DEVELOPMENT AND IMPLEMENTATION OF COMPUTER TECHNOLOGIES FOR IMPROVING DESIGN AND PRODUCTION PROCESSES

Mira, F.J. Clausell, J.J. Nogera, J.F.

ALICER. Centre for Innovation and Technology in Industrial Ceramic Design.

ABSTRACT

The purpose of this communication is to present the project "Development and implementation of computer technologies for improving design and production processes", which has been conducted at ALICER in the course of the last year, and in which it has been attempted to implement a computer system to help the designer who performs ink separations in his daily work. The system consists of a company colour database, software that performs the separations, and information flows that link the colour database to the ink separation software.

1. INTRODUCTION

The screen printing ink separation process differs at present from that performed in graphic arts where the inks are known (Cyan, Magenta, Yellow and Black), since each model will need a specific set of inks. The selection of these inks usually depends on the criterion of the technician who performs the separation, and who chooses the colour of these inks to match the original as closely as possible. On the other hand, the photolithos corresponding to each ink are obtained by methodologies that differ from technician to technician. After the photolithos have been filmed, the development department tries to match the result offered by the design department. This approach leads to the creation and management of a very large number of colours. This entire process (separation and filming of photolithos for screen printing) is carried out by a group of companies (colour producers, third-fire ceramics manufacturers) equipped with CAD systems, which supply the 350 companies that make up this sector.

The ink separation process involves performance of a series of repetitive tasks by the technician. These usually consist of:

- Selecting some of the channels that the software offers automatically (Cyan, Magenta, Yellow, Black, Red, Green, Blue, L, a, b) and subsequently modifying these if he considers this advisable.
- A second way is to obtain one or more channels by means of the colour range tool. This tool allows selecting the predominant colours in the image.
- Jointly visualizing the created channels to adjust each of these. This adjustment performs three changes: Colour, opacity and density of the photolitho. These adjustments are made separately for each channel. In particular, the colour change is not interactive, i.e., the choice of a colour does not automatically entail the simulation, but the colour dialogue box must be closed for the changes to be perceivable.

After these adjustments, the result may not be as expected, which means that the technician will go back and separate the unsatisfactory channel again.

2. EVALUATION OF THE PRESENT SEPARATION SYSTEM

We can analyze the weak points of the separation system set out above.

- When the technician who performs the separation breaks down the photolithos, he assign the most appropriate colour to each of these, so that the whole set together corresponds to the original image it is sought to reproduce. The selection of these colours is made on the whole RGB space and each separation is therefore performed with a different set of inks.
- The laboratory technician in charge of making the prototype proposed by the separating technician will first have to look and see if the proposed colours exist in the company store. If this is not the case, he will need to formulate new colours that will enlarge the store.

The separation process is a trial and error process, in which the technician's experience is fundamental, so that separation quality will depend on the profile of the technician. Even with years of experience, separation is a tedious and often slow process.

3. DESCRIPTION OF THE PROPOSED SOLUTION

The technician responsible for the separation (design) does not take into account the store of inks that may be available at that moment, so that the inks with which he performs the separation usually need to be formulated again by the development department. The necessary stockpiling of these inks to assure stable shades during production entails having a large store of inks, not so much because of the inks involved in the production of a given product, but because of keeping past models or providing for new series.

In order to avoid unnecessary ink stockpiling, the creation is proposed of a database of the colorimetric components of the colours already made by the company, so that the development department can establish the already existing colours in the company colour palette.

This puts in place a communication flow that uses the existing information networks in the company, since the colour data are transmitted through these networks to each computer of the design department, thus allowing the colours used in design to be the same that the company itself has stored or whose composition it knows perfectly.

The objective of the project is to obtain a system that enables performing automatic ink separations and uses the inks that each company has already defined previously.

This system affords the following advantages:

- Reduction of the number of colours existing in the company with the ensuing reduction in costs.
- Reduction of time devoted to separation.
- Providing a standard methodology for separation, reducing the decisions of the technician who performs the separation.
- Important savings in materials and storage space.

The definitive application has been developed in Java. This language allows the same application to be used in different operating systems, provided they have the free distribution virtual Java machine.

The system developed consists of two key elements:

- A company colour server where colorimetric data are kept in a database.
- Ink separation software that will act as a client and use the information flows assured by the company's telematic networks to communicate with the colour server and thus obtain the information needed to enable performing a separation consistent with the colours used by the company.

