

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ciencias y Humanidades

Conversión de imágenes para realizar  
patrones de bordado de cruceta

Guatemala

2004

Conversion de imágenes para realizar  
patrones de bordado de cruceta

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ciencias y Humanidades

Conversión de imágenes para realizar  
patrones de bordado de cruceta

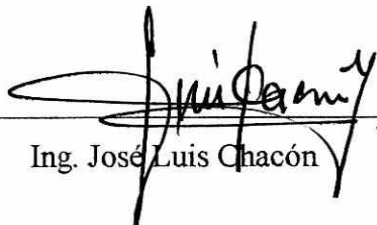
Trabajo profesional presentado para optar al grado académico de  
Licenciada en Ingeniería en Ciencias de la Computación

Guatemala

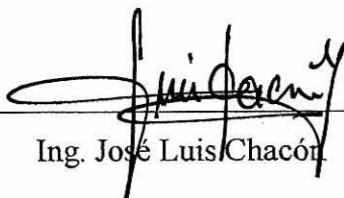
2004

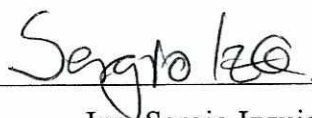
BIBLIOTECA  
DE LA  
UNIVERSIDAD DEL VALLE DE GUATEMALA

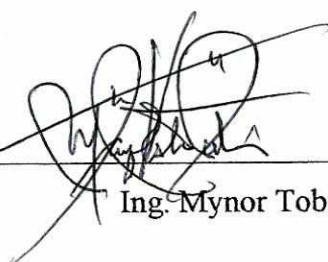
Vo.Bo.:

(f)   
Ing. José Luis Chacón

Tribunal:

(f)   
Ing. José Luis Chacón

(f)   
Ing. Sergio Izquierdo

(f)   
Ing. Mynor Tobías

Guatemala, 4 de Diciembre de 2004

**AGRADEZCO A:**

- DIOS:** Por su amor y las tantas bendiciones que me ha dado en mi vida. Por darme ésta oportunidad de aprender más.
- MIS PADRES:** Por su apoyo en cada momento, su guía y su cariño. Y en especial por su apoyo durante este proyecto. A mi mamá un especial agradecimiento por las muchas horas que pasó bordando los patrones de cruceta que fueron las pruebas. Los quiero mucho.
- MIS HERMANOS:** Por ser tan especiales conmigo, darme su cariño y siempre estar allí. Por ser mis amigos y motivarme a ser mejor. También un agradecimiento a Amy y Andrés.
- MIS ABUELITOS:** Con mucho, mucho cariño.
- A MI ASESOR:** Ing. José Luis Chacón, por darme él empujoncito que necesitaba para empezar este proyecto.
- A MIS PROFESORES:** Por toda su ayuda en mi formación académica.
- A LAS DEMAS PERSONITAS ESPECIALES DE MI VIDA:**  
Por su cariño y por hacerme feliz. En especial un agradecimiento a mis amigos especiales y a todos los pequeñitos de mi vida, que me han dado su cariño incondicional y han dado luz a mi vida.

# CONTENIDO

I. INTRODUCCIÓN	1
II. BORDADOS DE CRUCETA	3
A. Historia del punto de cruz o cruceta	3
B. Patrones de bordado	5
C. Puntadas	5
D. Tela	6
1. Tipos de tejidos	6
E. Hilos	7
III. IMÁGENES DIGITALES	9
A. Píxel	9
B. Imágenes tipo vector	9
C. Imágenes tipo ráster	9
1. Bitmaps	9
2. Profundidad del bit	10
3. Espacios de colores	11
IV. GENERACIÓN DE PATRONES A PARTIR DE IMÁGENES DIGITALES	13
A. Entrada	13
B. Salida	13
C. Proceso de creación de patrones	14
V. REDUCCIÓN DE TAMAÑO DE IMÁGENES	17
A. Cortar la imagen	17
B. Salto de píxeles	17
C. Promediar píxeles	18
1. Promediar píxeles	18
2. Promediar píxeles utilizando matriz ponderada o de pesos	19
D. Algoritmo del cociente para escalar la imagen	20
E. Análisis de los algoritmos en la aplicación	20
VI. REDUCCIÓN DE COLORES	22
A. Selección del mejor color	23
1. Utilizar paleta de colores creada	23
2. Selección de colores utilizando toda la gama de colores DMC	23
B. Dithering	24

