assumed to change linearly between V and 0 over the width of the insulating gap (G) between the measurement electrodes and the next electrode or screen.

2. Derive an expression for the **electrical potential (voltage) distribution V(r, θ) inside the sensor** for this same configuration.

The potential distribution $V_E(\theta)$ around the outside of the dielectric cylinder can be expressed as a Fourier series expansion:

$$V_E(\theta) = \alpha / \pi + (2/\pi).\sum_{n=1}^{\infty}((\sin(n\alpha).\sin(n.\beta))/(n^2.\beta)).\cos(k.\theta) \qquad (2.1)$$

where α and β are two angular parameters with dimensions in Radians and are defined as follows: W = 2.α - β,   G = 2.β.

3. (Optional). Apply **smoothing** or **weighting** functions to this potential distribution.

4. Define a **convention** for the **location** of the **centres of all of the electrodes** in terms of **θ. Similarly,** define the locations of the **source a**nd **detector** electrodes in terms of **θ**.

5. Calculate the **potential distribution V_s** for the chosen **source electrode,** by rotating the potential distribution **V(r, θ),** calculated in 2, through the angle between the **source electrode** and **θ = 0.** Similarly, calculate the **potential distribution** (**V_d**) for the chosen **detector electrode,** by assuming that this electrode is now a source electrode, with all other electrodes grounded. This second potential distribution is again obtained by rotating the potential distribution **V(r, θ),** calculated in 2, through the angle between the **detector electrode** and **θ = 0.**

It can be shown that the potential distribution $V_1(r,\theta)$ inside the cylinder wall is given by the equation:

$$V_1(r,\theta) = a_0 + \sum_{n=0}^{\infty} a_n.r^n.\cos(n.\theta) \qquad (2.2)$$

where the coefficients a_0 and a_n are calculated from the sensor geometry and knowledge of the permittivity of the materials inside the sensor.

6. Calculate the **electric field distributions E_s** and **E_d** inside the sensor for the source and detector electrodes by differentiating the **potential distributions** for these two electrodes.

7. Calculate the **sensitivity to a change in permittivity of each pixel in the image** for each **source-detector electrode pair**, by integrating the products **E_s** and **E_d** across the **area of each pixel** of the sensor in turn. The **set of pixel sensitivities** for each **electrode-pair** is known as a the **sensitivity map** for that electrode-pair and the **complete set of sensitivity maps** for **all electrode-pairs** is known as the **sensitivity matrix** of the sensor.

8. Calculate the absolute inter-electrode capacitances C_A by integrating the normal electric field component E_n over the arc length of the electrode using equation (2.3).

$$C_A = (8.854 . K_w . L / V) \int E_n . d\theta \qquad (2.3)$$

Where C_A is the capacitance in fF, L is the axial electrode length in millimeters, K_w is the relative permittivity of the dielectric tube wall and V is the voltage applied to the source electrode.

A11.3. USER INSTRUCTIONS FOR MAKEMAP SOFTWARE

When the Makemap program is run, the window shown in figure A11.3 below appears and allows the user to enter the parameters which define the sensor geometry:

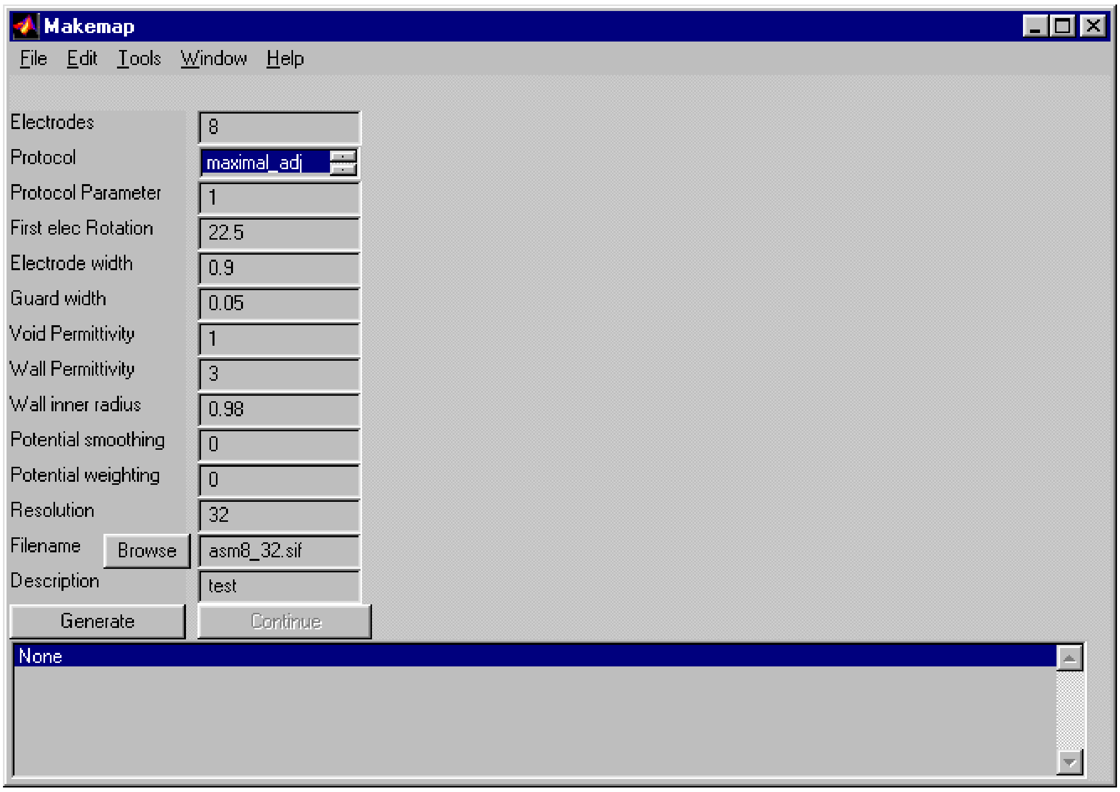


Figure A11.3 Makemap data entry window

Details of the parameters which must be entered before a matrix can be generated are listed in the next paragraph. Additional information about each of these parameters is given in the following section.

Details of input parameters

1. Electrodes (8)	*The number of electrodes around the sensor circumference*
2. Protocol	*The excitation protocol (only current option is maximally – adjacent)*
3. Protocol parameter (1)	*Enter 1 for single source and detector electrode-pairs (DAM200).*
4. First elect.rotation (22.5)	*Angular location of centre of the first electrode from 3 o clock **
5. Electrode width (0.9)	*Decimal fraction of max possible electrode arc*
6. Guard width (0.05)	*Guard arc length as decimal fraction of max electrode arc*
7. Void permittivity (1)	*Permittivity of material inside sensor*
8. Wall permittivity (3)	*Relative permittivity of sensor wall*
9. Wall inner radius (0.99)	*Normalised internal tube radius (in range 0-1)*
10. Potential smoothing (0)	*[1=yes,0=no]*
11. Potential weighting (0)	*[1=yes,0=no] (0)*
12. Map resolution (32)	*Resolution of pixel grid*
13. Filename (asm8_32.sif)	*file name for generated matrix*
14. Description (test)	*Description embedded in matrix file*

More detailed information about some of these parameters is given below:

1. Electrodes. The number of capacitance electrodes located around the circumference of the sensor in a single axial location.

2. The Capacitance Measurement Protocol. **The only measurement protocol currently available is where one or more (P) adjacent electrodes are connected together to form a single** virtual source electrode **and the same number of adjacent electrodes are connected to form a** virtual detector electrode**. All other electrodes are held at ground or virtual ground potential. We refer to this arrangement as the** maximally-adjacent protocol (**which produces the** maximum number of unique measurements **using combinations of** adjacent electrodes **(or single electrodes).**

3. **Protocol parameter**. The protocol number **P** defines how many electrodes are interconnected to form each virtual electrode. For ECT systems which use the PTL DAM200 unit, P must be set equal to 1.

4. First elect rotation. This is the angle in degrees between "3 o' clock" and the centre of electrode 1, measured in an anticlockwise direction. (The PTL convention is that this angle = 0 degrees for 6 and 12 electrode sensors and 22.5 degrees for 8 electrode sensors).

*** Note that this angle must be less than 360/E, where E is the number of electrodes.**

5. **Electrode width**. This is the **arc span** of the measurement electrode, expressed as a decimal fraction of the **maximum possible electrode arc span**. For example, **the maximum possible electrode arc span** for a 12-electrode sensor is 360/12 = 30 degrees. (This would not allow any insulating gap between adjacent electrodes.) If the actual electrode arc span is 24 degrees, then the fractional **electrode width** would be 24/30 =0.8.

6. **Guard width**. This is the **arc span** of the **axial earthed guard electrode** located between each measurement electrode, expressed as a decimal fraction of the maximum possible electrode arc span. The guard arc span depends on the size of the measurement electrode and the insulating gaps. For example, if the measurement electrode arc span is 0.8 (24 degrees), the maximum guard arc span will be 0.2 (= 6 degrees), although this would again not allow an insulating gap between the axial guard and the electrodes. If the actual guard arc span is 3 degrees, then the fractional **guard width** will be 0.1 (= 3/30).

7. **Void permittivity**. The **relative permittivity** of the **lower permittivity** material to be used inside the sensor. This parameter is only relevant for a sensor with electrodes located on the **outside** wall of a dielectric tube and assumes that **positive** permittivity excursions above the lower calibration level are of primary interest. If **negative** excursions below the upper limit are of primary interest, then it may be preferable to use the **higher permittivity** material value here.

8. **Wall permittivity**. The **relative permittivity** of the **tube wall** if the electrodes are located on the outside of the tube.

9. **Wall inner radius**. The **normalised radius** of the **inside of the tube wall** (where the external tube radius is asumed to equal 1). For example if the tube external diameter is 10 cm and the internal diameter is 9cm, the normalised wall inner radius will be 0.9. In principle, the value 1 should be used for sensors where the **electrodes** are located **inside** the tube wall. However, for sensors with **internal electrodes** (ie where no inner dielectric tube is present), it is best to input a value of wall inner radius = 0.98 rather than 1. This avoids generating maps with very high sensitivity coefficients close to the wall in the gap between the electrodes. Experience shows that if a value of inner radius = 1 is used, the resulting maps can generate images with spurious artefacts close to the wall.

10. **Potential smoothing**. This gives the option of applying fixed angular smoothing to the calculated potential distribution inside the sensor. Experience to-date suggests that good results are obtained without the use of this option and the parameter should normally be set to 0 (off).

11**. Potential weighting**. This gives the option of applying fixed radial weighting to the calculated potential distribution inside the sensor. Again, experience to-date suggests that good results are obtained without the use of this option and the parameter should normally be set to 0 (off).

12. **Map resolution**. The resolution of the pixel grid for the images. A suitable value is 32 for normal use, or 64 for higher resolution images.

13. **Filename**. The **name** of the **sensitivity matrix file**. For use with **PCECT** software, this must have the format **E8_bin.p32** etc. For use with **ECT32** software, the files must have the file extension **.sif**. The **browse** button should be used to select the folder in which the files will be generated.

14. **Description.** An optional text description which will be embedded in the sensitivity matrix file for identification purposes.

Once the input data has been entered, click on the **Generate** button to start the sensitivity matrix generation software.

The program will run and various intermediate sets of data are generated as described in the following sections. The matrix will be generated after a delay of a few minutes. However, if smoothing is enabled, the delay will be much longer than when it is disabled. The examples shown in the remainder of this section are for an 8-electrode sensor with internal electrodes.

A11.4 WINDOWS DISPLAYED DURING MATRIX GENERATION

The first data to be generated is an image showing the electric equipotential lines (scaled in volts) inside the sensor as shown in figure A11.4. Note that the source electrode is always shown at the 3 o' clock position in these plots for simplification. The equipotentials are plotted on a colour scale where red shows the highest potentials and the calculation assumes that the potential applied to the source electrode is 1 volt.

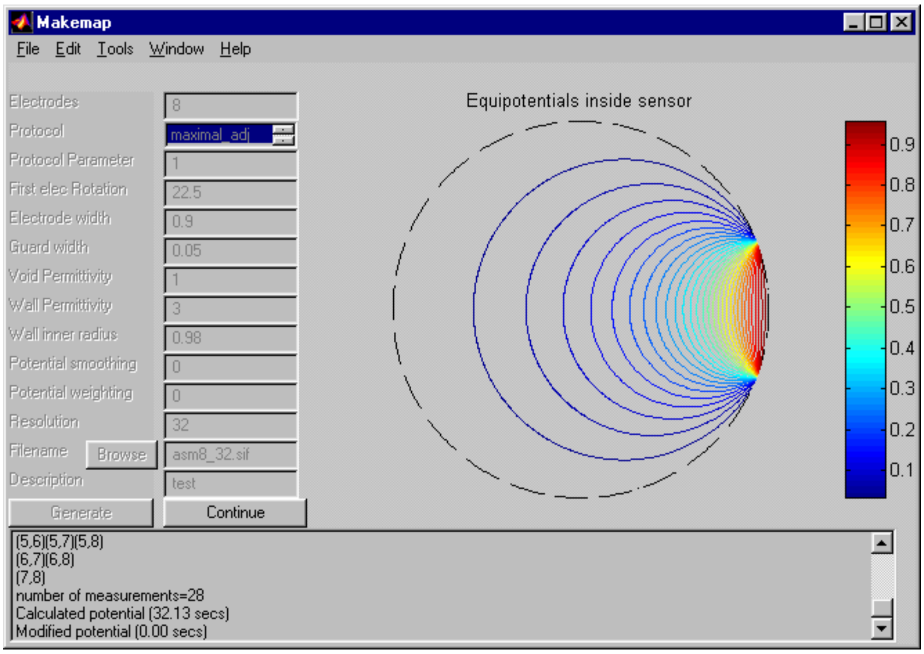


Figure A11.4 Equipotential lines inside an 8- (internal) electrode ECT sensor.

Click on the **Continue** button. The next image to be displayed (figure A11.5) is a representation of the electric field lines (electric flux) between the source electrode and the remaining grounded electrodes. The density of the lines indicates the local field strength.

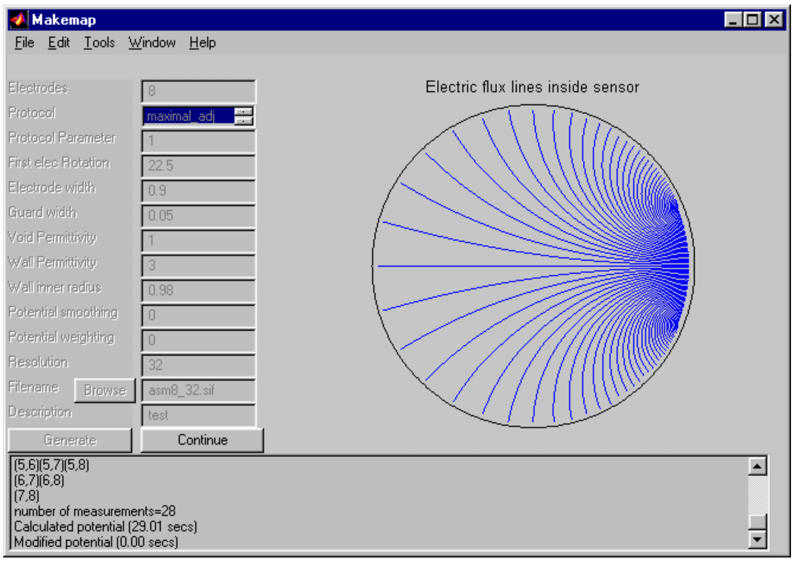


Figure A11.5 Electric field distribution inside an (8-internal-electrode) sensor.

Click again on the Continue button. The next image to be displayed is the electric potential distribution around the inner wall of the dielectric tube (or in the electrode plane if there is no dielectric tube), as shown in figure A11.6. The horizontal scale is in degrees, measured anticlockwise from 3 o' clock.

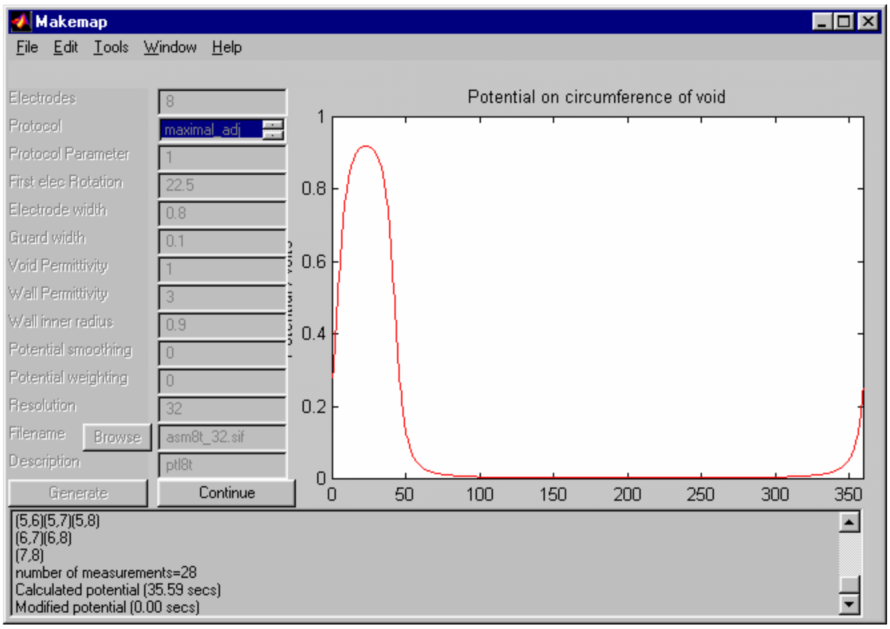


Figure A11.6 Electric potential distribution around inner wall of 8-internal-electrode sensor

Click on the Continue button again. The matrix generation sequence will now start and the maps for individual electrode-pair combinations will be generated one at a time. It takes approximately 11 seconds (on a 500MHz Pentium 3) to generate the map for each pair, so there will be a delay of a few minutes before the complete set of maps is produced. Once the map for the last pair has been generated, the set of primary sensitivity maps are displayed, as shown in figure A11.7 for an 8-electrode matrix.