The following image schematically illustrates the system:

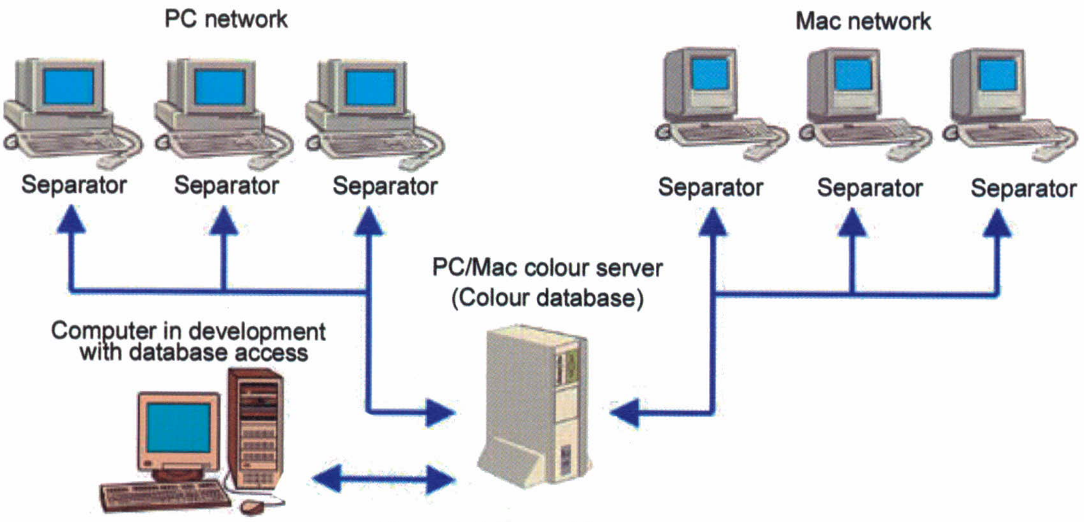

Figure 1. Scheme of the system.

3.1. COLOUR SERVER MODULE

The main function of this module is to provide the different separating programmes with the colours of the store that the company has digitized. This resides in a single computer and communicates with the different separators by means of the TCP/IP protocol.

This protocol enables establishing direct socket connections between the different separating programmes and colour servers. This communication is established by creating a socket (connection) directly from the separating programme with the colour server, keeping it "alive" while the data flow is established and all the colours are downloaded in the programme. On the server side, only one by one single communications will be accepted, keeping a connection queue in case several requests of colours should be made simultaneously. This connection queue will dispatch the colours to the separating programmes in strict order of connection.

As colour information does not take up an excessively large memory space, the simultaneous accesses to the server are made in a time that will be negligible to the user, so that he will usually hardly notice any delay in the start of the application.

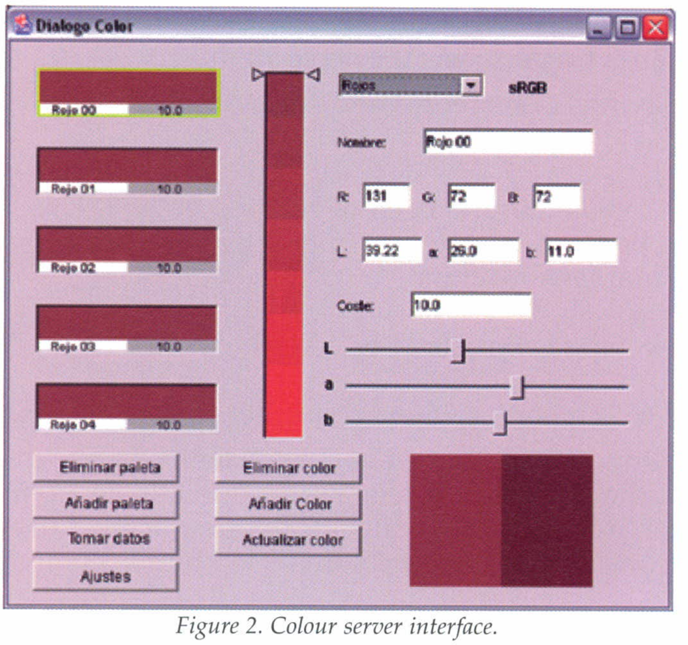

Figure 2. Colour server interface.