1. Floyd-Steinberg dithering	24
C. Análisis de algoritmos	26
VII. CONTORNOS	28
A. Operador de Sobel	28
B. Operador de Laplace	30
C. Análisis de algoritmos	31
VIII. COMPARACIÓN DE IMÁGENES	32
A. Error del promedio de los cuadrados (MSE)	32
B. Normal de los promedios de los cuadrados (NMSE)	33
C. Percepción visual	34
IX. INTERFASE CON MAQUINAS BORDADORAS	37
A. Formato XML	37
1. Documentos XML.	38
2. DTD del patrón	42
B. Lenguaje XSL	42
1. Componentes del XSL	43
2. Modelos de uso del XSL	44
3. Procesamiento de hojas de estilo	44
C. FORMATO .DST	45
1. Máquinas de bordar marca Tajima	45
2. Gramática del formato DST	45
3. XSL para transformar al formato .DST	47
X. CONCLUSIONES Y RECOMENDACIONES	49
XI. APÉNDICES	50
A. Glosario de términos	50
B. Encuesta	54
C. Selección de la muestra	56
D. Comparación con otros programas	58
E. Ejemplos	58
F. Manual del usuario	62
XII. BIBLIOGRAFÍA	66

## TABLA DE ILUSTRACIONES

2.1 Patrón a colores	5
2.2 Patrón a blanco y negro	5
2.3 Punto de Cruz	6
4.1 Patrón de símbolos generado por la aplicación	14
4.2 Proceso de creación del patrón	15
5.1 Imagen reducida con salto de píxeles	17
5.2 Ejemplo del promedio de píxeles	18
5.3 Comparación de tamaño utilizando diferentes algoritmos	21
6.1 Comparación de dithering y cambio de color simple	25
6.2 Comparación de cambio de colores. Con colores y tonos grises	27
7.1 Ejemplos utilizando operadores de Sobel	29
7.2 Ejemplos utilizando operadores de Laplace	30
8.1 Gráfica de comparación del NMSE de imágenes con distinta cantidad de colores	33
8.2 Gráfica con NMSR para imágenes utilizando diferentes algoritmos	34
8.3 Gráfica de comparación de imágenes más cercanas a la original	35
8.4 Gráfica de comparación de imágenes que se verían mejor bordadas	35
8.5 Gráfica de comparación de imágenes más fáciles de bordar	35
9.1 Transformada de XML en 2 partes: Transformada XSL y Formateo XSL	45
9.2 Bytes representantes de cada registro de puntada del formato .dst	46
9.3 Manera de bordar cada cruz con puntadas simples	47
11.1 Comparación con patrones generados por otros programas	58
11.2 Ejemplo de Patrón generado (Country)	59
11.3 Ejemplo de Patrón generado (Ave del Paraíso)	60
11.4 Ejemplo de Patrón ya bordado (Country)	61
11.5 Ejemplo de Patrón ya bordado (Ave del Paraíso)	61

## LISTA DE TABLAS

2.1 Hebras dependiendo del count de la tela	8
2.2 Número de colores disponibles por píxel	5
6.1 Difusión de error utilizando Floyd-Steinberg Dithering	25

# RESUMEN

Este trabajo proporciona la base teórica que se utilizó para la creación de una aplicación que convierte imágenes a patrones de bordado de cruceta. Este incluye varios temas; entre estos: reducción de colores, reducción de tamaño y contornos. Aquí se explican los diferentes algoritmos y como se aplicaron en el proyecto.

Además este trabajo explica cómo en el proyecto se generaron patrones en formato XML, y cómo a estos pueden aplicársele transformaciones para generar los formatos que utilizan máquinas bordadoras.

# I. INTRODUCCIÓN

El bordado de crucetas ha sido desde hace muchos años una de las aficiones de muchas mujeres en diferentes países alrededor del mundo. La cruceta es una clase de bordado que consiste en transferir diseños de patrones a la tela. En este tipo de bordado quien lo realiza coloca cruces de los colores que corresponden a los símbolos que se encuentran en el patrón.

La compra de patrones de símbolos resulta bastante cara. Hacerlos a mano es una tarea realmente difícil. Por otro lado, en muchas ocasiones las personas tienen dibujos o imágenes que les gustaría bordar, para los cuales no existe un patrón. En estos casos tener un programa que genere patrones de cruceta a partir de imágenes resulta indispensable.