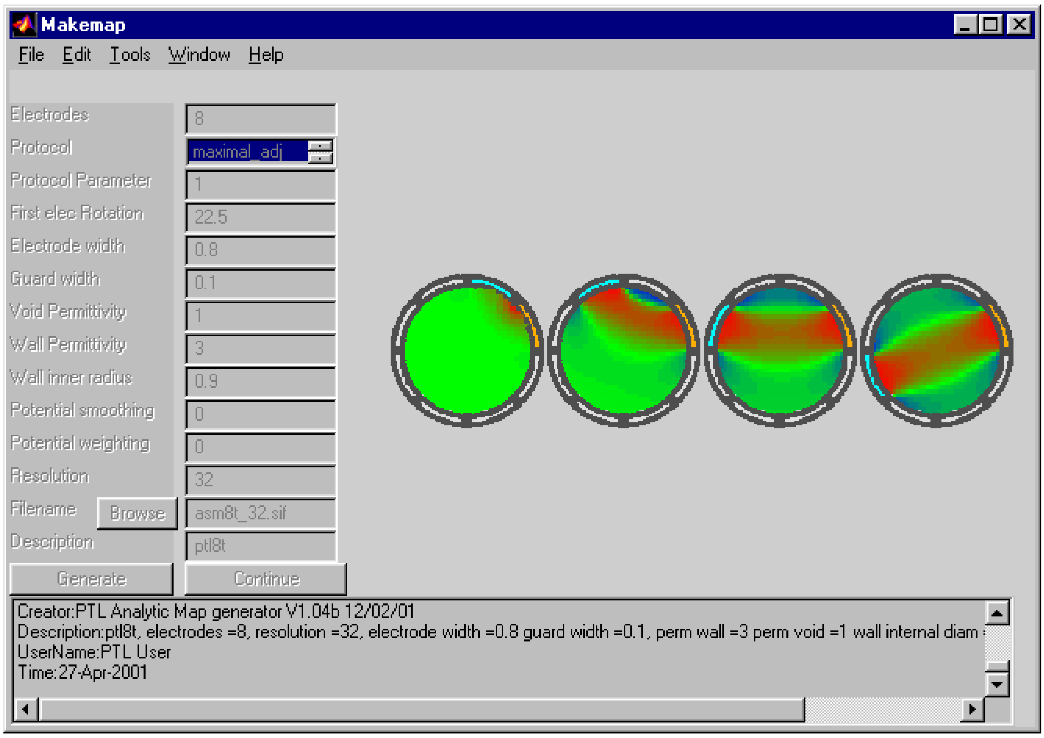


Figure A11.7 Set of primary sensitivity maps for an 8-electrode protocol 1 ECT sensor.

A11.5 SENSITIVITY MATRIX DISPLAY FORMAT

Figure A11.7 shows the **4 unique primary sensitivity maps** for the electrode pairs C12, C13, C14 and C15 for an **8-electrode sensor**. All of the other sensitivity maps can be derived from these 4 primary maps by a combination of rotation and mirror transformations.

The sensitivity maps are plotted in colour, with **red** indicating a **positive sensitivity coefficient**, **blue** indicating a **negative sensitivity coefficient** and **green** indicating a **coefficient close to zero**. The **electrodes** are shown in their correct positions, with the **source electrode** shown in **orange** and the **detector electrode** shown in **light blue**.

Because of the large range of values in each set of sensitivity coefficients, it is difficult to find a colour scale which accurately represents the coefficients in the map and allows the user to assess the content of each map. We have therefore adopted the following strategy in plotting the maps:

The colour red represents large positive values of sensitivity, blue represents large negative sensitivities and green represents values around zero. The graduation from green to red and blue is non-linear, with compression at the top and bottom extremes to assist viewing of the sensitivity coefficients.

The sensitivity maps show the relative sensitivities for the pixels inside each each map but do not show the relative magnitude of each sub-map to the other maps. The absolute values of the sensitivity coefficients in each map can be obtained using the **ect_smapplot** and **ect_smapdump** programs in the **MatECT** software.

A11.6 FORMAT OF SENSITIVITY MATRIX FILES

There are currently two versions of **PTL sensitivity matrix** files.

A11.6.1 PCECT FORMAT

The first version, for use with the original PTL **PCECT** and **transputer** software, consists of sets of **m x n** matrices (**one matrix** for each **sensitivity map**), where **m** is the number of **unique capacitance measurements** (66 for a 12-electrode sensor) and **n** is **the number of pixels in the image** (1024). These maps have file names of the form **E12_bin.p32**.

Each **sensitivity matrix** can be considered to be made up from a set of **m sensitivity maps**, where each **map** is the **sensitivity map** for a **unique electrode pair**. The maps are arranged so that :

m=1 is the map for electrode-pair 1-2
m=2 is the map for electrode-pair 1-3
.
.
m=66 is the sub-map for electrode-pair 11-12 etc.

Each of these sensitivity maps contains 1024 sensitivity coefficients, each coresponding to one pixel in the 32 x 32 pixel image. These pixels are identified as follows:

Pixel 1 is the first pixel in row 1 of the image (where row 1 is the top row).
.
Pixel 32 is the last pixel in row 1 of the image.
Pixel 33 is the first pixel in row 2 of the image. etc.
.
Pixel 1024 is the last pixel in row 32 of the image (the bottom row)

A11.6.2 ECT32 FORMAT

The overall format for the **ECT32** maps is similar to that for the **PCECT** maps except that the maps only contain sensitivity coefficients for the pixels inside the sensor (812 for a circular sensor). The **ECT32** sensitivity matrix file names are of the form **NSM12_32.sif** .

One further major difference between ECT32 maps and the original maps is that the information in the header section of the file is significantly different as the ECT32 maps contain additional information about the electrode and sensor geometry and excitation protocols. The main differences are detailed in the following section.

A11.6.2.1 ECT32 SENSITIVITY MATRIX HEADER FORMAT

The following data is held in the header section of an ECT32 format sensitivity matrix. The equivalent data for PCECT maps is indicated [inside square brackets].

Strings
Creator: 'PTL Analytic Map generator V1.04b 12/02/01 '
UserName: 'PTL User '
Description: [1x154 char]
Time: '07-Mar-2001 '
[PCECT: Set to unknown]

Number of physical Electrodes (8)
May differ from the number of virtual electrodes
[PCECT: Inferred from filename]

Number of virtual Electrodes (8)
[PCECT: Assumed to be same as number of physical electrodes]

Virtual to physical mapping (virtual elec : physical elec(s).)
1: 1
2: 2
3: 3
4: 4
5: 5
6: 6
7: 7
8: 8
[PCECT and normal protocol ECT32: Assumes 1:1 mapping.]

Measurement Sequence.
(1,2)(1,3)(1,4)(1,5)(1,6)(1,7)(1,8)
(2,3)(2,4)(2,5)(2,6)(2,7)(2,8)
(3,4)(3,5)(3,6)(3,7)(3,8)
(4,5)(4,6)(4,7)(4,8)
(5,6)(5,7)(5,8)
(6,7)(6,8)
(7,8)
[PCECT and normal protocol ECT32: Assumes standard measurement seq.]

Active pixels (812)
[PCECT: Geometry assumed circular, active pixels calculated from generated mask.]

Measurements (32)
[PCECT: Inferred from filename]

Geometry

Mask - indicates positions of active pixels. Only these sensitivity values are stored.

00000000000011111111000000000000
00000000011111111111111000000000
00000001111111111111111110000000
00000011111111111111111111000000
00000111111111111111111111100000
00001111111111111111111111110000
00011111111111111111111111111000
00111111111111111111111111111100
00111111111111111111111111111100
01111111111111111111111111111110
01111111111111111111111111111110
01111111111111111111111111111110
11111111111111111111111111111111
11111111111111111111111111111111
11111111111111111111111111111111
11111111111111111111111111111111
11111111111111111111111111111111
11111111111111111111111111111111
11111111111111111111111111111111
11111111111111111111111111111111
01111111111111111111111111111110
01111111111111111111111111111110
01111111111111111111111111111110
00111111111111111111111111111100
00111111111111111111111111111100
00011111111111111111111111111000
00001111111111111111111111110000
00000111111111111111111111100000
00000011111111111111111111000000
00000001111111111111111110000000
00000000011111111111111000000000
00000000000011111111000000000000

[PCECT: Geometry is assumed to be circular. The mask is generated on the 'fly', size inferred from filename.]

Boundary (only in files generated with makemap - not mandatory)
Coordinates of points describing the shape of the vessel boundary.

1.0000	0
0.9998	0.0175
0.9994	0.0349
0.9986	0.0523
0.9976	0.0698
.	
.	
.	
0.9945	-0.1045
0.9962	-0.0872
0.9976	-0.0698

0.9986   -0.0523
0.9994   -0.0349
0.9998   -0.0175
1.0000   -0.0000
[PCECT: Data not mandatory, so none generated]

Electrode positions (only in files generated with makemap - not mandatory)
Start and end point numbers indicating position of electrodes
3   43
48   88
93   133
138   178
183   223
228   268
273   313
318   358
[PCECT: Data not mandatory, so none generated]

Electrode labels (only in files generated with makemap - not mandatory)
List of strings to label each electrode.
01
02
03
04
05
06
07
08
[PCECTData not mandatory, so none generated]

Sensitivity Coefficients
Matrix of values (measurements x active pixels)
PCECT: Only data stored in file.

APPENDIX 12

MATECT MATLAB PROGRAMS FOR ECT

The **PTL MatECT** software contains a large number of **custom Matlab utilities,** which include options for modifying, viewing and editing basic sensitivity matrices, the generation of single-step transformation matrices (which can then be used to obtain improved permittivity images), the calculation of circular ECT sensor capacitances and the construction of images from capacitance data files and sensitivity matrices. The two sets of software can be supplied either separately or together. The **MatECT** software is supplied as a set of Matlab M files, which can be modified by users if required before they are run using Matlab version 5.3 or later. Users must therefore have a version of **Matlab 5.3** or later available to allow the **MatECT** software to be run.

CONTENTS

0. SOFTWARE INSTALLATION AND INITIATION

The **MatECT** software should be installed as described in Appendix 7. Once the software has been installed, the individual programs (m files) must be run from within **Matlab** (v 5.3 or later). Note that input data files must be located in the **Matlab Work folder** and output data files will be written to this same folder. The **MatECT** software will run on most PCs but will run very slowly if the PC has insufficient memory. We recommend that the software should not be installed on PCs with less than 128MB of RAM. A 500MHz processor or better is preferred.

Once the software has been installed, Matlab must be configured as follows:

1. Run **Matlab** and create a path to the **matect** folder within **Matlab** as follows:

> In the **Matlab command window**, click on **File** in the **menu bar**, then click on **Set Path** in the menu.
> Click on **Path** in the window **menu bar** and then click on **Add to Path** in the **menu**.
> Enter the name of the new folder (**c:\matect**), select **Add to Front**, then click **OK**.
> Close the Path window and confirm that new path is to be added permanently.

2. Most of the **MatECT programs** (m files) will run under either Matlab v 5.3 or 6.0. The only exception is the movie file program which will not run under Matlab 5.3.

3. Unless Browse buttons are provided, **all input data files** must be located in the **Matlab Work** folder before each program is run. **Output data files** will be generated in the same folder.

1. SENSITIVITY MATRIX VIEWING SOFTWARE

Sensitivity matrices or **transformation matrices** can be viewed in either **graphical** or **numerical** formats using the following programs:

1.1 VIEWING SENSITIVITY MATRICES IN GRAPHICAL FORMAT

To view a **sensitivity matrix** or **transformation matrix**, carry out the following:

1. Copy the **sensitivity matrix file** to be viewed to the **Matlab Work folder**. The file name must be in one of the standard PTL formats (**ECT32** or **PCECT**).

2. The individual sensitivity maps for each electrode-pair are plotted by typing the the following:

ect_smapplot('mapname',N)

where **mapname** is the name of the sensitivity matrix file to be viewed and N is a number which defines how many sensitivity maps are plotted. Note that electrode positions and labels will only be displayed if this information is present in the map file.

1.1.1 SINGLE SENSITIVITY MAP DISPLAY MODE

If N is an integer number greater than 0, a single map will be plotted. For example, if N=4, this will display the fourth map in the sequence defined in section 1.1.4. For a single map display, the electrode numbers and the colour scale are displayed, as shown in figure 1.1, which was generated by typing:

Ect_smapplot('asm8_32.sif',4)

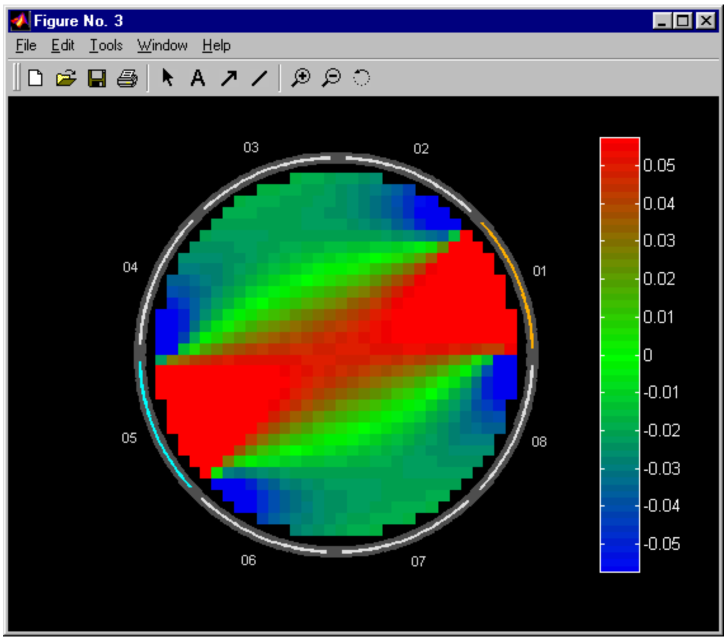


Figure 1.1 Sensitivity map for electrode-pair C15 (N = 4) for an 8-electrode sensor with internal electrodes

1.1.2 PRIMARY SENSITIVITY MAPS DISPLAY MODE

If N = -1. Only the primary maps are plotted as shown in figure 1.2. The electrode numbers are now omitted for clarity.

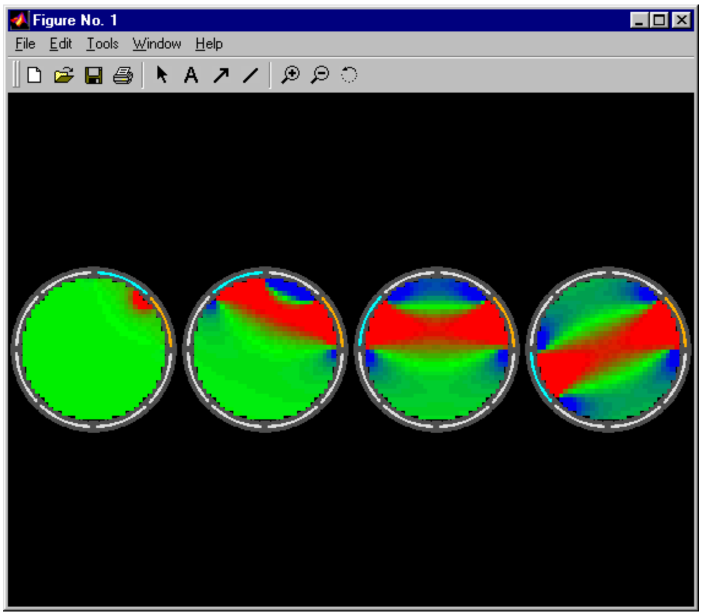


Figure 1.2. Set of unique maps plotted when N = -1

1.1.3 FIRST ELECTRODE SENSITIVITY MAPS DISPLAY MODE

If N = 0, all of the sensitivity maps for the first electrode are plotted as shown in figure 1.3.

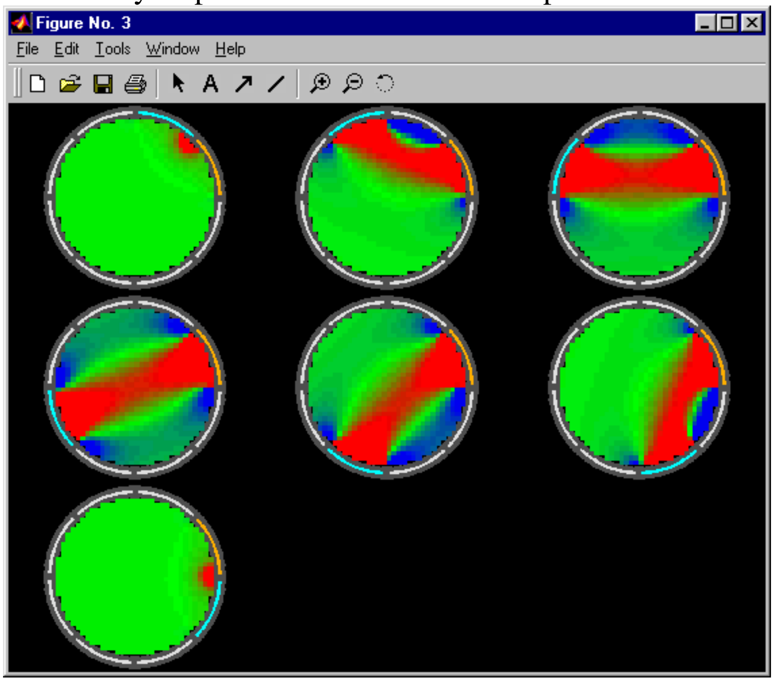


Figure 1.3. Maps for electrode 1 (N = 0).