Figure 2 shows the dialogue window that appears when we activate the colour database. The following sets out some characteristics of this module:

- Each colour is a register that contains its name, Lab coordinates and a field cost. With this last field, out of two similar colours, the separating programme can choose the less expensive one.
- The separations are made in the RGB space that we define (in Figure 2 the one used is sRGB)
- The possible number of palettes and the number of colours inside each one are only limited by the system memory.

The introduction of the company colours in this database can be made in two ways:

- Manually, using a user interface proper to the separating system, which allows introducing the colours in the database with a name, Lab components and cost field. The programme alerts us when we try to introduce a colour with the same name or same Lab components as another already existing colour.
- Automatically, by means of a Photoshop colour palette file created either by Photoshop itself or by the *GretagMacbeth ColorPicker* software included in the software package of the *Eye-one* spectrophotometer used in the project.

Once the colour base has been created, the different separators will be able to access this.

3.2. INK SEPARATOR MODULE.

The definitive application has been developed in Java. This language enables the same application to be used in different operating systems, provided they have the free distribution Java virtual machine.

Figure 3 shows the first window of the programme that is displayed when the user launches the application:

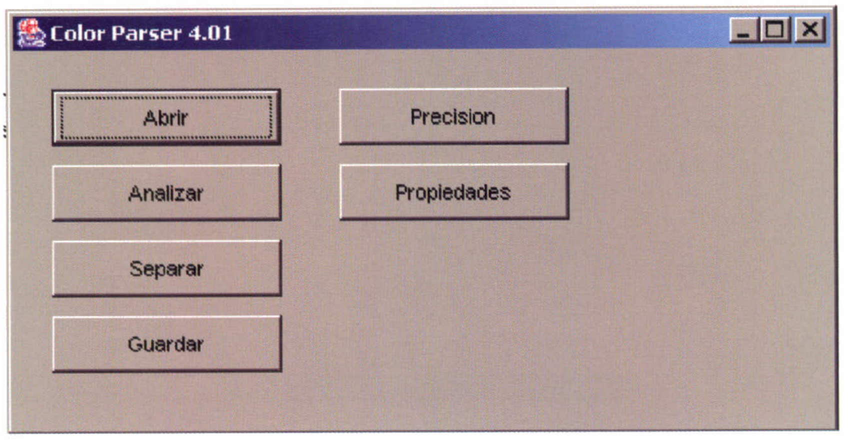

Figure 3. Initial user interface of the ink separator.

Open key: this launches a standard file navigator that allows selecting the desired image. After loading the image in memory, the programme is ready for the next step.

Analyze key: pressing the key immediately displays the window shown in Figure 4, which asks for the number of inks to be used in separating the image. After selecting the number of inks, the dialogue closes and a table is created reflecting all the different colours that appear in the image and the number of pixels of each colour. The search then commences for the colours that will form the enveloping polyhedron by means of a series of search algorithms.

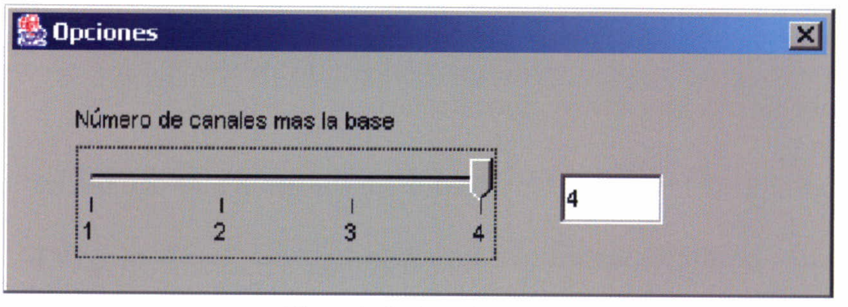

Figure 4. Options dialogue.

The programme calculates the accuracy for each solution found. If this is less than 70%, the search is continued. Finally, after exceeding the accuracy mentioned, the programme displays the following window (Figure 5), which presents the solutions found together with the accuracy.