Estas fueron algunas de las causas que motivaron el desarrollo de esta aplicación. Pero más que eso, fue ver cómo un proyecto así podía integrar muchos conocimientos sobre gráficas que son aplicables en otras áreas de la vida.

Este trabajo tiene como propósito explicar el proceso que se siguió para desarrollar la aplicación que genera patrones de bordado de cruceta a partir de imágenes.

En dicho proceso se tomó en cuenta que al bordar cruceta no se tiene toda la variedad de hilos como los posibles colores de píxeles que pueden existir en una imagen. Por eso se necesitó comparar los colores de la imagen original ante los colores de hilo disponibles, y así poder utilizar en el patrón solamente los colores de hilo que se tienen. La reducción de colores fue una de las áreas importantes de este trabajo.

Si al bordar por cada píxel se realizara una cruceta, una imagen podría requerir una gran cantidad de crucetas, y por lo tanto una gran cantidad de tiempo, de tela y de hilos; por esto se necesita de alguna manera reducir el número de píxeles originales a una menor cantidad. Por eso la reducción de tamaño fue otra de las áreas en las que se enfocó este trabajo.

Aunque se inició el estudio de algoritmos para contornos, este proyecto no resolvió la construcción de bordes que se adapten a los bordados, y que resalten solamente las partes más importantes de los mismos.

En este trabajo se realizó una comparación entre los diferentes algoritmos de manipulación de imágenes, sus ventajas y desventajas.

Otra de las partes importantes de esta aplicación es que la información de los patrones generados se almacenó en un formato XML, que es uno de los estándares para compartir información. Esto tiene

como objetivo que cada empresa pueda adaptar el XML obtenido a sus máquinas bordadoras por medio de transformaciones de este archivo. Se utilizó como ejemplo el formato .dst utilizado por las máquinas bordadoras Tajima.

Al concluir este trabajo se dejaron dos áreas por desarrollar en futuras investigaciones, una de ellas es la generación de contornos que resalten las partes más importantes del bordado, y la otra es optimizar el orden de las puntadas en el XML de las máquinas bordadoras, a fin de reducir el número de cortes de hilo y los saltos.

## II. BORDADOS DE CRUCETA

### A. Historia del punto de cruz o cruceta

El punto de cruz o cruceta es una de las puntadas que más se utilizan en los bordados. Esta puntada se utiliza individualmente para decorar márgenes, y motivos que embellecen tejidos lisos. A su vez, las dos puntadas que constituyen la cruz crean una perfecta simetría que puede ser repetida para construir líneas, curvas, formas, trenzados y diseños.

Actualmente es considerado como un punto de adorno, mas el punto de cruz no siempre ha tenido esa misión. Debido a su sencillez, los primeros puntos de cruz se emplearon para unir pieles de animales con el fin de proporcionar un abrigo y refugio a los hombres primitivos.

A partir de estos orígenes, esta actividad artesanal fue evolucionando para convertirse en una técnica decorativa y ornamental. El ejemplo más primitivo de uso de una cruceta completa es un diseño trabajado en seda, en una pieza encontrada en Egipto, la cual data del año 500 AC. Si bien, se han encontrado pocas piezas de esa época, no quiere decir que la puntada de cruz haya sido poco utilizada. No se sabe bien dónde pudo haber iniciado el bordado de cruceta, algunos historiadores proponen que fue en China durante la Dinastía T'ang y que de allí se expandió hacia India y Egipto (Cruceta y Punto, 2001).

A mediados de la Edad Media (siglos X y XI), el punto de cruz pasó a ser el pasatiempo principal de las mujeres, sobre todo de aquellas que esperaban a sus maridos, prometidos o hermanos mientras éstos se iban de expediciones o a guerras. En estas interminables esperas, las castellanas, por ejemplo, copiaban a punto de cruz los motivos de las alfombras que se traían desde el cercano Oriente, entre cruzada y cruzada. Los bordados se utilizaban para decorar dobladillos y mangas de trajes tanto masculinos como femeninos. En un momento el bordado de cruceta tuvo consigo la función de indicar las riquezas de la familia y su estatus en la comunidad (Tom Pudding Designs, 2004).

Pero también en esta época se utilizaba como un tipo de bordado campesino tradicional en Europa, que se empleaba para decorar distintos tipos de artículos, desde enseres para el hogar hasta prendas de uso cotidiano.