In both these cases, the sensitivity maps are plotted for each electrode pair in the following horizontal order: C12, C13 .....C1E, where **E** is the **number of electrodes** and **S12** means the **sensitivity map** for the **electrode pair electrode 1** and **electrode 2** etc.

1.1.4 FULL SET DISPLAY MODE

If N is omitted, the full set of sensitivity maps are plotted as shown in figure 1.4. In this case, both the electrode positions and numbers are omitted for clarity.

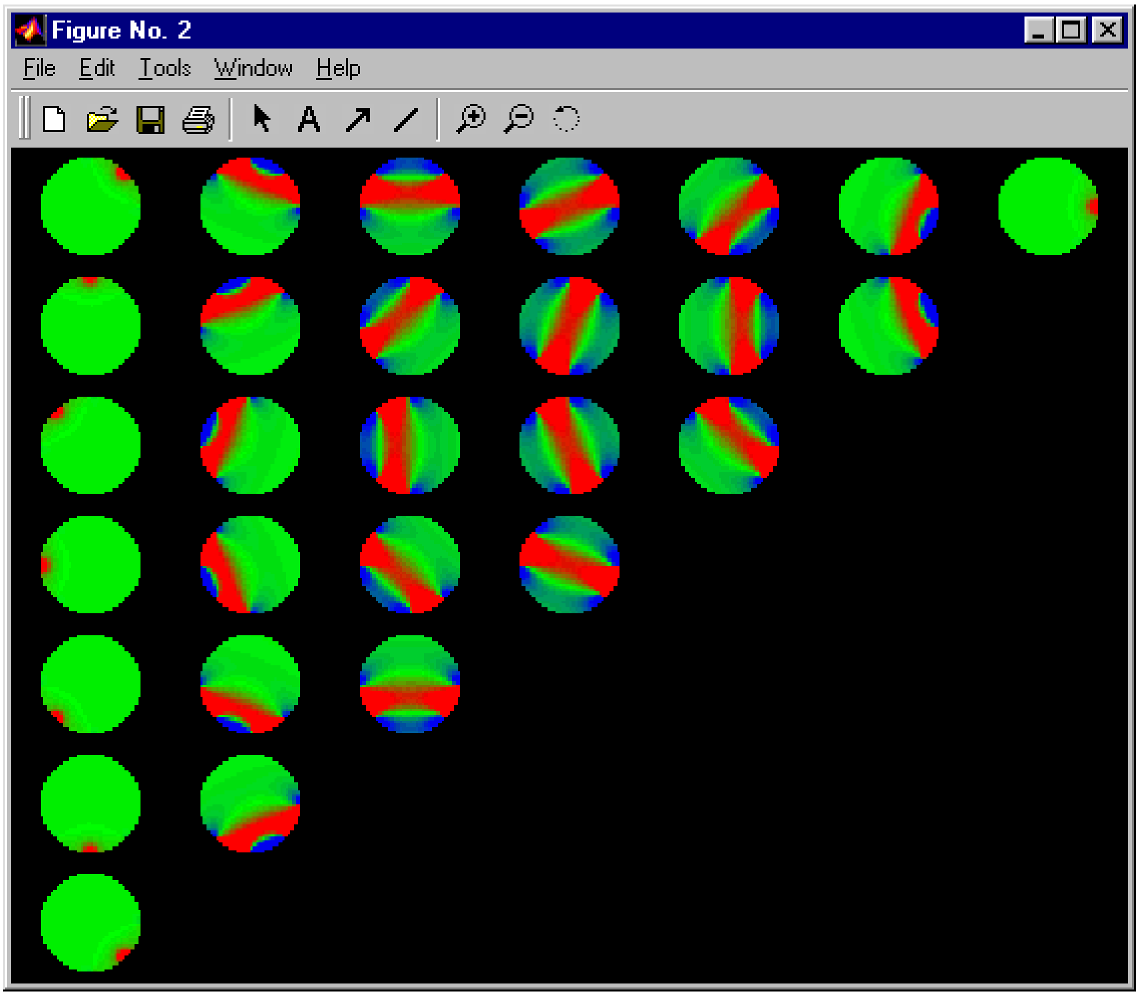


Figure 1.4. Complete set of sensitivity maps for an 8-electrode sensor with protocol 1 excitation

In this case, the sensitivity maps are plotted for each electrode pair in the following order:
C12, C13 .....C1E
C23, C24....C2E

etc

1.1.5 ACTIVE PIXEL DISPLAY MODE

If N = -2, the location of the **active pixels** in the image is displayed as shown in figure 1.5.

Active pixels **are the pixels located** inside **the sensor for sensor cross-sections where the square pixel grid does not match the sensor geometry (as is the case for circular sensors). The pixels located outside the sensor are non-active and do not take part in reconstruction.**

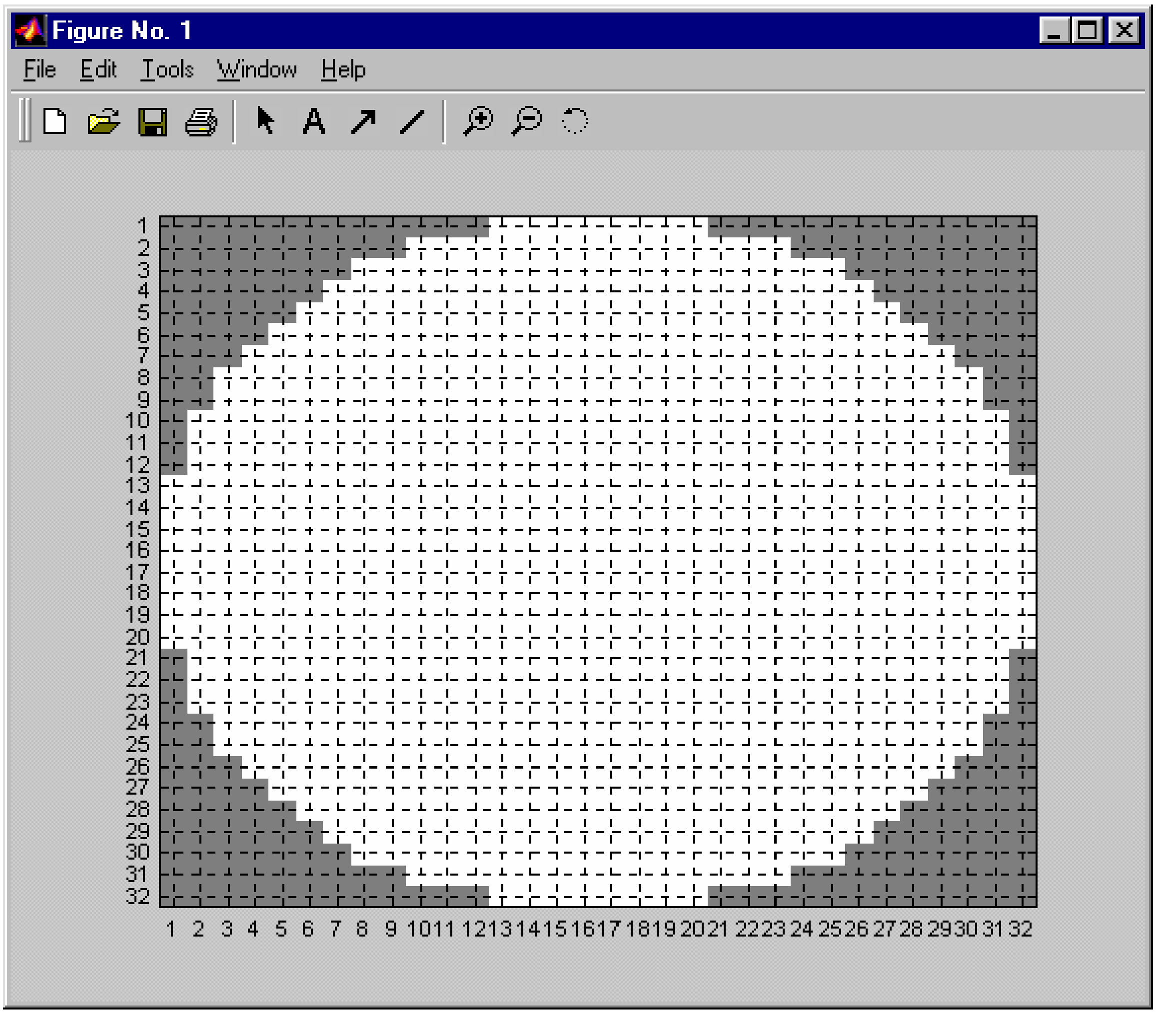


Figure 1.5. Active pixel display window

The intersection of the grid lines mark the centres of each pixel and active pixels are shown in white, while non-active pixels are shown in grey. Pixels are identified relative to the top LH corner of the grid, so the first pixel (in the top LH corner and which is non-active) has the (x,y) coordinates (1,1). The coordinates of the first active pixel (in the top row) are (13,1)

1.2 VIEWING SENSITIVITY MATRICES IN TEXT FORMAT

If a text view of the matrix is preferred, this can be achieved in a similar manner to the graphical view by typing:

ect_smapdump('E12_bin.p32',N)

where N is the required sensitivity map, counting in the order defined in section 1.1.4. (Note that N is in the range 1-28 for an 8-electrode sensor or 1-66 for a 12-electrode sensor etc.) This will produce an ascii text dump of the (typically 32 x 32 pixel) selected map in the Matlab window.

If an extra output file parameter is included in the command line, an ASCII ouput file containing the sensitivity-map data is generated in the Matlab work folder. For example, typing:

ect_smapdump('E12_bin.p32',4,'map4')

will produce a text file with the file name 'map4' of the fourth sub-map (C15).

1.3 COMPARING SENSITIVITY MATRICES

As the sensitivity maps plotted using **smapplot** do not show the relative values of sensitivity coefficients between each map, it is useful to be able to compare sets of maps.

This can be done using the program **ect_smapcomp.m**.

The program is run by typing:

ect_smapcomp('asm8_32.sif','asm8t_32.sif',4)

where the first two parameters are the two matrices to be compared and the last parameter (4) is the number of sensitivity maps to be plotted, starting with the first map. To plot the primary sensitivity maps, use 4 for an 8-electrode sensor, 6 for a 12-electrode sensor etc.

A typical output is shown in figure 1.6.

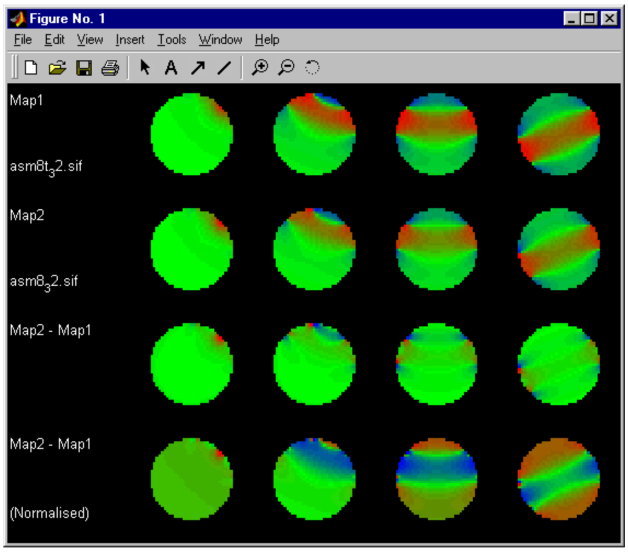


Figure 1.6. Output window for ect_smapcomp

The first row of data shows the set of primary sensitivity maps calculated for an 8-electrode sensor with internal electrodes.

The second row of data shows the same plots for a similar sensor with external electrodes.

The third row of data shows the numerical difference between these two sets of maps. Because of the large range of numbers and the limited colour scale, it is difficult to see the differences between the maps.

The fourth row of data shows the difference between the maps after normalising the maps to have the same maximum values for the highest sensitivity pixels. The relative differences are now clearer.

The information in figure 1.6 shows that the map for the external electrode sensor (map 2) has a higher sensitivity for the first electrode-pair than the map for internal electrodes, but lower overall relative sensitivities for the remaining electrode-pairs.

2. SENSITIVITY MATRIX MODIFICATION SOFTWARE

A number of programs are available to produce modified sensitivity matrices.

2.1 MATRIX CONVERSION SOFTWARE

The program **convmap.m** converts a sensitivity matrix file from **PCECT** format to **ECT32** format and vice-versa. It is used by typing:

Convmap

The user is prompted to input the filename of the matrix to be converted, followed by the name of the output file to hold the converted matrix. The following shows an example of the input required to convert an 8-electrode sensitivity matrix from **ECT32** format to **PCECT** format.

input smap:	**asm8_32.sif**
include extended geometry (y/n)?	**n**
output smap:	**e8_bin.p32**

Note that the filename format determines the direction of conversion. However, because of the extra data contained in an ECT32 sensitivity matrix, it is not possible to generate an ECT32 matrix which will show electrode positions etc. (extended geometry) using a PCECT format matrix and vice-versa. Other information such as dates and comments will also be lost in a PCECT format file. Convmap can also be used to convert an ECT32 sensitivity matrix with extended geometry into one without these features.

4.2 GENERATION OF WATER SENSITIVITY MATRICES

Sensitivity matrices for sensors with external electrodes containing water differ from the standard sensitivity matrices. When the sensor electrodes are located outside a dielectric wall, the capacitances between adjacent electrodes (and next-to adjacent electrodes if the wall is thick) will decrease in value rather than increase when the sensor is filled with water. If standard maps are used in this situation, unrealistic images are obtained. As an empirical fix to obtain images in this situation, the sensitivity coefficients in the sensitivity maps for adjacent electrodes are reduced to zero for sensitivity matrices used to image water.

Water sensitivity matrices can be generated from standard sensitivity matrices in **ECT32 format** using the conversion program **ect_watcon.m.** The new water maps can be generated in either **PCECT** or **ECT32** format, by specifying the **output file name** in the required format.

The program is run by typing:

```
ect_watcon('infile','outfile',adj)
```

where: infile is the input matrix
outfile is the modified matrix
adj is the adjacency factor (1=adjacent electrodes, 2=next adjacent ...)

Examples of usage follow:

```
ect_watcon('asm12_32.sif','wsm12_32.sif',1)
```

This generates an ECT32 - format water sensitivity matrix where the adjacent-electrode maps all contain zero sensitivity coefficients.

```
ect_watcon('asm12_32.sif','w12_bin.p32',2)
```

This generates a PCECT-format water water sensitivity matrix where both the the adjacent electrode and next-to adjacent electrode maps all contain zero sensitivity coefficients.

2.3 GENERATION OF A MATRIX WITH A LIMITED NUMBER OF PIXELS

The program **ect_modmap.m** produces a modified sensitivity matrix which only contains a few specified pixels. This matrix can then be used with the **ECT32** software to produce a limited pixel image. This is particularly useful for correlating individual pixels from images produced by a twin-plane ECT system, as the output image file contains the specified pixel values. Note that the pixels chosen must be **active pixels** inside the sensor (see section 1.1.5), otherwise an error message will be generated.

To use the program, type:

```
ect_modmap('inputmap.sif','outputmap.sif',[pixel in x,y format],'form')
```

example 1, use of form = blank

```
ect_modmap('asm8t_32.sif','asm8tp_32.sif',[16 1;1 16;32 16;16 32])
```

This produces the message:

pixels chosen :4 375 406 808

and the matrix is then output to the file asm8tp_32.sif. This matrix contains data for just 4 pixels located at (column 16, row 1), (column 1, row 16), (column 32, row 16) and (column 16, row 32). When this matrix is used to generate an image with the ECT32 software, only these selected pixels are displayed on the screen. The pixels are shown in their correct positions and with the correct sizes.

Example 2, use of form = "line" :

```
ect_modmap('asm8t_32.sif','asm8tp_32.sif',[16 1;1 16;32 16;16 32],'line')
```

This produces a similar matrix, but when used with ECT32, the selected pixels are displayed as bands of permittivity in a single line across the image. The pixels are not shown in their correct positions or size, but are displayed in pixel number order.

Example 3, use of form = "crop" :

If the pixels are located close together, the 'crop' option produces an image where the pixels are displayed in their correct relative positions, but with enhanced sizes.

```
ect_modmap('asm8t_32.sif','asm8tp_32.sif',[16 1;18 1;17 3;19 3],'crop')
```

3. IMAGE RECONSTRUCTION SOFTWARE RECON

3.1 PROGRAM INITIATION

This program constructs images from a given set of capacitance data, a forward transform and an inverse transform. The program is run by typing:

Recon

in the **matlab command** window, when the following window appears.