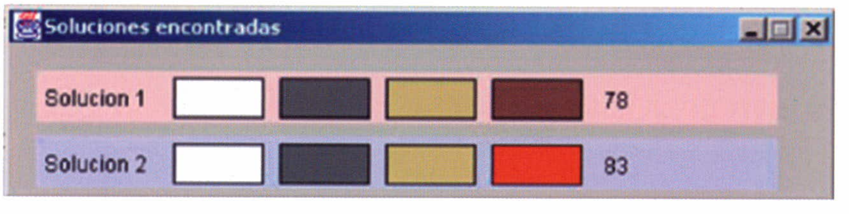

Figure 5. Solutions found dialogue.

The proposed solutions can be modified from a new dialogue box (Figure 6), therefore enabling the technician and designer to set the most appropriate colours in case the solution proposed by the system is unsatisfactory.

Figure 6. Ink selection dialogue.

This dialogue box enables adjusting the RGB components of each ink to be used in separating the image. The accuracy is automatically recalculated so that the technician can adjust the colours without any risk to accuracy.

The proposed colours are those that will allow separation of the image in photolithos. This fact differs from the customary separation procedure where the photolithos are first separated and the colours are subsequently adjusted. This difference presents a great advantage, since the order of the inks can be readily interchanged without impairing accuracy. Figure 6 shows that the system has proposed the white ink as first application or base. However, there would not be any disadvantage to positioning this white ink as last application, since the system would generate the photolithos taking this circumstance into account.

If we wish to find a colour selected by the technician within the company colour palette, it suffices to press on the colour sample to be changed, and the programme will display a pane with the closest colour in the Lab space within the company colour palette (Figure 7).

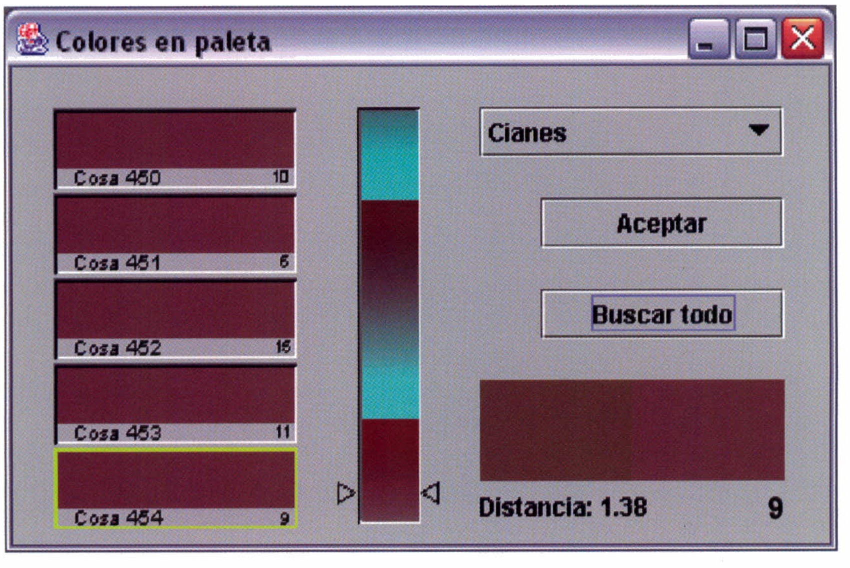

Figure 7. Colours in palette dialogue.

In this dialogue box, we can see the closest colour and its distance in the Lab space. Furthermore, we can look for the closest colour in a particular palette, by selecting it in the palette drop-down box or looking for the colour in all the palettes by pressing the "Search all" key. The cost information field can be used for the most economical selection between two similar inks.

Now that we have selected the relevant colours to perform the separation, the programme is ready to separate the image. After this process, we could select different colours or alter the order of application of the inks to obtain different photolithos. If the separation has been approached as base + 3 inks, it is possible to generate 24 different sets of photolithos (4!).

When the software has finished separating the image, pressing "Save" saves the image in multi-channel Photoshop format, so that the technician that has conducted the separation can continue working with the image and make the relevant adjustments.

The "Accuracy" key displays the accuracy of the proposed solution, i.e., the same as the original would be on screen relative to the multi-channel image. The "Properties" key displays additional information concerning the image, its size, number of channels and number of colours.

4. TECHNICAL DESCRIPTION AND FUNCTIONAL SPECIFICATIONS

Before proceeding with the technical description of the project, we shall briefly set out the variables involved in ink separation in the ceramic sector, since the entire development is based on these variables.