Durante el Renacimiento, esta afición se extendió por toda Europa y pasó a formar parte de la educación femenina. De esta manera aparece el muestrario o sampler, que eran unas piezas de tejido donde las mujeres bordaban grecas, flores, símbolos religiosos, letras, entre otros, para ejercitarse. Dichos muestrarios se transmitirían de generación en generación, quedando como patrimonio familiar y constituyendo una enciclopedia de consulta, sobre la cual sería posible encontrar el motivo mas adecuado

para la realización de una determinada labor. La idea de los muestrarios es que las crucetas pudieran imitarse. Esos muestrarios podían llevar años en terminarse. En un libro de contabilidad en 1502 se encontró la primera referencia de compra de lino para hacer muestrarios (Cruceta y Punto, 2001).

Así el punto de cruz se convirtió un arte popular, una herencia artesanal, en un gran número de países de todo el mundo. En cada uno de los cuales se van desarrollando distintos motivos y colores, según las características de cada país. Por ejemplo, en China, se llegó a utilizar mucho el hilo de color azul marino sobre tejido de gasa en los bordados a punto de cruz.

En el año 1500 comenzaron a circular los primeros diagramas impresos, procedentes de Italia y Alemania, con motivos florales y animales inspirados en Oriente y su heráldica. En el siglo XVII cuando llegaron nuevos colorantes procedentes de América, se empezaron a teñir los hilos con colores llamativos, teniendo el rojo y azul como colores predominantes para bordar sobre fondos blancos y a los negros y marrones para definir contornos. A esa época se le conoció como la revolución del rojo, ya que casi todos los bordados tenían hilos rojos. También en esa época, las mujeres empezaron a aprender a escribir, lo cual hace que empiecen a aparecer letras y números. En el año 1700, los dibujos se hacen más realistas y por fin, en la segunda mitad del siglo XVIII comienzan a aparecer los primeros paisajes (Cruceta y Punto, 2001).

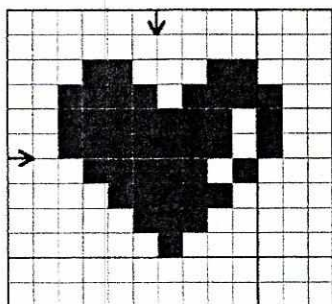
En el siglo XIX, se considera la edad de oro para el punto de cruz. Desde América, donde se introdujo junto con otras tradiciones del viejo continente, hasta Europa, la cruceta alcanza su máximo apogeo. Gracias al desarrollo de la industria textil, que permitió la fabricación de telas de algodón e hilos en muchísimos colores, y también gracias a la difusión de revistas, y en particular a los diagramas coloreados a mano sobre base cuadrículada. Desde 1525 aparecieron los primeros patrones, pero fue en 1804 que el primer vendedor de impresiones en Berlín, Phillipson, introdujo patrones coloreados en cuadrícula. En donde cada cuadro representaba una puntada. Esta técnica ya estaba siendo utilizada en los textiles, más no en los bordados. Así de las tipografías de Berlín sale la mayoría de esquemas cuadrículados para su distribución por toda Europa. El punto de cruz se llega a convertir en parte de la enseñanza en las escuelas primarias (Cruceta y Punto, 2001).

Sin embargo, con la llegada del siglo XX, las modas y las aficiones cambian. Las mujeres prefieren otras puntadas y otros adornos, el punto de cruz apenas aparece tras la II Guerra Mundial. A comienzos de los años 80, vuelve a reaparecer ayudado por la creación de revistas especializadas y el empuje de casas fabricantes (como DOLLFUS-MIEG & Compagnie (DMC) y Coats). Con sus nuevos tipos de telas y colores de hilos, la publicación de colecciones de prestigiosas editoriales, la formación de nuevas diseñadoras, la publicidad y el mercadeo, el gusto por la cruceta vuelve a resurgir. En la actualidad, la variedad de telas y colores de hilos, permite desarrollar un sin límite de bordados (Tom Pudding Designs, 2004).

## B. Patrones de bordado

Existen patrones de cruceta, a colores o blanco y negro. Los de colores muestran cuadros con el color que se va a utilizar, estos ayudan a visualizar de una mejor manera como quedará el bordado final. Los que están a blanco y negro tienen símbolos. Estos símbolos permiten seguir los patrones más fácilmente, en el momento de realizar los bordados, ya que no hay confusión entre dos colores que sean muy parecidos.