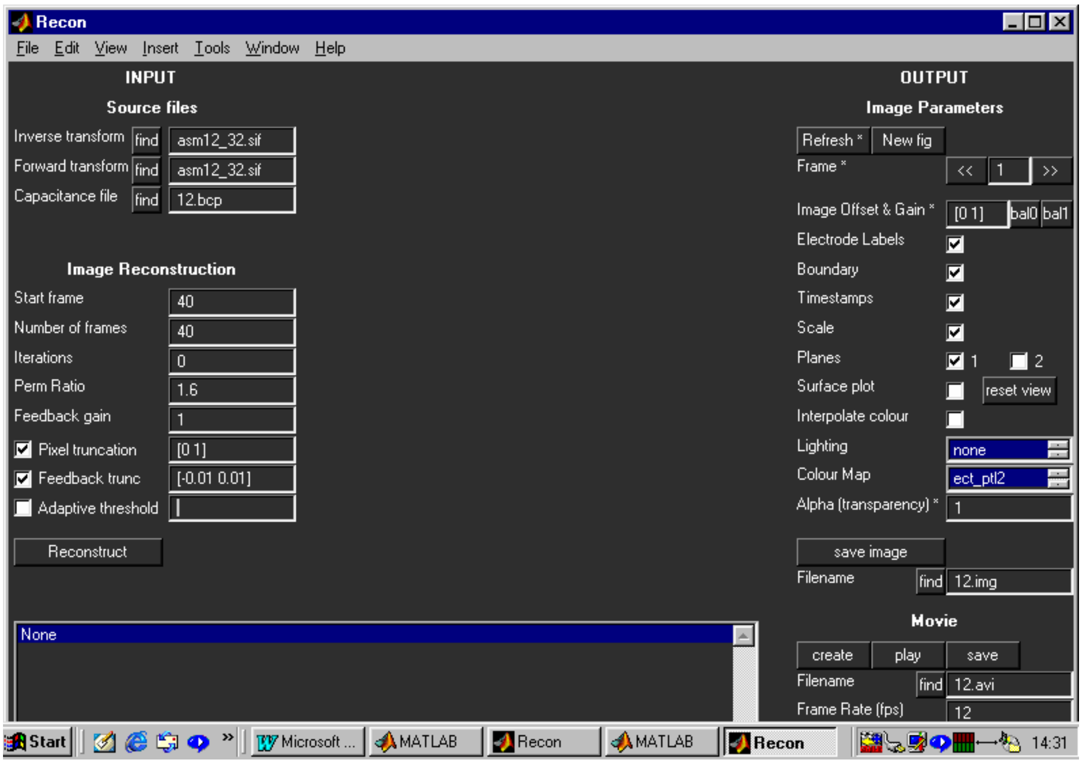


Figure 3.1 Recon startup window

The startup window contains several groups of input and output data.

The **INPUT DATA** on the left hand side of the window, containing the Source files (capacitance data file and forward and inverse transforms) and Image Reconstruction data, must be entered before an image can be reconstructed.

Note that the matrix file defined as the inverse transform will automatically be transposed within the Recon software. Consequently, if simple LBP is to be used, the sensitivity matrix file name should be entered for the inverse transform.

The **OUTPUT parameters** on the **right** hand side of the window, contained in the **Image** and **Movie** parameter groups can be entered or changed after the initial image(s) have been reconstructed.

The data formats are as follows:

3.2 INPUT DATA

3.2.1 Source Files

Inverse transform: The file name of the inverse transform to be used (for LBP use the sensitivity matrix file name)

Forward transform: The file name of the forward transform to be used (for LBP use the sensitivity matrix file name)

Capacitance file: The file name of the capacitance data to be used.

3.2.2 Image Reconstruction parameters

Start frame: The number of the first frame to be converted.

Number of frames: The total number of frames to be converted, commencing with the start frame.

Iterations: The number of iterations to be performed for each frame. (=0 for LBP). For further information, see **PTL Application Note AN4**.

Perm Ratio: The permittivity ratio (>1) of the higher and lower permittivity materials used for calibration. (Use 1 for parallel model reconstruction). If >1, the series model is used.

Gain: The gain to be used in the feedback loop of the iteration algorithm.

Pixel truncation: The lower and upper limits of normalised permittvity to be used in the iteration algorithm. These two parameters must be inside square brackets, with a single space between the upper and lower permittivity limits. Pixel truncation is enabled by ticking the box.

Feedback truncation: The truncation values to be applied to the values of normalised error capacitance fed back in the iteration algorithm (in the range -1 to 1). These two parameters must be inside square brackets, with a single space between the upper and lower capacitance limits. Feedback truncation is enabled by ticking the box.

Adaptive threshold: This feature is included as a simple technique for reducing the spread of the image caused by the LBP algorithm. If this feature is enabled, the pixels in the image with values below a specified level T are set to zero. The value of T is given by the formula:

$$T = (1 - X\,.C_{av}).K_{av}$$

Where:
C_{av} is the average value of the measured normalised capacitances
K_{av} is the average value of the normalised pixel permittivities.
X is a thresholding factor chosen by the user.

Once the input data has been entered, the image(s) can be reconstructed by clicking on the generate button. An information window is shown at the bottom of the main window and displays the current status of the reconstruction process. A typical image is shown in figure 3.2 below.

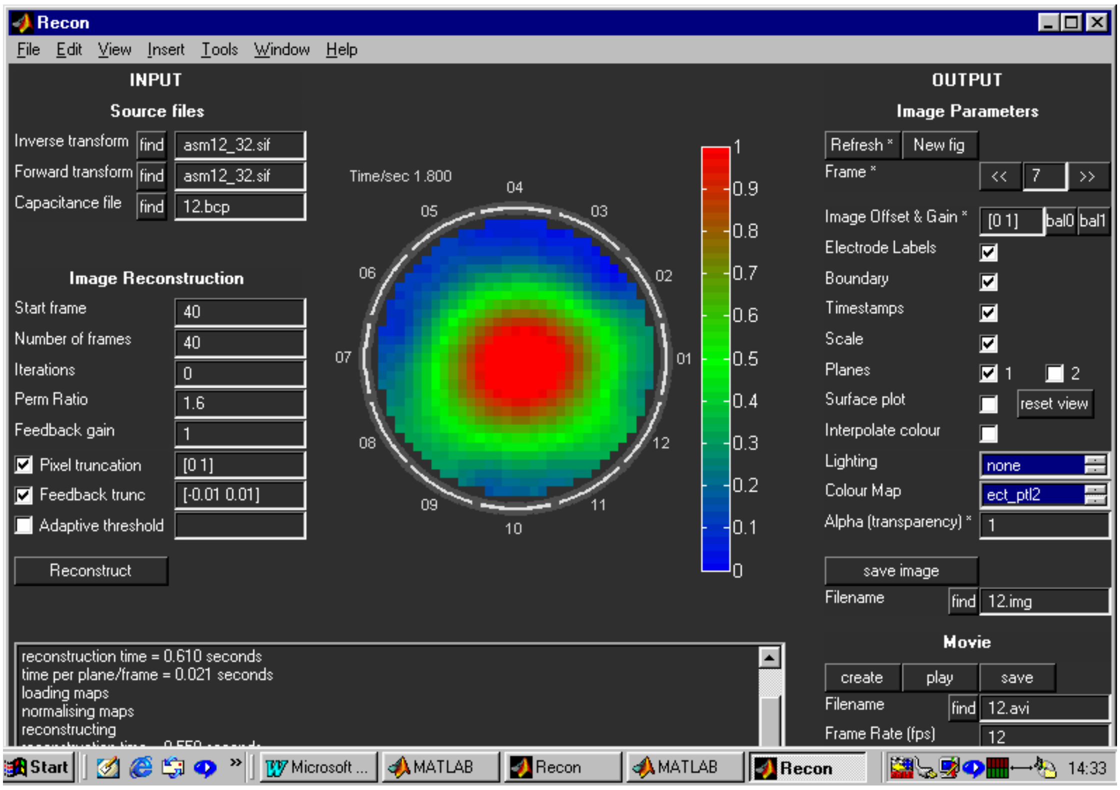


Figure 3.2 Recon window showing 2-D image

3.3 OUTPUT PARAMETERS

Once the images have been reconstructed, they can be viewed by using the **controls** and by modifying the **parameters** on the right hand (**OUTPUT**) side of the window.

The **Image parameters** control the image format. Many of the parameters affect the image immediately, but some of the parameters, indicated by a * symbol require the **Refresh*** button to be clicked before the updated image is displayed.

3.3.1 Image display parameters and controls

Refresh* Button:	Updates the displayed image after changing a * parameter.
New fig. Button:	Displays the image in a new window for printing/editing operations
Frame number box:	The frame number is displayed in this box. A new frame can be displayed directly by entering the new frame number in this box and clicking on the Refresh* button.

Frame buttons:	<< and >> select previous and next frames. Note that there is a short delay before the new frame, together with its frame number, is displayed.
Image Offset and Gain:	The offset and gain factors to be applied to the image. These parameters must be contained within square brackets with a space between the parameters. For no offset and unity gain, use [0 1].
Bal0 button	Displays image offset symmetrically around the lower permittivity calibration point for the selected gain setting.
Bal1 button	Displays image offset symmerically around the upper permittivity calibration point for the selected gain setting.
Electrode labels:	Electrode numbers displayed if ticked.
Boundary:	Sensor boundary and electrodes displayed if ticked.
Timestamps:	Frame time data displayed if ticked.
Scale:	Permittivity scale displayed if ticked.
Planes:	Displays the image for the selected data plane.
Surface plot:	Enables 3D views if ticked. See Matlab manual for information on viewing 3D images. (Click on image and drag to rotate in 3D)
Reset view button:	Resets image to 2D plot following 3D display
Interpolate colour:	Smooths image if ticked.
Lighting:	Changes image display format shadowing according to selected option.
Colour map:	Changes the colour scale used to display the image. Default is RGB (ect_ptl2). Other useful scales are binary (ect_bin) and spectrum (jet).
Alpha (transparency):	Changes opaqueness of image. (Matlab 6 only feature)

3.3.2 Image data file

The reconstructed image data can be saved as an ASCII image file by defining a suitable name in the **file name** box. The **find** button acts as a **browse** button to specify the file name and folder.

3.4 MOVIE FILES

The reconstructed images can be converted into a sequence of images which then be can be re-played continuously. If the **recon** program is run under **Matlab6**, this data can also be saved as a file in .avi format.

The **movie file** is generated as follows:

1. Once a set of images has been reconstructed, click the **create** button. This generates a window, displaying the first frame in the sequence, as shown below:

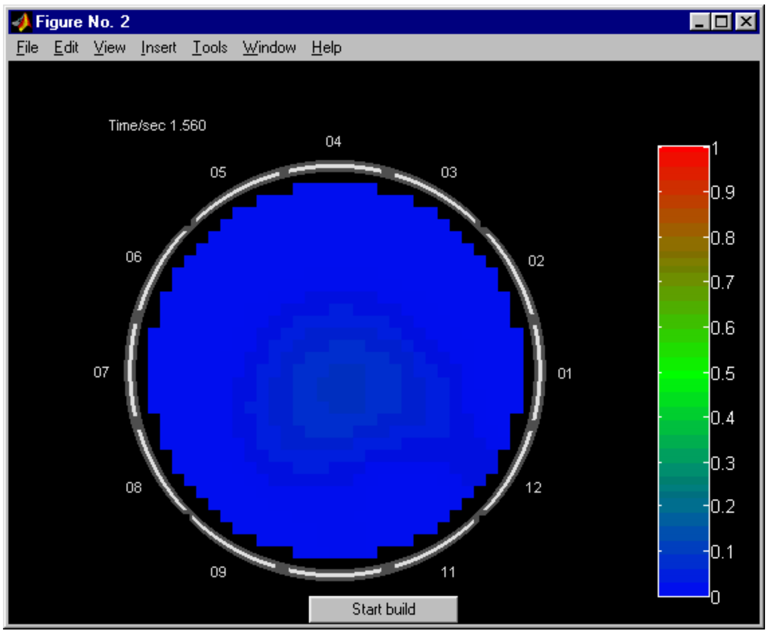


Figure 3.3 Movie creation window

This window should be re-set to a convenient size and the **Start build** button can then be clicked. The movie file will then be built and a **finished** button will appear at the bottom of the window.

Click the **finish** button, when the standard **Recon** window will again be displayed.

The image data can then be replayed by clicking the **play** button.

To save the movie image **file** (Matlab6 **only), click on the save button. An .avi file will be generated which can then be viewed with suitable media replay software.**

4. IMAGE RECONSTRUCTION PROGRAM MASKRECON

The program **maskrecon.m** reads a twin-plane .bcp capacitance file and a sensitivity matrix file and produces two image data files containing only a limited number of pixel values, where the pixels to be listed are defined by the user.

4.1 PROGRAM INITIATION

Type: **`maskrecon`**

This produces the following input dialogue:

» maskmap

sensitivity matrix filename (asm8_32.sif)	*The sensitivity matrix filename*
mask [x1 y1;x2 y2; ....] ([16 16;17 17])	*The pixels to be listed (cartesian coordinates)*
filename (tp1.bcp)	*The input capacitance data file*

A file information message is then generated in the following format:

file info: frames =125 ave rate =25.2016fps,Created by:ECT32 for Win32 Version 1.11 (Sep 28 1999 23:44:13),by:MB,on:08:40:47 GMT Daylight Time, 04Oct99:Contents of capture mode buffer

The input dialogue then continues as follows:

start frame (2)	*The number of the first frame of data*
number of frames (100)	*The number of frames to be processed*
perm ratio [series model] (1)	*Series permittivity model used if >1*
iterations (0)	*If 0, uses LBP, if >0 gives iterated image data*
gain (1)	*The feedback gain used for reconstruction*
image filename (image)	*The filename for the output image data*

Note. The program produces two separate image files with the names **filename1** and **filename2**. It also generates a modified sensitivity matrix with the file name mask.sif provided that no iteration is selected.

4.2 SAMPLE MASKRECON PROGRAM OUTPUT

The program produces the following output data:

loading maps
pixels chosen :390
normalising maps
reconstructing
reconstruction time = 7.520 seconds
time per plane/frame = 0.303 seconds
The program produces 2 output image data files, in this case, **image1** and **image2,** corresponding to the pixels in planes 1 and 2 respectively.

The contents of part of **image1** are as follows:

0.9980 0.9966
0.9614 0.9586
0.9406 0.9379
0.9244 0.9214
0.9176 0.9152
0.9038 0.9019
0.8822 0.8807
0.8457 0.8464
0.7776 0.7805
0.6617 0.6658
0.5021 0.5061
0.3480 0.3523
0.2003 0.2042
0.0747 0.0761
0.0068 0.0073
0.0000 0.0000
0.0000 0.0000

ie two columns of figures corresponding to the 2 pixels selected. The data in **image2** is similar and hence these 2 files can then be correlated as required using suitable external software.

5. CAPACITANCE DATA FILE PROCESSING SOFTWARE

Two programs are available for modifying captured capacitance data.

5.1 CAPACITANCE DATA INTERPOLATION

Program **interp.m interpolates** and **de-skews** capacitance data sets to **an arbitrary capture rate**. This is useful for correcting **timing errors** which can occur during data capture, both because of problems within the MS Windows operating system and also inherent problems caused by capturing a frame of data as a sequential operation. It can be used with either single-plane or twin-plane data files and can also be used to generate extra frames of data between actual captured frames or to reduce the framerate. For example, it is possible to generate an output data file containing 50 frames per second from an input data file recorded at 25 fps. It is also possible to time shift the plane 2 data relative to the plane 1 data (see section 5.1.1).

The program is run by typing:

`interp`

which generates the following input dialogue:

Interp

input filename (record_2.bcp) tp1.bcp *Name of input data file*

file info: frames =125 ave rate =25.2016fps,Created by:ECT32 for Win32 Version 1.11 (Sep 28 1999 23:44:13),by:MB,on:08:40:47 GMT Daylight Time, 04Oct99:Contents of capture mode buffer

output filename(tp1_interp.bcp)	*Name of output file*
start frame (1550)	*Initial frame*
number of frames (100)	*Number of frames to be converted*
long settle time (us) (450)	*DAM200 hardware timing parameter*
short settle time (us) (35)	*DAM200 hardware timing parameter*
output interval (ms) (39.68) 40	*Frame interval for output file*
timeshift (ms) (0)	*See section 5.1.1*
interp mode (cubic)	*See Matlab **interp1** function for options*
plot channel [0=no plotting] (6)2	*Capacitance channel to be plotted*
compare mode [off|on|diff] (diff)	* see below

Current plot held

Generated:100 frames in:tp1_interp.bcp»

This generates the required data file, (as a binary capacitance file) and also plots the selected capacitance channel if plotting has been selected. Figure 5.1 shows the plot output for a twin-plane capacitance data file where the number of output frames is twice the number of input frames. The original measured data points are shown as o and the corrected/interpolated points as +. Data is shown in black for plane 1 and blue for plane 2. The horizontal scale is in milliseconds. The output data file can be viewed using either the ECT32 software or the **extractchan** software (see section 5.2).

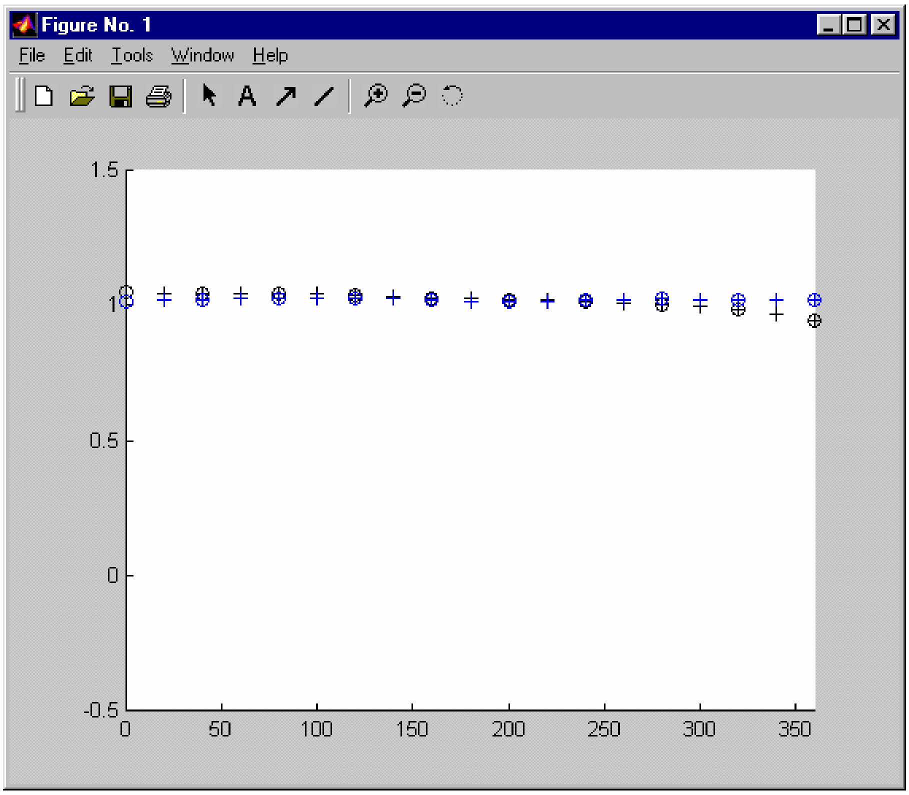


Figure 5.1 Plot output

The operation of compare mode is as follows:

If a single-plane file is used, the output file contains the following data according to the compare mode selected:

Compare mode status	Output data plane 1	Output data plane 2
Off	Interpolated data	N/A
On	Original data	Interpolated data
Diff.	Original data	Difference between original and interpolated data.