Number of inks. The number of inks that make up the separation for a single-firing process, which is the process normally used for floor tiles, is usually 3 plus the base glaze on which they are printed.

In other processes than the single-firing process, such as third fire, used for the creation of trims, the number of inks may vary considerably from one model to another, and can easily involve up to 8 or more inks. In these cases, separation poses fewer problems as the inks are usually highly differentiated.

Colour of the inks. The colour of these inks is currently chosen by the separating technician to match the original to be reproduced as closely as possible. The development laboratory is in charge of reproducing these colours.

The base glaze colour is chosen so as not to produce any important visual difference with the set of printed inks in order to avoid the appearance of a frame at the edge of the piece, owing to the difficulty of screen printing flush to the edge of the piece.

Order of the inks. The order in which the inks are applied affects the result of the final piece, as each ink is printed on top of the foregoing ones.

The number of inks has been limited to 5 (Base + 4 photolithos), but it would not be difficult to extend the system to a greater number of colours.

The process used by the system can be summarized in the following steps:

- Reading of the image.
- Analysis of the image.
- Search for the colours in all the RGB space that can reproduce the image.
- Substitution of these colours by the closest colour (and least expensive, if so desired) available in the colour base.
- Specification, if necessary, of the order of ink application (performed by the user).
- Calculation of the percentage of each ink for each pixel of the original.
- Generation of photolithos.
- Saving of the resulting document.

The technical weight of the project lies in point 6, relating to the calculation of the densities of each selected ink to obtain the colours of the image.

The developed system uses optical colour mixing to produce a new set of colours. The colour model used is the RGB space, since it allows linear relations. In the case of two colours, it is possible to obtain all those that are in the line that joins them; in the case of three colours, it is possible to obtain those that are in the triangle they form; in the case of four colours, it is possible to obtain those that are in inside the tetrahedron they form, and finally in the case of five colours, it is possible to obtain the colours that are inside the resulting polyhedron. Figure 8 shows the RGB cube, a cloud of points that represent the colours that exist in an image (Cubes), and finally the tetrahedron formed by four inks (spheres). All the colours that lie inside the tetrahedron can be generated by mixtures of the colours that are in the corners of the tetrahedron. The colours that lie outside can be projected on the faces of the tetrahedron, and follow the same process.

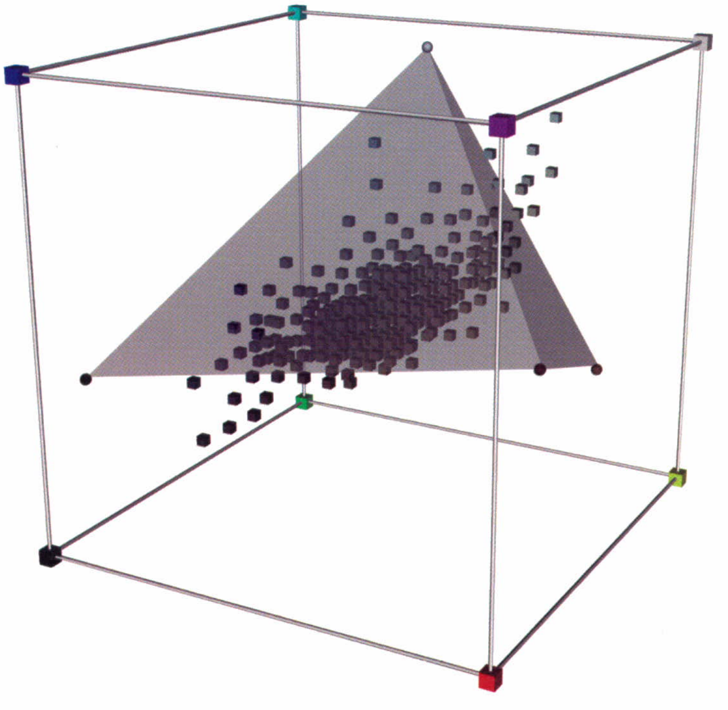

Figure 8. RGB cube

In order to explain the optical separation process technically, we will take the case of a tetrahedron. Given four points in three-dimensional space P1(p1x, p1y, p1z), P2(p2x, p2y, p2z), P3(p3x, p3y, p3z) and P4(p4x, p4y, p4z), which represent the corners of a tetrahedron T and a point P(x, y, z), which represents the control point to be evaluated.