Heart in Cross-Stitch - Beginner Pattern  
Copyright 1999 by Cherie Marie Leck



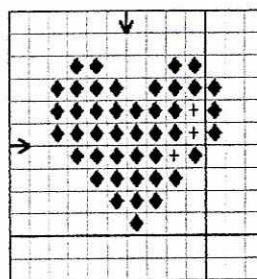
DMC Floss Colors Key:

■ 304 Christmas Red Med

□ White

Fig. 2.1 Patrón a colores

Simple Heart in Cross-Stitch Beginner Pattern  
Copyright 1999 by Cherie Marie Leck



DMC Floss Symbols Key:

◆ 304 Christmas Red Med

⊕ White

Fig. 2.2 Patrón a blanco y negro

Cada símbolo o cuadro de color representa una cruceta sobre un cuadro de la tela sobre la cual se está bordando. El patrón de cruceta tiene una clave, en la que se describe qué número de hilo y qué color está siendo representado por el cuadro de color, o por el símbolo.

## C. Puntadas

Una puntada simple de cruceta es llamada puntada inglesa de cruceta. Ésta tradicionalmente se hace de la esquina inferior izquierda hacia la esquina superior derecha, y luego de la esquina inferior derecha a la superior izquierda.

Las puntadas de cruceta se bordan en el anverso (frente) de la tela. Cada cruceta está formada por dos partes: una puntada diagonal de base inferior y una puntada diagonal de cobertura en la parte superior. Todas las puntadas que sirven de base tienen la misma dirección (ej. Izquierda a derecha), y las que sirven de cobertura tienen dirección contraria (ej. Derecha a izquierda). El hacerlo de esta manera permite que se logre una apariencia más uniforme.

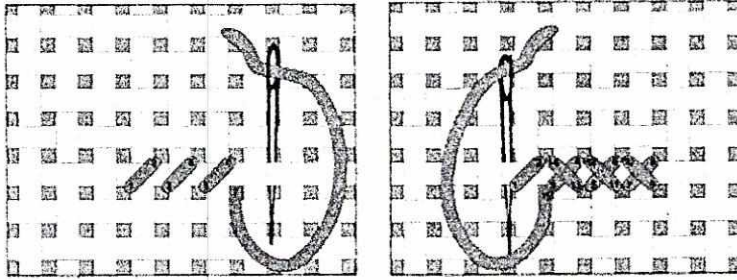


Fig. 2.3 Las puntadas 1-2, 3-4, 5-6 y 7-8 son puntadas de base y las otras son puntadas de cobertura. Nótese que todas las de base llevan una misma dirección.

Cuando se está bordando cruceta se evitan hacer nudos con el hilo. Para fijar el hilo sin necesidad de nudos se deja un segmento libre de hilo en la parte trasera de la tela, el cual se fija al momento de estar realizando las crucetas.

## D. Tela

Las telas más adecuadas son las que ofrecen una base regular para bordar con hilos contados: existen desde la tela Aída, de trama muy regular, hasta el lino, del cual existe una amplia gama de tipos y tamaños para utilizar. También se puede bordar cruceta en telas sin hilos contados (tejanos, terciopelos, punto, etc.), pero resulta más complicado.

Las telas están formadas por hilos que se cruzan entre sí, unos que van horizontales y otros verticales. Se le llama trama a los hilos horizontales y urdimbre a los hilos verticales de la tela. En todos los casos, los cuadros de la tela han de ser perfectos, de modo que la cantidad de hilos de la trama ha de corresponder exactamente con el número de hilos de la urdimbre. De no ser así, el bordado puede quedar desfigurado, ya que las crucetas no serían cuadradas. Cuando se utiliza tela Aída, se habla de agujeros por centímetro, y cuando se utilice lino, se hablará de hilos por centímetro. Es importante hacer notar, que dependiendo del tipo de tela que se utilice, variará el tamaño del bordado. Ya que, si se utiliza alguna tela con más agujeros por centímetro, el bordado quedará más pequeño (Cruceta y Punto, 2001).