If a twin-plane data file is used, the output file contains the following data.

Compare mode status	Output data plane 1	Output data plane 2
Off	Interpolated data P1	Interpolated data P2
On	Original data P1	Interpolated data P2
Diff.	Original data P1	Difference between original data for P1 and interpolated data for P1

5.1.1 USE OF INTERP TO TIME SHIFT THE PLANE 2 DATA.

If the data file is from a twin-plane sensor, it is possible to time-shift the plane 2 data relative to the plane 1 data using interpolation. This is done by including a time shift parameter in the input dialogue. If the time shift parameter in milliseconds is positive, the plane 2 data is delayed by the set number of milliseconds, while if the parameter is negative, it is advanced by the set number of milliseconds. If the modified data file is viewed in twin-plane mode, using eg the ECT32 software, this facility can be used to carry out simple correlation tests on the captured twin-plane data.

5.2 EXTRACTION OF DATA FOR A SINGLE CHANNEL

A second program, **extractchan.m,** extracts a set of capacitance data in ASCII format for a single capacitance-pair from a file of measured capacitance data in binary format.

To run the program type :

```
Extractchan
```

which generates the following input dialogue:

filename (record_2.bcp)	*The input data file name*
start frame (550)	*The first frame to be extracted*
number of frames (200)	*The number of frames to be extracted*
channel (6)	*The required measurement channel (eg in range 0- 66 for 12 electrodes)*
output filename (chan6)	*The output data file name*

The output file can then be viewed with a suitable ASCII text editor or word-processor.

6. INTER-ELECTRODE CAPACITANCE CALCULATION SOFTWARE

The program **ect_cap.m** calculates inter-electrode capacitances (due to the capacitive coupling **inside** the sensor only) for a given sensor geometry when the sensor is filled with lower and higher permittivity materials respectively.

To run the program, type:

`ect_cap`

This produces the following input dialogue:

electrodes? 12	*The number of electrodes*
elec width (0-1) ? 0.8	*Electrode arc span as decimal fraction of maximum*
guard width? 0.1	*Axial guard span as decimal fraction of maximum*
wall radius (0-1)? 0.9	*Normalised inner wall radius*
electrode length (mm)? 100	*Axial length of electrode*
perm void 1? 1	*Permittivity of lower permittivity material*
perm void 2? 3	*Permittivity of higher permittivity material*
perm wall? 2	*Permittivity of tube wall*

This then produces the following output:

Meas	Clow (fF)	Chigh (fF)	Ch-Cl (fF)	Ch/Cl
(1,2)	311.2	486.8	175.6	1.564
(1,3)	44.4	118.4	73.9	2.664
(1,4)	20.6	59.0	38.4	2.867
(1,5)	13.4	39.4	25.9	2.934
(1,6)	10.7	31.7	21.0	2.960
(1,7)	10.0	29.5	19.6	2.966

Where:

Clow is the capacitance in fF between the electrode-pair when the sensor is filled with the lower permittivity material .

Chigh is the capacitance in fF between the electrode-pair when the sensor is filled with the lower permittivity material.

Ch-Cl is the difference between these capacitances.

Ch/Cl is the ratio of these capacitances.

Note that for a sensor with no wall, the ratio Ch/Cl is equal to the permittivity ratio of the higher and lower permittivity materials (in this case, 3).

7. GENERATION OF ENHANCED INVERSE TRANSFORMS

It is possible to calculate **enhanced inverse transforms (tranformation matrices)** which give better quality images than those produced by LBP, (which uses the simple transpose of the sensitivity matrix as the inverse transform).

A number of different **transformation matrices** can be used, but two methods which give useful improvements over back-projection are based on methods originally described by **Landweber** and **Tikhonov**. The **Landweber** method gives similar results to the iterative method used in the **IU2000** software when the **pixel truncation facility is disabled in the IU2000 software**. Both the **Landweber** and **Tikhonov transformation matrices** can be obtained from the **sensitivity matrix** for the sensor and this section describes how this can be achieved using two custom Matlab programs, **ect_landmap.m** and **ect_tikmap.m** The operation carried out by the matlab code in the two m files is known by mathematicians as **regularisation**.

7.1 LANDWEBER INVERSE TRANSFORM

7.1.1 DEFINITION OF LANDWEBER TRANSFORM

The **transformation matrix** $\mathbf{Q_L}$ used in **Landweber's** method can be derived from the **sensor sensitivity matrix S** by defining a **transform parameter L** and an **iteration parameter N** (which defines the number of iterations).

$\mathbf{Q_L}$ is defined in equation 7.1 as follows:

$$\mathbf{Q_L = V \,.\, F\,(\,W, t, N\,)\,.\, U'} \qquad \textbf{(7.1)}$$

where:
V, W and **U** are the matrices obtained by applying the **Single Value Decomposition** (**SVD**) process to the sensitivity matrix **S.** This operation produces a diagonal matrix **W** of the same dimensions as **S**, and unitary matrices **U** and **V,** so that **S = U . W . V** .

F is the **SVD** filter function matrix defined in equation **A1.2**.

$$\mathbf{f = (\,(1 - (\,1 - \,t\,.\,w\,)^N\,)\,/\,w} \qquad \textbf{(7.2)}$$

where:

f is one element of the filter matrix **F**

w is one element of the diagonal matrix **W**

t is a relaxation parameter (referred to in the main text as the Landweber transform parameter L).

N is the number of iterations.

Typical safe values for **L** are in the range 0.01 to 0.0001. Experience shows that high values of **L** (0.01) can give rise to spurious artefacts around the edges of the image, while low values of **L** appear to give results similar to those obtained using the simple **LBP algorithm**. Typical practical values for the number of iterations **N** are in the range 10 to 100.

7.1.2 GENERATION OF LANDWEBER TRANSFORM

The Landweber **transformation matrix** is generated using the Matlab M file **ect_landmap.m.**

The **input (sensitivity matrix) files** must exist in the **working folder** before the program is run. The **output files (the new transformation matrices)** will be generated in the same folder and must then be copied to the appropriate **ECT or PCECT folder** before they can be used, as described later in section 7.1.4.

The steps involved to generate a **Landweber transformation matrix** Q_L are as follows:

1. Copy the input **sensitivity matrix** file to the **working folder.**

2. Run the program **ect_landmap.m** by typing:

```
ect_landmap('inputfilename','outputfilename',L,N)
```

where:

inputfilename is the sensor **sensitivity matrix** file name: eg **E12_bin.p32** for **PCECT** files or **NSM12_32.sif** for **ECT32** files.

outputfilename is the name of the output **transformation matrix**.

Note that:

1. The **output (transformation matrix) file** will be in the format implied by the output file name (see below), so separate **transformation matrices** must be generated for use with each of the **PCECT** and **ECT32** software versions.

2. For files in **ECT32** format, the filename can be descriptive eg **L12_50_0.001.sif** for a **Landweber transformation matrix** with (N=) 50 iterations and a transform factor of 0.001 in **ECT32** format. However, file names in **PCECT** format must follow the filename convention for **PCECT** sensitivity matrix files, because of limitations in the **PCECT** software.

L is the transform parameter in the range 0.01 to 0.0001.

N is the number of iterations.

So some typical command lines would be:

Example 1

```
ect_landmap('E12_bin.p32','L12_bin.p32',0.01,100)
```

This will convert a **PCECT**-format **sensitivity matrix** into a 100-iteration **Landweber transformation matrix,** also in **PCECT** format, with a transform factor of 0.01.

Example 2

`ect_landmap('NSM12_32.sif','L12_100_0.01.sif',0.01,100)`

This will convert an **ECT32** format **sensitivity matrix** into a 100 iteration **Landweber transformation matrix** in **ECT32** format with a transform factor of 0.01.

Example 3

`Ect_landmap('NSM12_32.sif','L12_bin.p32',0.01,100)`

This will convert an **ECT32** format **sensitivity matrix** into a 100 iteration **Landweber transformation matrix** in **PCECT** format with a transform factor of 0.01.

7.1.3 FILE FORMATS

The transformation matrix generated is in the form of an equivalent forward transform, that is, it is the transpose of the true inverse matrix. This allows the transformation matrix to be plotted in the form of an equivalent sensitivity matrix. The transformation matrix may be used as an inverse transform directly in all PTL software, as it is re-transposed into the correct format by the software which uses it.

7.1.4 USE OF THE LANDWEBER TRANSFORMATION MATRIX

The action needed to use the **new transformation matrix** to generate images depends on the software version in use. As the procedure for the **ECT32 software** is simpler, this will be described first.

7.1.4.1 USE WITH ECT32

1. Simply copy and/or rename the new **transformation matrix** file to the **ECT32_1.00\configure** folder. It must have a file name in the format: **name_32.sif** in the **ECT32** folder.

2. Run the **ECT32** software and select the required **transformation matrix** file in the **Sensor information file box** in the **Configuration window**.

3. When the **ECT32 program** is run, images will be constructed using the new **transformation matrix** (.sif file).

7.1.4.2 USE WITH PCECT

Because of the limited functionality of the older **PCECT** software, the procedure for using different **sensitivity matrices** or **transformation matrices** is more complex. It is necessary to create a **unique sub-folder** for each new category of **sensitivity matrix** or **transformation matrix** and to select the appropriate **sub-folder** using the **Mapsel.exe** program via the **Select Map Type** icon in the **PCECT program group window**. The detailed steps are as follows:

1. Create a **new sub-folder** in the **c:\PCECT\sensmaps** folder and give it an appropriate name (eg **Landweber**).

2. Copy all of the **sensitivity matrices** from the **standard sensitivity matrix folder** to the **new sub-folder**. (This is necessary because the sensitivity matrix selection program **Mapsel.exe** needs a complete set of maps before it can be used.)

3. Replace the existing relevant **sensitivity matrix file** with **the new transformation matrix file** by deleting the appropriate **sensitivity matrix file** in the **new sub-folder** (eg **E12_bin.p32**). Rename the transformation matrix file to that of the file which you have just deleted (in this case, **E12_bin.p32**).

For example, for a 12-electrode 32x32 pixel sensitivity matrix, copy the new file to **E12_bin.p32**. This rather convoluted procedure is necessary because of the limited functionality for the use of different **sensitivity matrix** or **transformation matrix** files within the **PCECT** software.

4. Open the **PCECT program group window** and select the **new sensitivity matrix folder** using the **Select Map Type** icon.

5. Run the **PCECT software**. The appropriate **sensitivity matrix** or **transformation matrix** in the new sensitivity matrix folder will now be used to construct the image.

7.1.5 PHYSICAL BASIS OF THE LANDWEBER TRANSFORM

We can currently offer no physical insight into how the Landweber transform works. The transform is well-known to mathematicians working on image enhancement, but we have failed to receive a convincing explanation to-date. If any users can give any physical insight into the origins of this transform and explain why it works in physical or simple mathematical terms, please contact us at enquiries@tomography.com.

7.2 TIKHONOV TRANSFORM

7.2.1 DEFINITION OF THE TIKHONOV TRANSFORM

The **transformation matrix** $\mathbf{Q_T}$ used in **Tikhonov's** method can be derived from the **sensor sensitivity matrix** by defining a single transform parameter **T**.

$\mathbf{Q_T}$ is defined in equation 7.3 as follows:

$$\mathbf{Q_T = S^T . (S . S^T + } t \mathbf{. I)^{-1}} \quad \textbf{(7.3)}$$

where:

S is the sensitivity matrix
$\mathbf{S^T}$ is the transpose sensitivity matrix
t is the Tikhonov transform factor (referred to in the main text as T).
I is the identity matrix (the m x m matrix with 1s on the diagonal and zeroes elsewhere)
m is the number of unique capacitance measurements

Typical safe values for **T** are in the range 0.1 to 100. Low values of **T** yield very noisy higher-definition images while higher values of **T** produce images similar to those produced by LBP.

7.2.2 GENERATION OF THE TIKHONOV TRANSFORM

The **transformation matrix** is generated using the Matlab M file **ect_tikmap.m.**

Again the **input (sensitivity matrix) files** must exist in the **working folder** and the **output files (the new transformation matrices)** will also be generated in this folder. The output files must then be copied to the appropriate **ECT folder** before they can be used, as described later.

The steps involved to generate a **Tikhonov transformation matrix** $\mathbf{Q_T}$ are as follows:

1. Copy the **sensitivity matrix** file from the **appropriate ECT folder** to the **working folder.**

2. Run the program **ect_tikmap.m** by typing:

```
ect_tikmap('inputfilename','outputfilename',T)
```

where:

inputfilename is the sensor **sensitivity matrix** file name: eg **E12_bin.p32** for **PCECT** files or **NSM12_32.sif** for **ECT32** files.

outputfilename is the required output **transformation matrix**.

The same comments given in section 7.1. apply to the file names in this section.

So some typical command lines would be:

Example 1

```
ect_tikmap('E12_bin.p32','T12_bin.p32',1)
```

This will convert a **PCECT**-format **sensitivity matrix** into a **Tikhonov transformation matrix** in **PCECT** format with a transform factor of 1.

Example 2

```
Ect_tikmap('NSM12_32.sif','T12_10.sif',10)
```

This will convert an **ECT32** format **sensitivity matrix** into a **Tikhonov transformation matrix** in **ECT32** format with a transform factor of 10.

Example 3

```
Ect_tikmap('NSM12_32.sif','T12_bin.p32',1)
```

This will convert an **ECT32** format **sensitivity matrix** into a **Tikhonov transformation matrix** in **PCECT** format with a transform factor of 1.

7.2.3 USE OF THE TRANSFORMATION MATRIX

The action needed to use the new transformation matrix to generate images is similar to that for the **Landweber** method as described in section 7.1. The matrix generated is again the transpose of the true inverse matrix.

7.2.4 PHYSICAL BASIS OF THE TIKHONOV TRANSFORM

In contrast to the **Landweber** transform, it is fairly straightforward to explain where the **Tikhonov** transform comes from. We are indebted to Dr Andrew Reader of UMIST (DIAS) for the following explanation:

In the **LBP** method, the forward and inverse transforms are defined by the two matrix equations:

$$\mathbf{C} = \mathbf{S}.\mathbf{K} \qquad (7.4)$$

$$\mathbf{K} = \mathbf{S}^{\mathbf{T}}.\mathbf{C} \qquad (7.5)$$

where the forward transform (equation 7.4) is assumed to be accurate, but the inverse transform (equation 7.5) is known to be inaccurate.

Let **K** be the actual (physical) permittivity distribution and $\mathbf{K_{BP}}$ be the (erroneous) distribution calculated from the capacitance measurements **C** using equation (7.5).

Then $$\mathbf{K}_{BP} = \mathbf{S^T.C} \quad (7.6)$$

Substituting for **C** using equation 7.4 we obtain:

$$\mathbf{K}_{BP} = \mathbf{S^T.S.K} \quad (7.7)$$

Hence $$\mathbf{K} = \mathbf{K}_{BP}.\ (\mathbf{S^T.S})^{-1} \quad (7.8)$$

and substituting for $\mathbf{K}_{BP}$ from equation (7.6) we obtain:

$$\mathbf{K} = \mathbf{S^T.C}.\ (\mathbf{S^T.S})^{-1} \quad (7.9)$$

giving $$\mathbf{K} = \frac{\mathbf{S^T.C}}{\mathbf{S^T.S}} \quad (7.10)$$

That is, the true permittivity distribution **K** is obtained by using the inverse transform $\frac{\mathbf{S^T}}{\mathbf{S^T.S}}$ instead of the simple transpose matrix $\mathbf{S^T}$.

However, the matrix $\mathbf{S^T.S}$ has squared sensitivity coefficients along its primary diagonal and if these coefficients are small, their squared values will be very small. There is a clear danger that dividing the matrix $\mathbf{S^T}$ by this denominator matrix $\mathbf{S^T.S}$ will result in some divisions by very small numbers or even by zero. Consequently, the primary diagonal elements of the denominator matrix are padded out with constants to prevent this happening by adding a scaled identity matrix **I** to the denominator.

That is equation (7.10) becomes:

$$\mathbf{K} = \frac{\mathbf{S^T.C}}{\mathbf{S^T.S} + t.\mathbf{I}} \quad (7.11)$$

where t is a scalar constant (the Tikhonov constant) and **I** is the identity matrix (a matrix with ones along the primary diagonal and zeroes elsewhere). As can be seen, equation 7.11 is identical to equation 7.3 (section 7.2.1) which defines the Tikhonov transform.

In physical terms, what we are doing is trying to correct the effect of the erroneous matrix $\mathbf{S^T}$ (which causes low-pass spatial filtering of the permittivity distribution, removing all of the fine detail) by the use of a complementary transform (which has the effect of passing the data through an amplified high-pass filter). However, there is a limit to how much gain can be used without simply amplifying the noise in the low-pass-filtered capacitance measurements. The identity matrix and scaling factor effectively limit the gain which is applied to the correcting transform. Larger values of t will have a large moderating effect while small values of t will have minimal effect. The choice of t is therefore critical to avoid spurious artefacts in the reconstructed image.