First it is necessary to ascertain whether point P lies inside the tetrahedron, which is done by means of distances to the plane. That is to say, a tetrahedron is made up of four planes. In calculating the distances from the point P to each of the planes, if all the distances have the same sign, the point lies inside the tetrahedron, otherwise the point would have to be projected on the closest place in the tetrahedron.

All the distances inwards "into" the four faces of the tetrahedron have the same sign, i.e., that if a face of the tetrahedron has a positive distance for the points that lie inwards, the other faces will follow this rule of positivity; the same occurs when the distances are negative.

This property or characteristic of the tetrahedron makes it easier to know whether a point lies inside the tetrahedron or not, since in order to know which sign the distances have, we only need to calculate one of its faces, and then apply the result to the rest of the faces.

Now we know the directions in which we have calculated the planes; we shall subsequently need to take them into account when we compute the distances to the point to be evaluated.

Since the tetrahedron will be immobile in three-dimensional space, and it will be the coordinates of the point to be evaluated that will change, the most tiresome calculations only need to be made once as an initialization, so that the calculations to be performed for each point and plane will be simpler (3 multiplications, 3 sums and 1 division).

When a point (colour) lies outside the limits of the tetrahedron, we shall need to project that point on one of the four planes that form the tetrahedron of the four inks to be used.

In order to do this, we will use the previously calculated distances to the tetrahedrons to establish on which plane we must project the point.

Once it has been decided on which plane we are going to project the point, we shall use the calculated normal vector of the plane **(normal N)**, the distance **(with sign d)** and the coordinates of the point to be projected **(P)**.

Knowing these data, we can apply the following formula to establish the coordinates of the projection of P : P´.

$$\mathbf{P' = P - (Nd)}$$

Once we have the point representing the desired colour within the tetrahedron formed by the four inks, we can proceed to apply the separation algorithm in a tetrahedron.

Basically this algorithm attempts to find the way for the colour to follow from its position to plane 1, from plane 1 to edge 1, and from edge 1 to the base colour.

3D space → Plane 1 → Edge 1 → Base Colour

This is unless the colour already lies in a plane, on an edge or in a certain corner of the tetrahedron, in which case the appropriate functions in each particular case would be applied.

As initial considerations we may note that the ink densities vary from 0 (maximum density) to 255 (minimum density), and that to start with, we initialize densities at:

Base : 0 (the base will always have a maximum density).
Ink 1 : 255.
Ink 2 : 255.
Ink 3 : 255.

We shall now briefly deal with special cases:

- The colour lies in a corner.

If the colour lies in one of the corners of the tetrahedron, the solution is immediately at hand: the straight line that joins the corner with the base, giving a maximum point density (0.0) to the channel whose colour corresponds to the corner treated.

- The colour lies on an edge.

Saying that the colour lies on an edge is the same as saying that the colour lies on one of the straight lines that joins any two corners of the tetrahedron.

When an edge is involved that joins any corner with the base, to find the solution, a rule of three is simply applied that yields the quantity of ink used. This calculation must take into account that with a larger number, there is a smaller density, therefore making it necessary to reverse the solution.

Solution = 255 - Solution.

Since a tetrahedron has 6 edges, 3 still remain to be solved, which do not include the base. Although they are not as immediate as the previous ones, their calculation is quite simple. We only need to follow the shortest path to the base, taking into account the order of the inks.

For example, if we consider:
DistT2T1 as the distance between ink 2 and ink 1, DistT1P as the distance between ink 1 and colour, the solution is as follows:

Base: 0
Ink 1: 0 (Solution in corner 1)
Ink 2: 255 - (255 * DistT1P/DistT2T1)

- The colour lies in one of the planes.

When the colour lies in one of the planes (4 possibilities) the solution involves projecting the point to one of the edges pertaining to the plane, and from there to one of the corners.

For example, if point P is located in Plane 1 formed by the base, ink 1 and ink 2:

Given two straight lines: RBsT1 (Base ≥Ink1) and RT2P (Ink2≥Colour), we find Pr, which is the intersection of these straight lines. Next, given the distances DPrT2 and DPrP (distance between Pr and P), gives:

Ink 2: 255 - (255 * DPrT2/DPrP)

And we continue with the solution on **edge 1** and point Pr.

- The colour lies inside the tetrahedron (None of the previous cases).