1. Tipos de Tejidos: Existe variedad de tejidos sobre los que se puede hacer cruceta, entre éstos están:

- Tela Aída: Es una tela de algodón creada especialmente para el punto de cruz o cruceta. Su trama regular delimita cada uno de los puntos. Es la más fácil de bordar. Existen diferentes tramas: 8, 11, 14, 16 y 18. Estas cifras expresan el número de puntos por pulgada o count. La más utilizada es de 14 counts.
- Tela esterilla: Este término designa numerosas telas de algodón o de lino. Es útil para cualquier tipo de bordado, ya que su trama es bien visible y regular, por lo que permite posicionar cada punto o recortar con precisión los calados (Hardanger). Existen diferentes grosores designados por una cifra que corresponde al

número de hilos de trama por centímetro o también por pulgada (2,54 cms) o count. A diferencia de la tela AIDA, esta solo muestra los hilos, no trae los cuadros, ya hechos (Cruceta y Punto, 2001).

- Lino: Esta tela es más fina, es más resistente y duradera. Debido a su finura, conviene más para las bordadoras muy experimentadas. Suele basarse sobre dos hilos. Existen diferentes grosores designados por una cifra que corresponde al número de hilos de trama por pulgada (a veces por centímetro) o count.

- Asis: Es un tejido puro de lino de trama regular y muy gruesa. Se utilizan cuatro hebras de hilo y cuatro hilos del tejido tanto a lo ancho como a lo largo, por cada puntada. Es ideal para realizar colchas, manteles y cojines (Cruceta y Punto, 2001).

- Adamascado: Es un tejido brillante. Presenta claros y oscuros debido al peculiar trazado de su trama y urdimbre. Resulta ideal para mantelerías. Tiene recuadros de tela Aída para bordar con dos o tres hebras.

- Panamá: Es un tejido muy agradable de trabajar, pues se pueden hacer los medios puntos perfectos. Los cuadros resultan algo grandes.

- Lugana: Es una tela similar al lino y preparada especialmente para el punto de cruz por su trama perfecta. Está disponible en varios colores.

- Hardanger: Es un tejido de algodón cuyo tramado de hilos verticales y horizontales se hallan dispuestos por parejas. Esto da como resultado una red más marcada en la que los agujeros son mucho más perceptibles.

- Cañamazo de bordar: Es una tela especial de algodón muy fino y de trama sencilla. Sirve de soporte para bordar crucetas sobre telas que por su composición no permiten el bordado sobre hilos contados, por ejemplo, sobre telas vaqueras. Para utilizarlo, se hilvana el cañamazo sobre la tela que se desea bordar; después se borda el motivo deseado, y al finalizar, con mucho cuidado se va deshilando la trama del cañamazo (Cruceta y Punto, 2001).

## E. Hilos

Los hilos juegan un papel muy importante en el resultado final de los bordados de cruceta. Existen de varias marcas; entre ellas DMC, Anchor, Coats y Madeira. Cada una de ellas tiene una amplia gama de colores. Los hilos más utilizados son los de algodón tipo Mouliné, que se presentan en madejas de seis cabos, y que antes de su utilización para la cruceta, se deberán separar en hebras individuales y reordenarlas después hasta el número deseado para obtener el grosor requerido: una, dos ó tres hebras (Cruceta y Punto, 2001).

Los tramos de los hilos, al momento de bordar, deberán tener una longitud máxima de 40 centímetros ya que las hebras mas largas tienden a perder su brillo y a deteriorarse durante el procedimiento.

La marca que más se utiliza es la DMC, cuyo lote de hilos es el de tipo Mouliné Art.117 en una gama de más de 460 colores (DMC, 2004). Estos hilos vienen en madejas de seis hebras. Lo normal es utilizar dos y tres hebras para hacer las cruces y una hebra para los pespuntos o *backstitch*.

Cuántas hebras se utilicen para bordar depende mucho del tipo de tela a utilizar (*vid.* Tabla 2.1).

Tipo de tela	Número de hebras a utilizar
11 counts	3 ó 4 hebras
14 counts	2 ó 3 hebras
16 counts	2 hebras
18 counts	2 hebras

Tabla 2.1 Hebras dependiendo del count de la tela.

## III. IMÁGENES DIGITALES

### A. Píxel

La palabra píxel viene de *Picture Element* (elemento de una imagen). Se le llama píxel al componente más pequeño de una imagen al cual se le puede asignar un color. Un píxel solo puede tener un color a la vez, pero son tan pequeños que al juntarlos pueden crear sombras y mezclas de colores.