8. SOME EXAMPLES OF PROGRAM OUTPUTS

We have used some of the programs described in this manual to generate some sample output files for an 8-electrode sensor. We have done this for a number of image reconstruction methods, namely, simple LBP, LBP using the Landweber and Tikhonov inverse transforms, iterative LBP and finally, an iterated version of the Tikhonov method.

In each case, we show the **complete set of primary inverse transforms (or sensitivity matrices)** where appropriate**,** plotted using the program **ect_smapplot.m**. We then show the **image of a circular rod** in the appropriate sensor, constructed using the program **recon.m** when the specified **transform** is used as the **inverse transform** and the **standard sensitivity matrix** is used for **the forward transform**.

8.1 RESULTS OBTAINED USING LINEAR BACK-PROJECTION

Figure 8.1.1 shows the **primary** sensitivity maps for each unique electrode-pair for an 8-electrode sensor with external electrodes and a wall thickness of 10% of the tube diameter. The electrodes are shown in their conventional positions. The first map corresponds to electrode pair 1-2, where electrode 1 is at 3 o' clock and electrode 2 is at 2 o' clock etc.

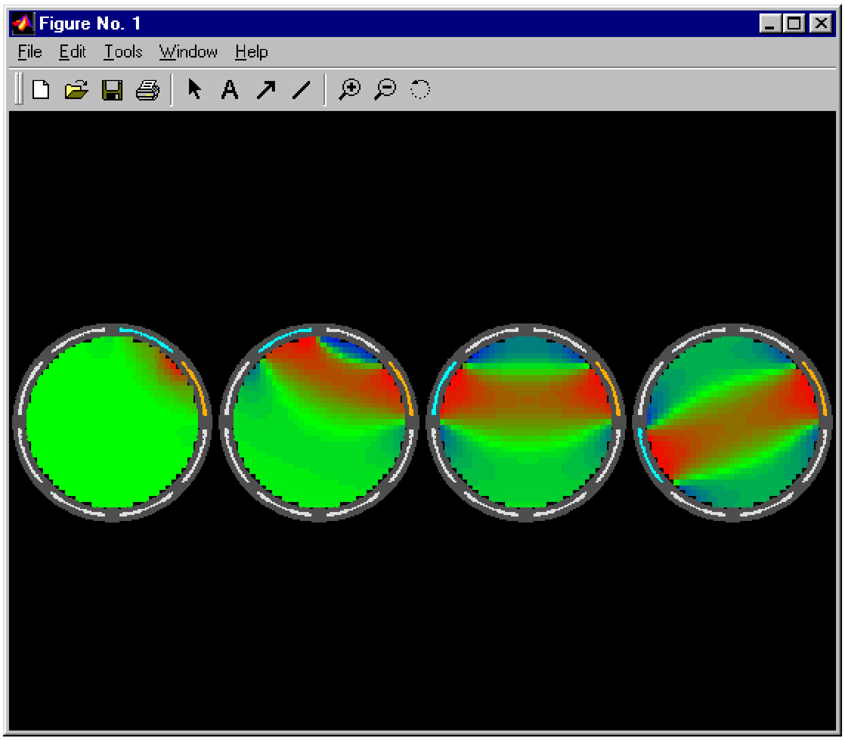


Figure 8.1.1 Primary sensitivity maps for 8-electrode sensor

Figure 8.1.2 is a detailed view of the **sensitivity map** for the electrode-pair 1-5 (**opposite electrodes**) and is an enlarged view of the last sensitivity map. Note that there is only one **main band** where the sensitivity coefficients **are positive** (red) in the **basic sensitivity maps.**

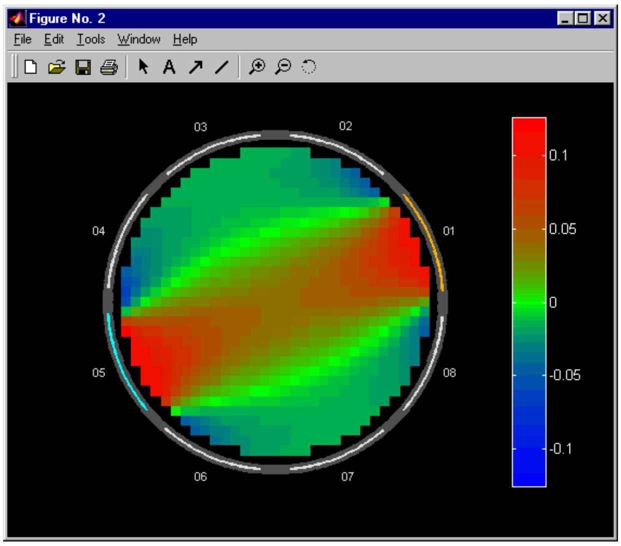


Figure 8.1.2 Sensitivity map for electrode-pair 1-4

Figure 8.1.3 displays the result of constructing the image using LBP with the program **recon.m** and shows a relatively indistinct, large image.

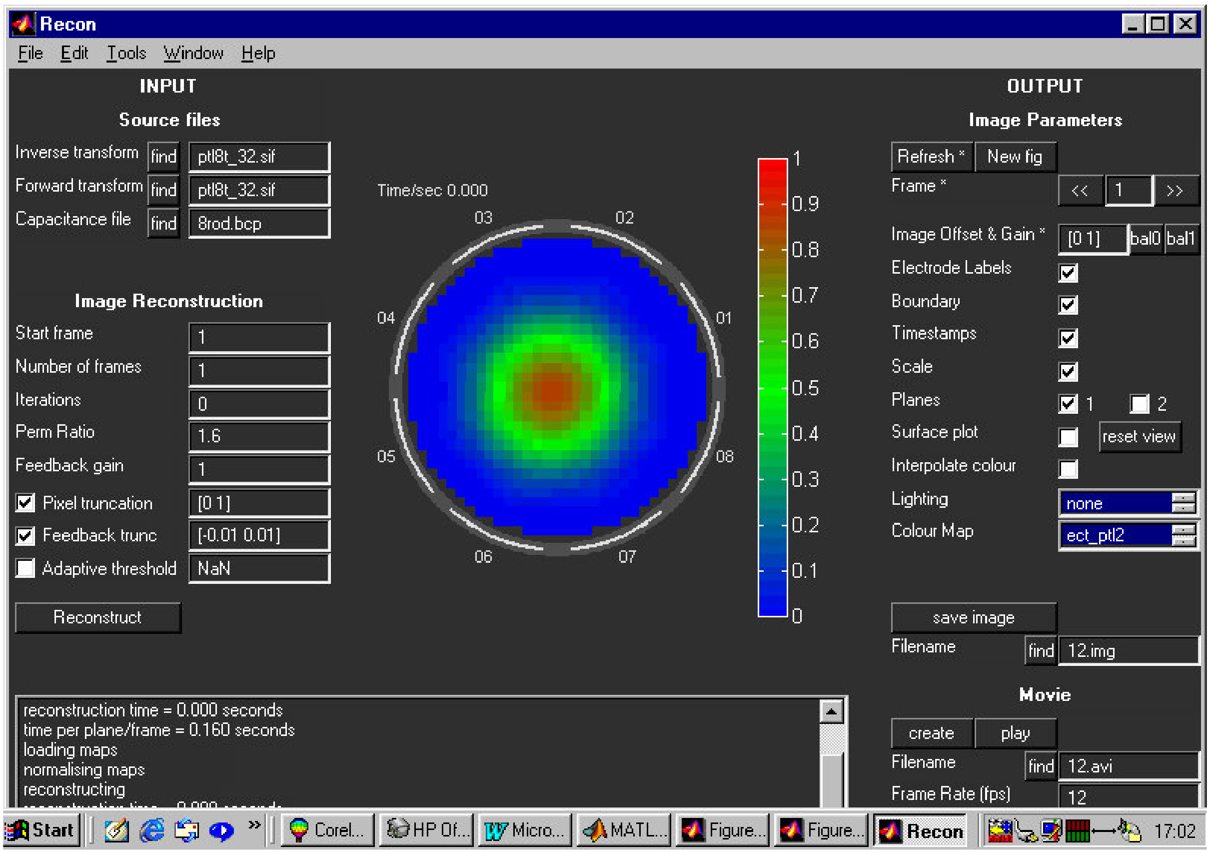


Figure 8.1.3 Reconstructed image using LBP

8.2 RESULTS OBTAINED USING LANDWEBER INVERSE TRANSFORM

Figure 8.2.1 shows the transpose of the **primary Landweber inverse transforms** (N=100, L = 0.01) for each unique electrode-pair for an 8-electrode sensor with external electrodes and a wall thickness of 10% of the tube diameter. The transforms are plotted as equivalent sensitivity matrices and the electrodes are shown in their conventional positions. The first transform corresponds to electrode pair 1-2, where electrode 1 is at 3 o’ clock and electrode 2 is at 2 o’ clock etc.

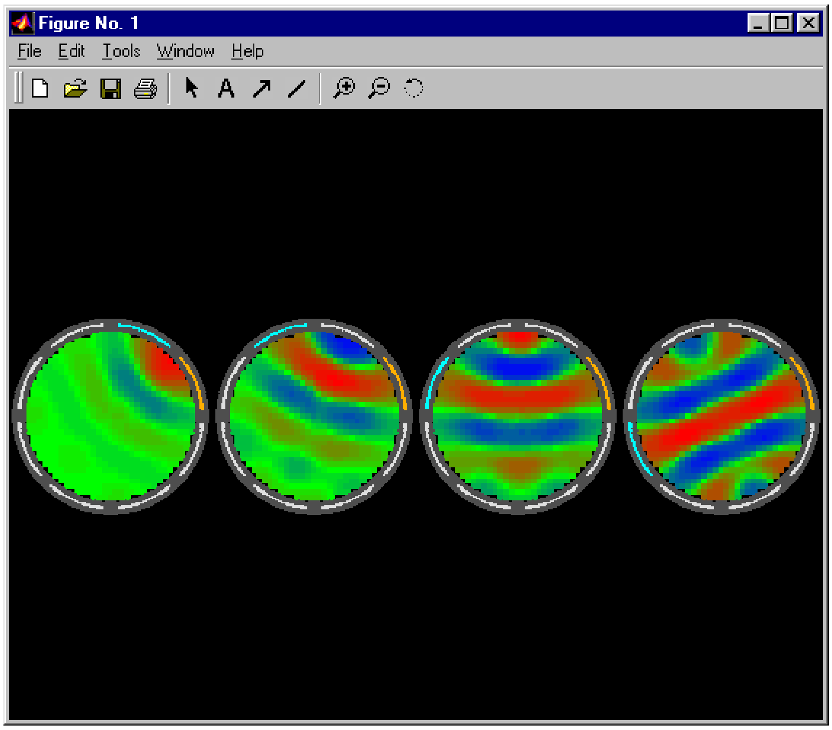


Figure 8.2.1 Primary Landweber transforms for 8-electrode sensor (N=100, L=0.01)

Figure 8.2.2 is a detailed view of the **transform** for the electrode-pair 1-5 **(opposite electrodes**) and is an enlarged view of the last transform in figure 8.2.1. Note that there are now several **bands** where the sensitivity coefficients **are positive** (red) in the Landweber transform.

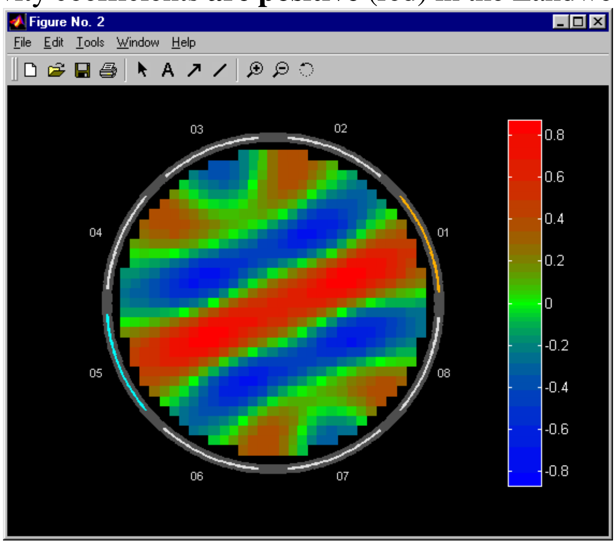


Figure 8.2.2 Landweber transform for electrode-pair 1-4 (N=100, L=0.01)

Figure 8.2.3 displays the result of constructing the image using the Landweber inverse transform with the program **recon.m.** The image is more distinct and compact than that obtained using LBP, but there are now some spurious permittivity artefacts around the edge of the sensor. Use of a lower Landweber factor (eg L = 0.001) reduces these artefacts but also reduces the improvement in the primary image as shown in figure 8.2.4.

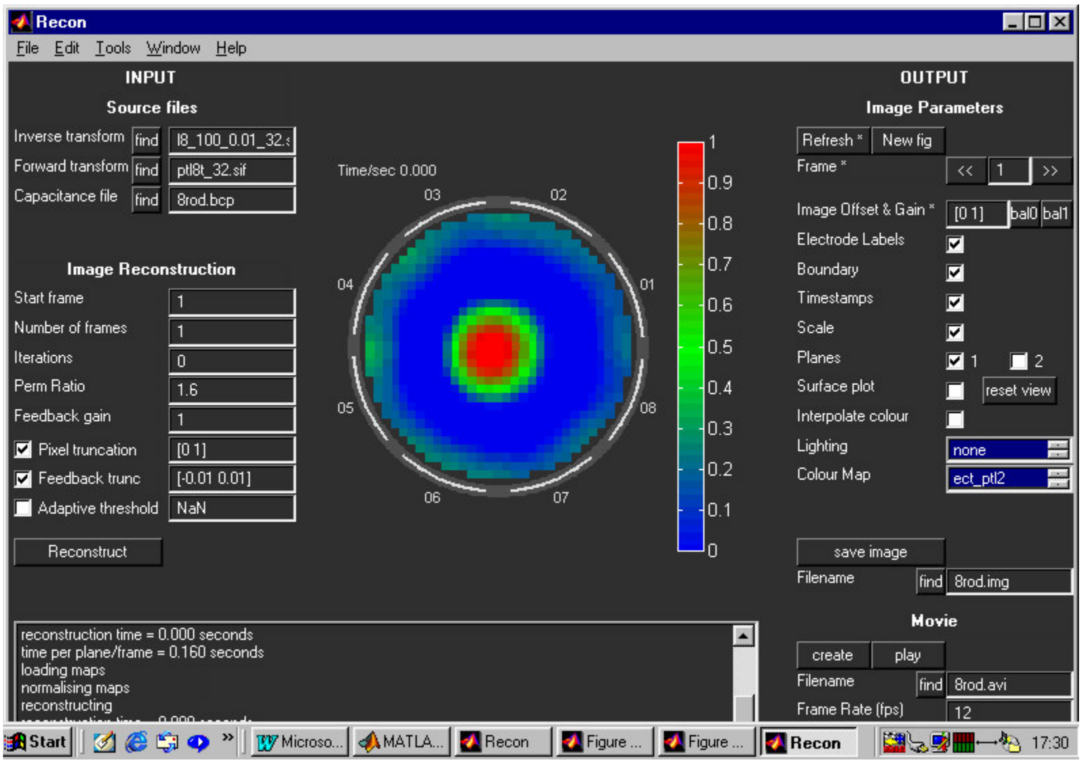


Figure 8.2.3 Reconstructed image using Landweber inverse transform (L = 0.01)

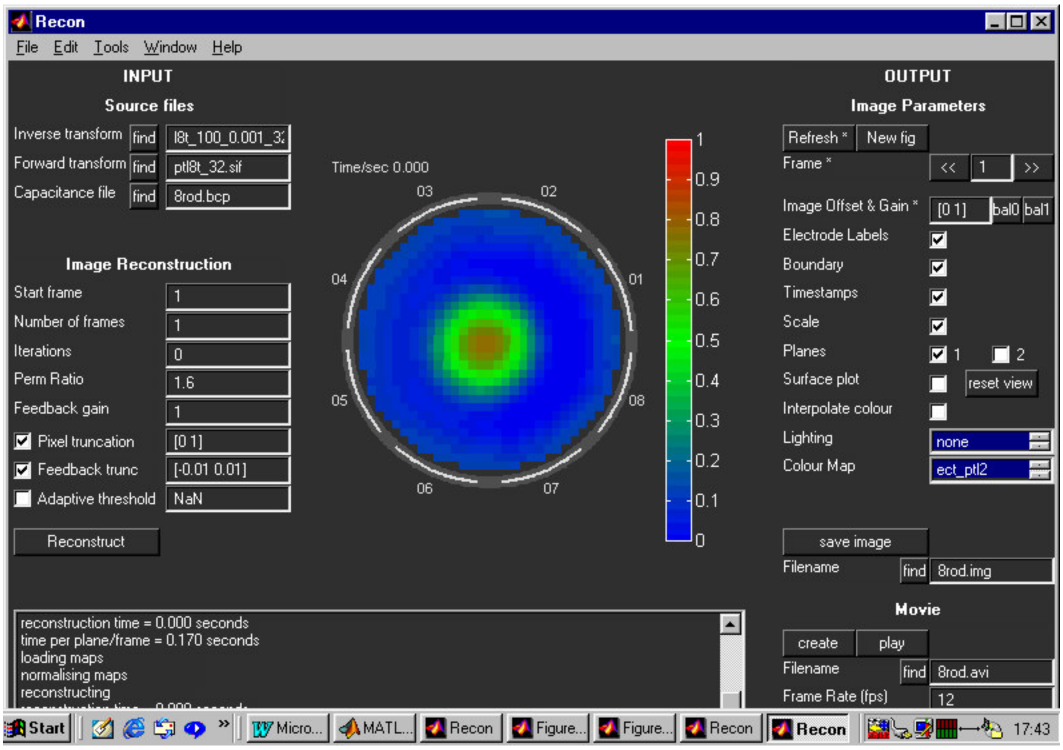


Figure 8.2.4 Reconstructed image using Landweber inverse transform (L = 0.001)