If the colour lies inside the tetrahedron, the steps are as follows:

- Finding the projection of the colour in plane 1.
- Solving the densities from plane 1 to edge 1.
- Solving the densities from edge 1 to the base.

Given plane1 and straight line RT3P(Ink3 ≥P), the projection of the straight line lies in the plane, yielding projection point Pr. Next, given the distances DPrT3 and DPrP, we calculate the density of Ink 3.

Ink 3: 255 - (255 * DPrT3/DPrP)

Now the solution of Pr in plane 1 is calculated, yielding Pr2. Point Pr2 is used to calculate the solution on edge 1.

To transfer the calculations for solving the densities using a polyhedron of more than four corners requires dividing the polyhedron into tetrahedrons, ascertaining in which of these the point to be solved lies, and using the same routines described previously to find the solution.

5. NEW TECHNOLOGICAL FEATURES OF THE SYSTEM

The following new technological features may be noted:

- Automation of the ink separation process applied to the ceramic sector, which allows obtaining more reliable photolithos or engraving files in a shorter time.
- Use of a limited palette in each company within which the separations are made. This reduction of the palette enables significantly reducing store space. On the other hand, the time devoted to formulating the colours proposed by the technician who performs the separation will be notably reduced, since new ink formulations will only be required in the case of significantly different models from those being manufactured.
- The customary separation procedure first makes the photolithos and then adjusts the colours. In our case the colours that are capable of reproducing the image are chosen first, after which the photolithos are generated, taking into account the order of the applications. This methodological change allows altering the order of the applications and quickly obtaining new photolithos.
- Using the multiplatform feature of the Java programming language to enable performing installations in different operating systems without needing to carry out different developments.
- Centralizing company colour palette data in a single piece of equipment, using the company's telematic networks, thus avoiding having various different files (all the modifications are accessible from the separation programmes).

6. CONCLUSIONS

During the development and implementation of the system at the companies, communication with these companies has been and remains very important, with a view to improving the system. This has led to a list of suggested improvements proposed by the companies and by ALICER.

- Increasing the number of inks to separate the image. Some companies sometimes need to separate an image into more than four inks and a base. One of the objectives of the project is to have the algorithms work with up to nine inks and a base.
- Enable selecting colours of the original image by clicking on the desired region of the image.
- Previsualizing the densities of greys before entirely separating the image, to obtain a preliminary idea of how the channels will be left before applying the separation algorithms.
- Improving system efficiency by expediting the separation algorithms, paying particular attention to the version of the system for MacOs 9, which generates a series of memory problems on being unable to use the latest version of the programming language (Java) for its development.
- Modification of the search algorithms, taking into account the weights of image colours (the times they are repeated), generating a centre of gravity that will allow finding more relevant solutions.
- Including the characteristic of ink opacity, which will allow greater control of the possible interactions between channels.
- Adding a new parameter that enables controlling channel superimposition, if necessary sacrificing the accuracy of the solution on screen, in a controlled form.

Summing up, a system is involved that is fully functional and which, thanks to the collaboration of companies from the sector, is capable of becoming a computer system able to perform any separation in a fully automatic way, without any subsequent modifications of the proposed solutions.

7.ACKNOWLEDGEMENTS

This project has been partially supported by IMPIVA (Instituto de la Pequeña y Mediana Indusria de Valencia) under the program "Plan of Competitiveness and Consolidation of the SME" and by MCYT (Ministerio de Ciencia y Tecnología) under the program for fomenting the Technical Investigation.

REFERENCES

[1] Mortenson, Michael E. 1999. Mathematics for Computer Graphics Applications: Industrial Press.

[2] Glassner, Andrew S. 1995. Principles of Digital Image Synthesis: Morgan Kaukmann.

[3] Hunt, R.W.G. 1995. The Reproduction of Colour: Fountain Press.

[4] Giorgianni, Edward J. y Madden, Thomas E. 1997. Digital Color Management, Encoding Solutions: Addison Wesley.

[5] Foley, James D., Van Dam, Andries, Feiner, Steven K. and Hughes, John F. 1993. Computer Graphics, Principles and Practice. Second Edition: Addison Wesley.

[6] Jaworski, Jamie. 1999. Java 1.2 Al Descubierto: Prentice Hall.

[7] Hardi, Vincent J., 2000. Java 2D Api Graphics: Prentice Hall.