El número de colores que un píxel puede representar está dada por la cantidad de bits utilizado para simbolizarlo. Por ejemplo, si un píxel está formado por ocho bits, éste puede desplegar un máximo de 256 colores ( $2^8$ ). También se pueden utilizar 16 o 24 bits para representar un píxel. En este caso hay mayor variedad de colores que puede mostrar el píxel. A medida que hay más píxeles en una imagen, ésta tiene mayor resolución.

El ojo humano no puede percibir cada píxel individualmente, por eso al ver una imagen, lo que se ve son diferentes cambios suaves de tonalidades.

### B. Imágenes tipo vector

Este tipo de imágenes se logra utilizando definiciones matemáticas. En este tipo, las imágenes están creadas por una serie de puntos y las líneas que los interconectan. Éstos a diferencia de los bitmaps, no ocupan tanto espacio, ya que sólo contienen información sobre las líneas que contiene la imagen, no sobre la imagen en sí. Este tipo de imagen puede ser escalada, sin que se pierda mucho de la imagen. Pero, si se reduce demasiado, las líneas delgadas tienden a desaparecer.

### C. Imágenes tipo ráster

Al hablar de estas imágenes, se utiliza mucho el término bitmap, ya que no se está trabajando con líneas o figuras, sino con los píxeles. La unión de todos los píxeles son los que forman las figuras.

1. **Bitmaps.** Si se trata de definir un bitmap de una manera simple, se puede decir que es un arreglo de píxeles. Como su nombre lo dice, un bitmap se puede definir como el mapa o la estructura que corresponde bit por bit a la imagen que es desplegada en pantalla. Un bitmap se caracteriza por el ancho y el alto de la imagen en píxeles, y el número de píxeles por bit, ya que eso determina los colores que puede tener.

Pixmap se le llama a los bitmaps que representan una imagen de colores. Éstos usualmente van a tener de uno a ocho bits para cada uno de los componentes (rojo, verde y azul).

Los bitmaps ocupan bastante espacio ya que para cada píxel se guarda la información de su color. Por eso existen diferentes tipos de algoritmos de compresión, los cuales reducen el tamaño de las imágenes.

Los bitmaps o imágenes tipo ráster, son dependientes de la resolución. Al ampliar los bitmaps demasiado, tienen el problema que las imágenes se ven poco naturales, ya que se ven los diferentes píxeles. Pero al comprimirlos, puede causar pérdida de claridad.

Programas como Photoshop están diseñados para utilizar imágenes tipo ráster, ya que lo que hacen es modificar cada píxel o grupo de píxeles a fin de hacer los cambios a la imagen. Las imágenes de tipo ráster no pueden tener curvas, éstas están formadas por cuadros. Los píxeles forman una cuadrícula, y al cambiar de color cada píxel es que se va creando la imagen.

2. Profundidad del bit (*bit depth*). Las imágenes ráster están compuestas por un área de píxeles, y cada uno de esos píxeles están dados por su propia información del color. Si una imagen tiene demasiadas sombras, se necesita que cada píxel tenga mayor información para poder crear la imagen. Por eso existe una relación proporcional entre la calidad de la imagen y el espacio que esta ocupa.

La profundidad del bit esta asociado con los colores de paletas que se utilizan en los programas de tipo ráster. Los bitmaps, originalmente se les llamaba a los que eran píxeles con un solo bit de profundidad, lo cual hacía que pudiera desplegar sólo blanco o negro.

Profundidad de bits	Número de colores disponibles por píxel
4 bits	16 colores ( $2^4$ )
8 bits	256 colores ( $2^8$ )
8 bits	256 tonos de gris ( $2^8$ )
24 bits	16 millones de colores ( $2^{24}$ )
30 bits	1 billón de colores ( $2^{30}$ )
36 bits	60 billones de colores ( $2^{36}$ )

Tabla 2.2 Número de colores disponibles por píxel

Cuando se utilizan 24 bits de profundidad, se están definiendo tres canales por píxel. Cada uno de los canales (rojo, verde y azul) está formado por ocho bits. Lo cuál hace que esta pueda tener 16 millones de colores diferentes.