8.3 RESULTS OBTAINED USING TIKHONOV INVERSE TRANSFORM

Figure 8.3.1 shows the transpose of the **primary Tikhonov inverse transforms** (T = 0.01) for each unique electrode-pair for an 8-electrode sensor with external electrodes and a wall thickness of 10% of the tube diameter. The transforms are again plotted as equivalent sensitivity matrices with the electrodes are shown in their conventional positions

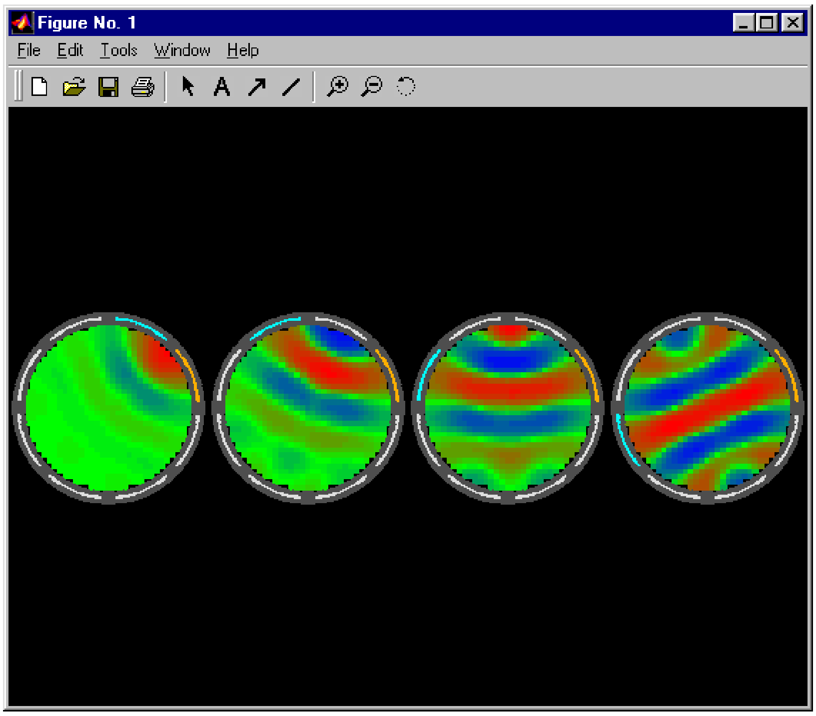


Figure 8.3.1 Primary Tikhonov transforms for 8-electrode sensor (T=1)

Figure 8.3.2 is a detailed view of the transform for the electrode-pair 1-5 (**opposite electrodes**) and is an enlarged view of the last transform in figure 8.3.1. Note that there are again several **bands** where the sensitivity coefficients **are positive** (red) in the Tikhonov transform.

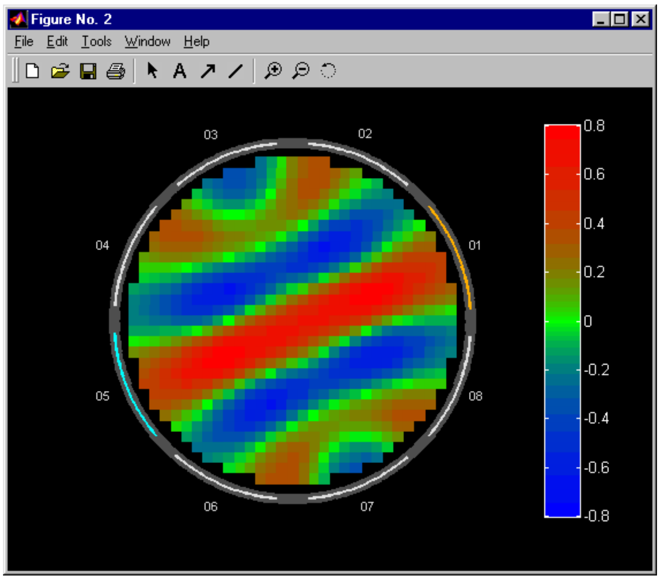


Figure 8.3.2 Tikhonov transform for electrode-pair 1-5 (T = 1)

Figure 8.3.3 displays the result of constructing the image using the Tikhonov transform with the program **recon.m.** The image is again more distinct and compact than that obtained using LBP, but there are again some spurious permittvity artefacts around the edge of the sensor. These can be reduced by increasing the Tikhonov factor to 10 as shown in figure 8.3.4.

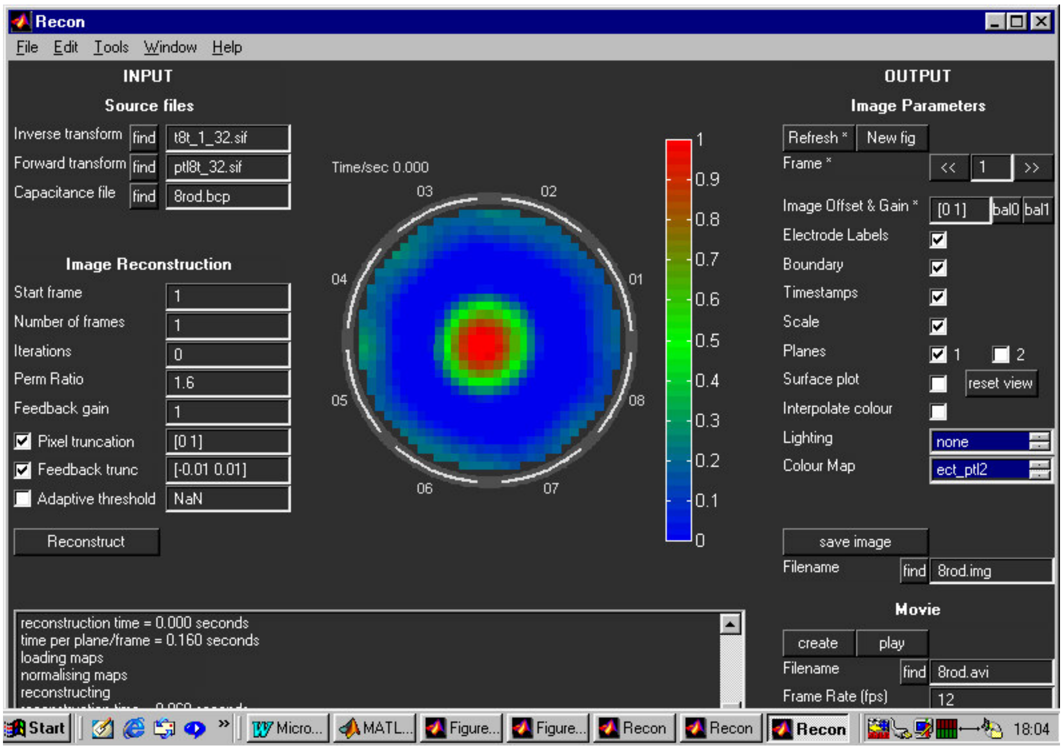


Figure 8.3.3 Reconstructed image using Tikhonov inverse transform (T = 1)

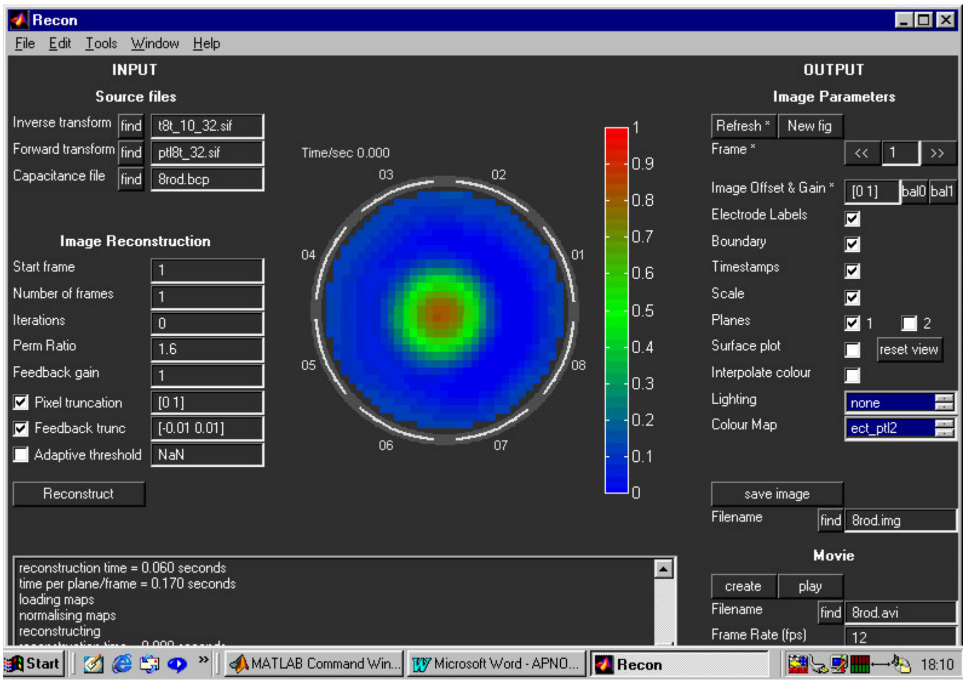


Figure 8.3.4 Reconstructed image using Tikhonov inverse transform (T = 10)

8.4 RESULTS OBTAINED USING ITERATION

Figure 8.4.1 displays the result of constructing the image from the same data using an iterative version of the LBP algorithm, where the sensitivity matrix and its transpose are used for the forward and inverse transforms. The image is now the correct size.

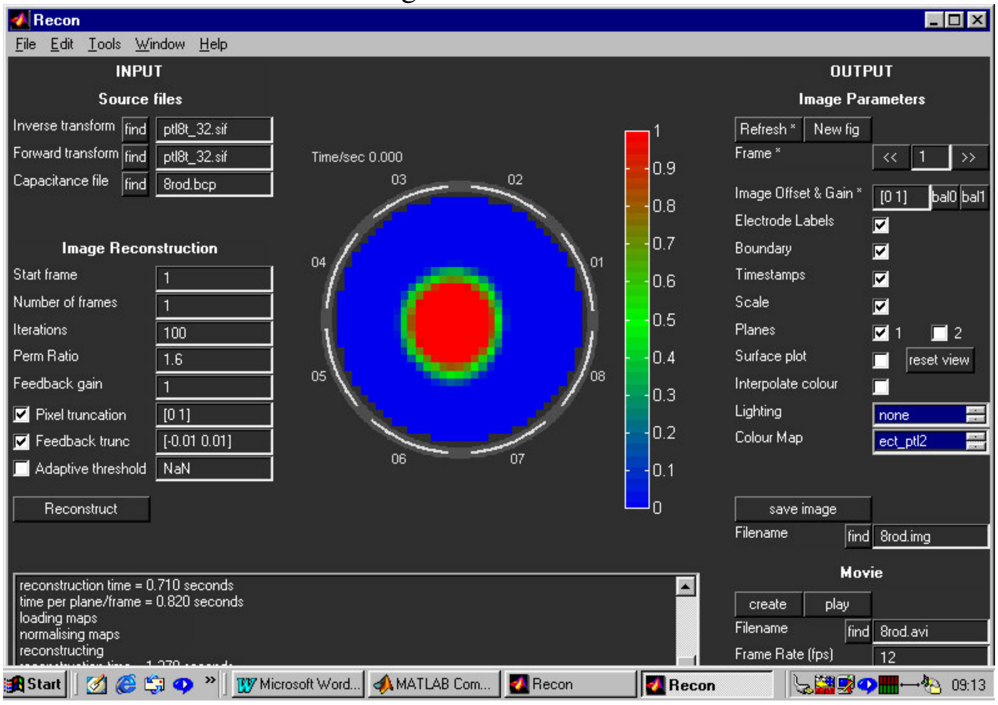


Figure 8.4.1 Reconstructed image using iterative LBP (N=100)

For comparison, figure 8.4.2 below shows the same image obtained using iteration, but this time, using the Tikhonov inverse transform instead of the transpose of the sensitivity matrix and for only 5 iterations.

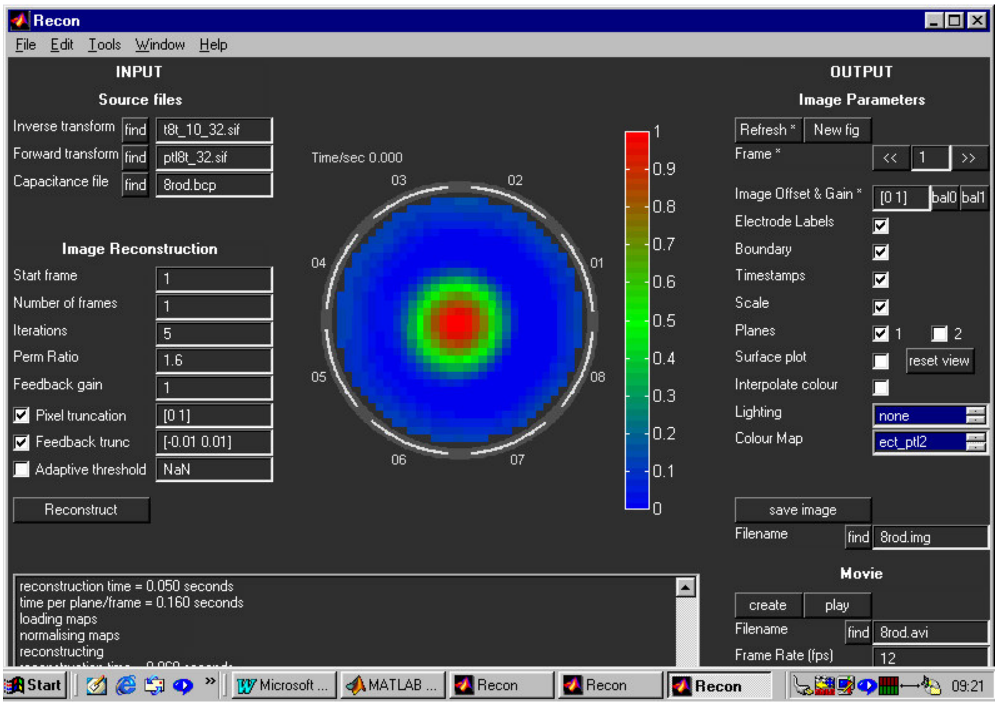


Figure 8.4.2 Reconstructed image using iterated Tikhonov transform (T=10, N =5)

9. IMAGE RECONSTRUCTION ACCURACY TESTS

We carried out some tests using a 12-electrode sensor with **interna**l electrodes to determine the accuracy of image reconstruction based on the calculation of voidage. We used sensitivity matrices calculated using the **Makemap** software, the **iterative algorithm** and a range of **permittivity models** (see PTL Application Note AN2, available for downloading on the PTL web site *www.tomography.com*). The detailed results are shown in this section but can be summarised as follows:

For the case of a dielectric rod in air (ie **a higher permittivity object** inside a **lower permittivity space**) the **series permittivity model** gives the most accurate values of voidage when the permittivity ratio of the materials used for sensor calibration is used.

For the case of a cylindrical void containing air inside a sensor filled with glass beads (ie a **lower permittivity object** inside a **higher permittivity space**) the **Maxwell permittivity model** gives the most accurate values of voidage when the permittivity ratio of the materials used for sensor calibration is used. Details of the individual tests are given in sections 9.1 and 9.2.

9.1 PLASTIC ROD CONTAINING GLASS BEADS IN AIR

The following results were obtained by calibrating a 12-electrode sensor, of internal diameter 128mm, with air and glass beads (effective K = 3). We then introduced a thin plastic tube of external diameter 40mm filled with glass beads into the sensor. The true voidage is (40/128)^2 = 9.8%.

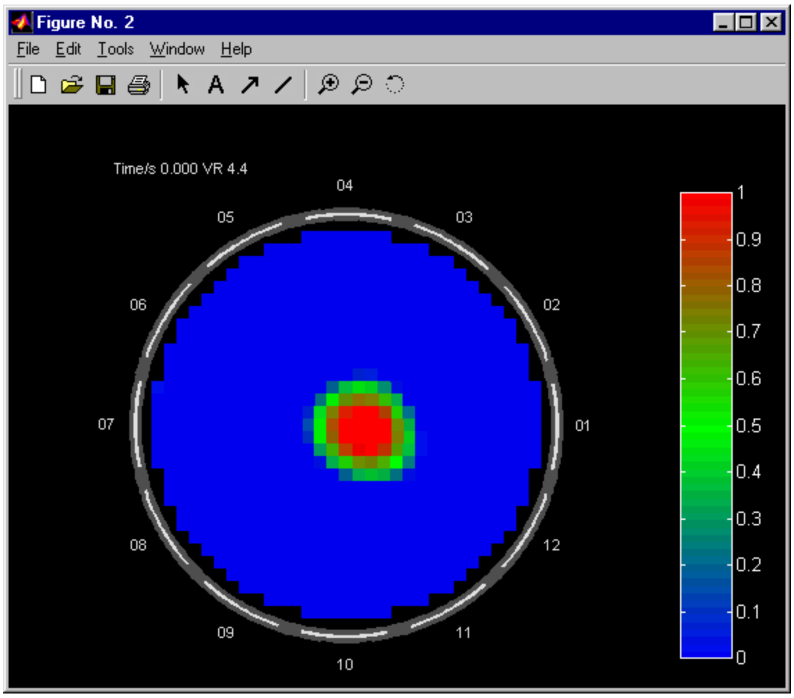


Figure 9.1.1 Parallel model, (K = 1), N=100, VR = 4.4%

Figure 9.1.1 shows the image obtained using the **parallel permittivity model** for 100 iterations and a permittivity ratio of 1. The voidage (VR) obtained is 4.8%, which is less than the known value (9.8%).

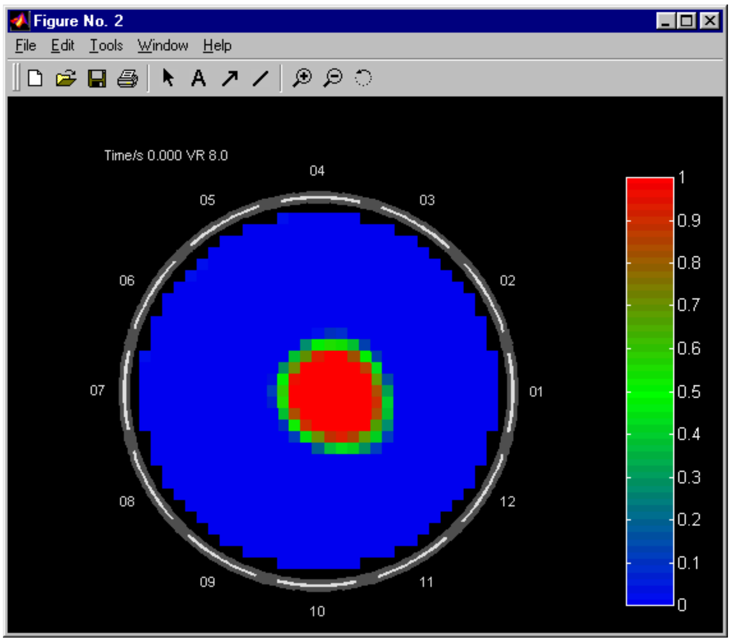


Figure 9.1.2 Series model, K = 2, N=100, VR = 8.0%

Figure 9.1.2 shows the same data but this time using the **series model** for a permittivity ratio of 2. The voidage (VR) has increased but is still less than the known value.

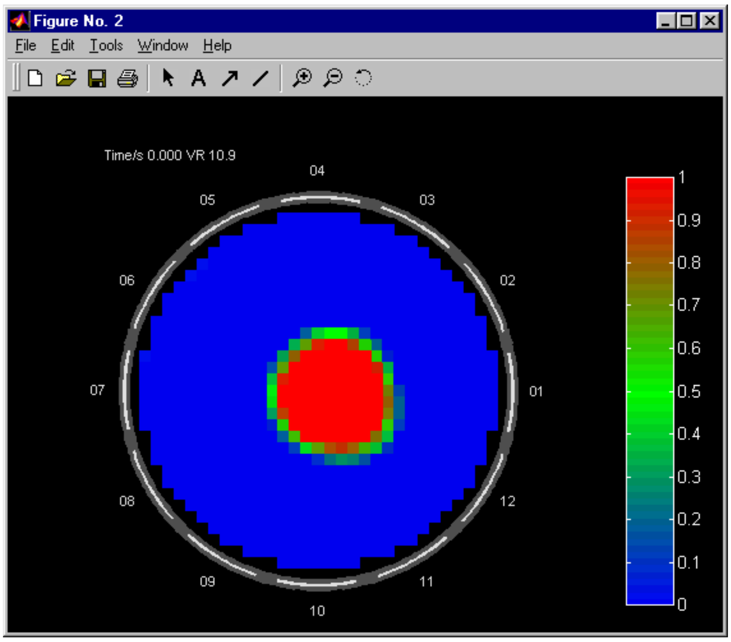


Figure 9.1.3 Series model, K = 3, N=100, VR = 10.9%

Figure 9.1.3 shows the same data but this time for the correct permittivity ratio of 3. The calculated voidage (VR) is 10.9% and now exceeds the known value. If the Maxwell model is used to construct the image, the VR is found to be 7.05% using the same data as used in figure 9.1.3.

9.1.2 CYLINDRICAL AIR VOID INSIDE GLASS BEADS

For a second set of tests, the sensor was filled with glass beads and the same plastic tube, this time empty, was inserted inside the sensor. The known voidage is 90.2%.

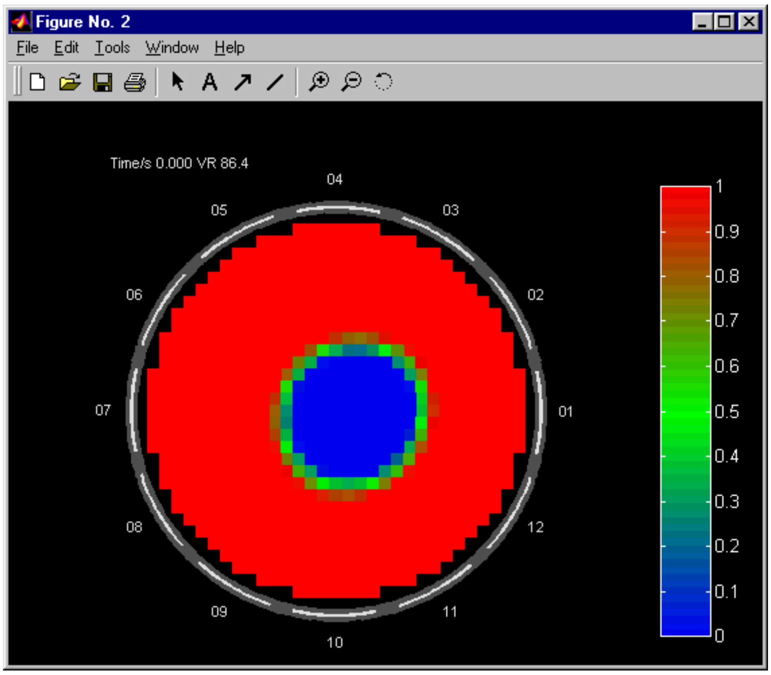


Figure 9.2.1 Parallel model, (K = 1), N=100, VR = 86.4%

Figure 11.2.1 shows the image constructed using the **parallel permittivity model** (K = 1). The calculated voidage (86.4%) is lower than the known voidage (90.2%).

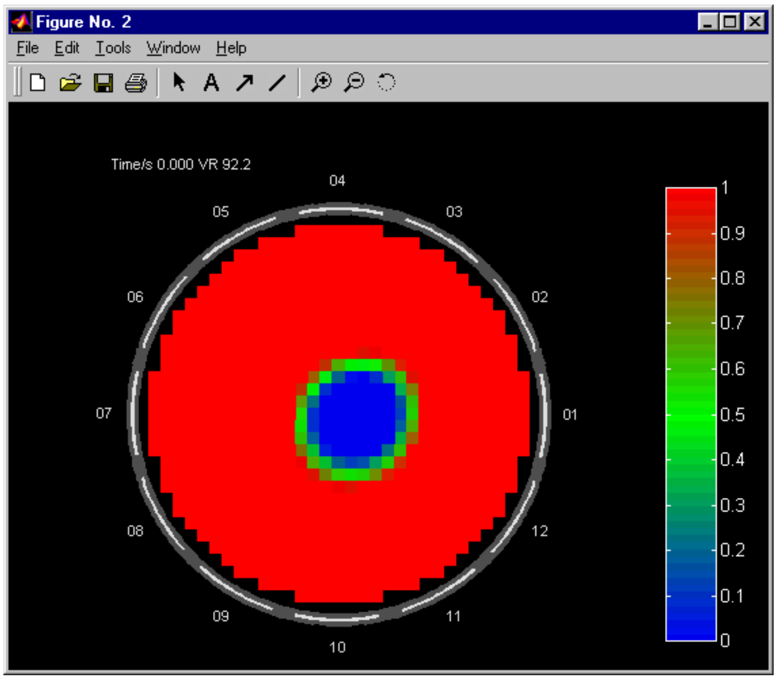


Figure 9.2.2 Series model, K = 2, N=100, VR = 92.2%

Figure 9.2.2 shows the same data calculated using the **series model** and a permittivity ratio of 2. The calculated voidage now exceeds the known voidage.

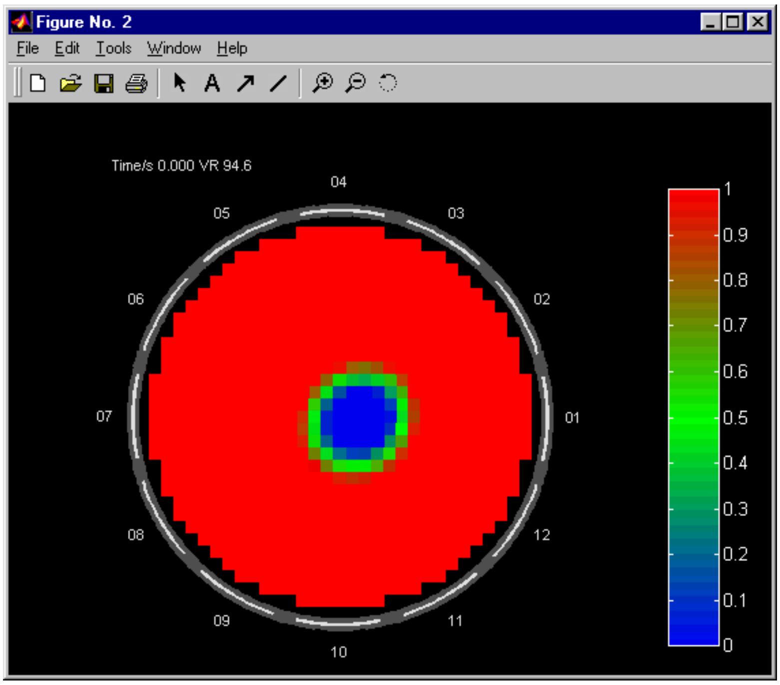


Figure 9.2.3 Series model, K = 3, N=100, VR = 94.6%

Figure 9.2.3 shows the results recalculated for the correct permittivtiy ratio of 3. The results show an even higher voidage (94.6%).

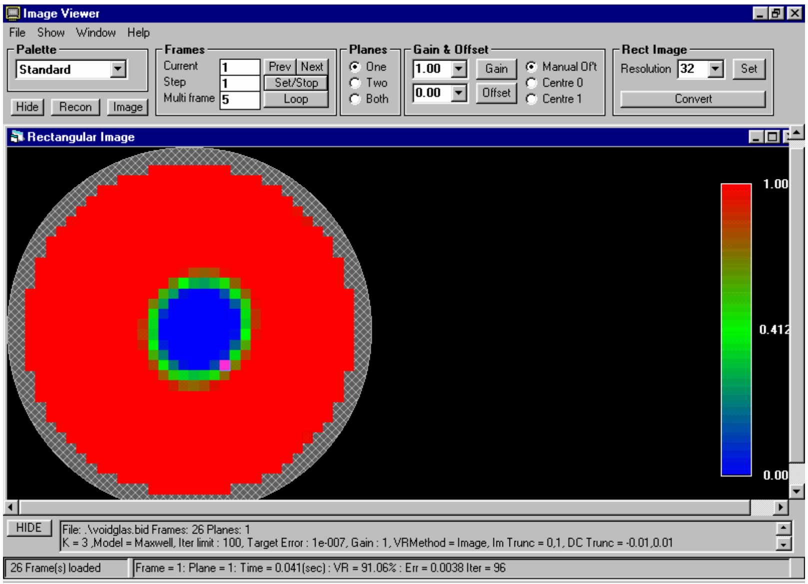


Figure 9.2.4 Maxwell model, K = 3, N=100, VR = 91.1%

For comparison, figure 9.2.4 shows the same data recalculated using the Maxwell permittivity model. The results are now close to the true voidage.

APPENDIX 13

FILE CONVERSION SOFTWARE FOR CAPTURED ECT DATA

BCPCONVERT

The **BCPCONVERT** software converts a **measured normalised capacitance data file** into a range of alternative files, including **image** and **absolute capacitance** files. A **reference file** can also be used to compensate for the effects of any residual offsets in the original measured data file. The **BCPCONVERT** software is supplied as either a stand-alone executable program file or as a Matlab .m file.

A13.1 PROGRAM DETAILS

When the **BCPconvert** program is run the following data input window appears:

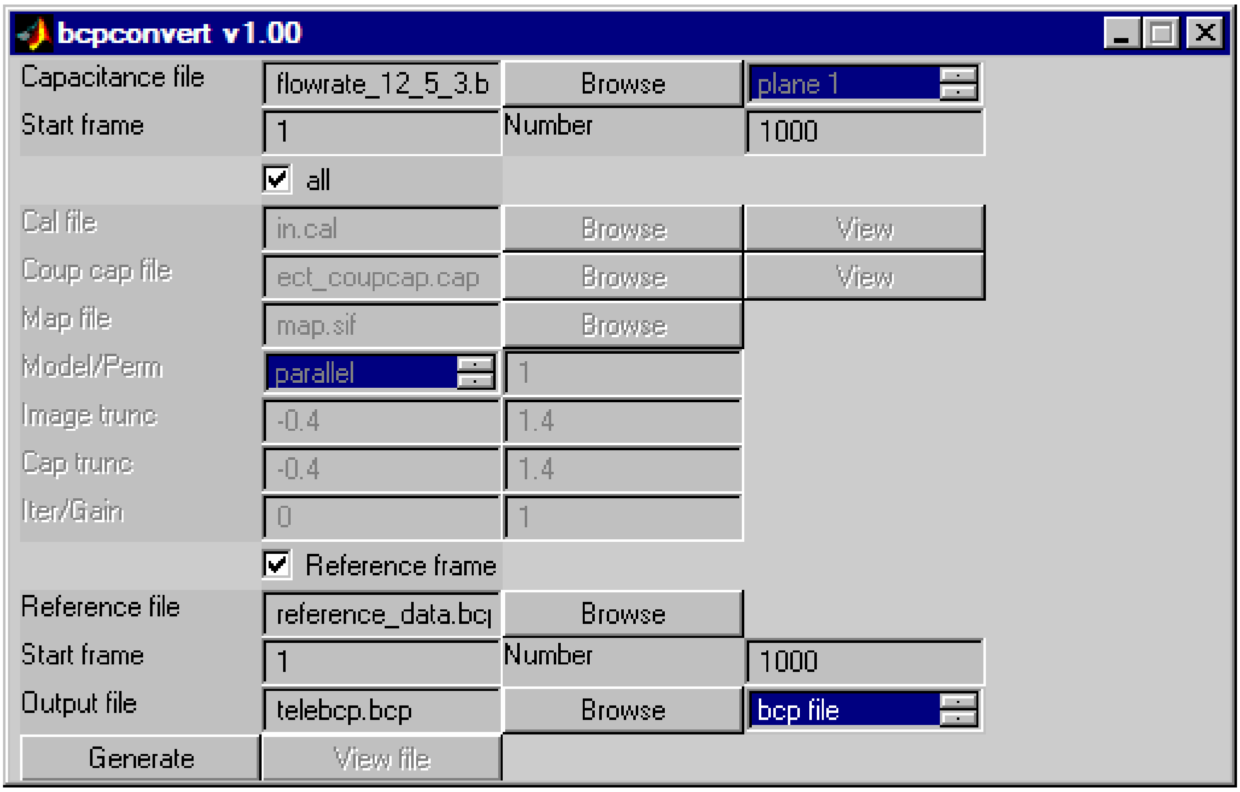


Figure A13.1. The BCPconvert window at start-up

The various data, scroll and tick boxes are used to enter the input and output data. Note that the data to be entered depends on the form of output data required. The data input parameters are as follows:

Capacitance file: The captured **.bcp normalised capacitance data file** to be converted. Select the required file using the adjacent **Browse** button.

Start frame: The number of the **first frame** in the data file to be converted.

Number: The number of **consecutive frames** to be converted,

All: If this box is ticked, **all of the frames** in the **input data file** will be converted.

Calibration file: The **calibration file** used to generate the recorded data (only required to generate absolute capacitance files). Select using the adjacent **Browse** button.

View button: Displays the contents of the **selected calibration file** as **absolute** capacitances in fF.

Coupcap file: The coupling capacitance file for the DAM200E unit used to generate the recorded data (only required to generate **absolute capacitance files**). Select using the adjacent **Browse** button. This file can be **generated or copied** to the **control PC** from the **embedded PC** inside the **DAM200E** unit using the **ECT Toolkit** software.

View button: Displays the contents of the selected **calibration file** as **absolute** capacitances in fF.

Map file: The **sensitivity matrix file** used to convert the .bcp data into **image data.** Only required to produce image data files. Select the required file using the adjacent **Browse** button.

Model/Perm: The **capacitance/permittivity model** and **permittivity ratio** used to generate the output image data. Only required to produce **image data files.** Select the required model (**parallel/series/Maxwell**) using the **scroll box**.

Image trunc: The **upper and lower normalised permittivity pixel limit**s used to generate the **image** file.

Cap trunc.: The **upper and lower normalised capacitance limits** used to generate the **image file**.

Iter/Gain: The **number of iterations** and the **feedback gain** to be used for **iterative image reconstruction**. For LBP reconstruction, set Iter = 0.

Reference frame: If this box is ticked, capacitance data from a **reference frame** (see below) is **subtracted from all of the frames in the data file before they are converted.** This facility is useful for **removing residual offset errors** from captured data files.

Reference file: The file name of the file containing the reference data. Select using the **Browse** button. The **data used for the reference frame** is the **average** of all of the frames selected below:

Start frame: The number of the **first frame** to be used to calculate the reference frame data.

Number: The **number of consecutive frames** to be averaged to produce the **reference frame** data.

Output file: The name of the file to hold the **converted output data**. Select the **file type** (norm cap, absloute cap, image, bcp file) using the adjacent scroll box.

Generate button: This button generates the output file.

View fie button: Views the converted output data file.