3. Espacios de colores. Un espacio de colores es un sistema matemático para representar colores. También un espacio de colores se puede definir como los colores posibles que pueden ser producidos por un dispositivo de salida en particular; tales como un monitor, una impresora de colores, entre otros. Los espacios de color surgen de la necesidad de estandarizar descripciones del color.

a. RGB. El ojo contiene dos clases de receptores: bastones y conos. Mientras que los bastones ven los grises, los conos permiten que el cerebro perciba tonalidades del color. Existen tres tipos de conos, el primero es sensible a la luz rojo-anaranjada, el segundo a la luz verde y al tercero a la luz azul-violeta. Cuando se estimula un solo cono, el cerebro percibe el color correspondiente. Es decir, si se estimulan los conos verdes, los seres humanos perciben el color verde. Si se estimulan los conos rojo-anaranjados, se ve el rojo. Si los conos verdes y rojo-anaranjados se estimulan simultáneamente, se ven tonalidades amarillas.

En 1931, Vienna-based Commission Internationale de l'Eclairage (CIE) desarrolló el modelo *Red-Green-Blue* (RGB), el cual utiliza los colores primarios transmitidos por la luz. Este modelo es un modelo aditivo, es decir que se basa en unir, sumar los colores rojo, verde y azul para obtener el blanco.

Los monitores de las computadoras emiten luz y cuando se mezclan los diferentes espectros de luz se crea el color que se quiere presentar al usuario. Este espacio de color se basa en los colores primarios de la luz que pueden ser percibidos por el ojo: rojo, verde y azul. La intensidad de cada punto de la imagen corresponde a la intensidad de color almacenada en la información RGB para ese píxel. RGB es un espacio de color aditivo, es decir que cuanto mayor luz se le da, el color se vuelve más claro. Al mezclar todos los espectros de colores se produce la luz blanca, por eso en el espacio RGB cuando todos los puntos se iluminan totalmente, se crea el color blanco.

Los colores grises se crean al mezclar iguales cantidades de rojo, verde y azul en bajas intensidades. Al mezclar iguales cantidades de rojo y verde se perciben como amarillo, iguales cantidades de verde y azul se perciben como turquesa. Al mezclar iguales cantidades de azul y rojo se perciben tonalidades moradas.

Para almacenar la información RGB, se almacenan en un grupo de bits la información correspondiente al color rojo, en otro grupo el color verde, y en otro el azul. Dado que el ojo humano puede percibir 200 niveles de brillantes, se eligieron 256 niveles (ocho bits) para representar todos los posibles colores que puede distinguir el ojo humano, y así crear mayor fidelidad en las imágenes. Al utilizar ocho bits por color, implica que se necesitan 24 bits para representar un color RGB. Esto hace que con 24 bits se pueda representar un promedio de 16 millones de colores diferentes.

En el espacio de color RGB, para cambiar los colores, se cambia la distancia entre los colores RGB. Pero si se quiere cambiar la intensidad, se necesita proyectar el vector hacia el color de RGB más fuerte. Para cambiar la brillantez lo que se hace es mantener la diferencia entre los colores RGB, pero incrementar los diferentes colores simultáneamente.

b. **CMYK.** Otro de los modelos utilizados es un modelo que se basa en la luz reflejada. Este modelo es un modelo substractivo, logra obtener el blanco al substraer los pigmentos cyan, magenta y amarillo.

c. **Bitmap.** Este modelo de color, se refiere a aquel que utiliza solo blanco y negro para representar la imagen.

## **IV. GENERACIÓN DE PATRONES A PARTIR DE IMÁGENES DIGITALES**

### **A. Entrada**

La aplicación recibe como entrada inicial una imagen, que es la imagen de la cual se quiere generar el patrón. La imagen debe ser simple, ya que la dificultad del patrón va a ser proporcional a la de la imagen.

En esta aplicación se decidió que la imagen fuera un bitmap. Esta tiene almacenados todos los puntos de la imagen, y los datos de color no tienen compresión alguna.

Además recibe la lista de colores DMC que se tienen, es decir, los hilos que la persona que va a bordar tiene disponibles. Si la persona no quiere ingresar una lista, podrá utilizar como lista por omisión una que contenga todos los colores DMC, y esta será la que reciba como entrada.

### **B. Salida**

A partir de la imagen, luego del proceso, lo que se quiere obtener es un patrón con símbolos en el que se muestren los lugares de la cuadrícula en los que hay que bordar crucetas. Y junto con este patrón, una lista de colores que haga corresponder un color de hilo DMC con cada símbolo.

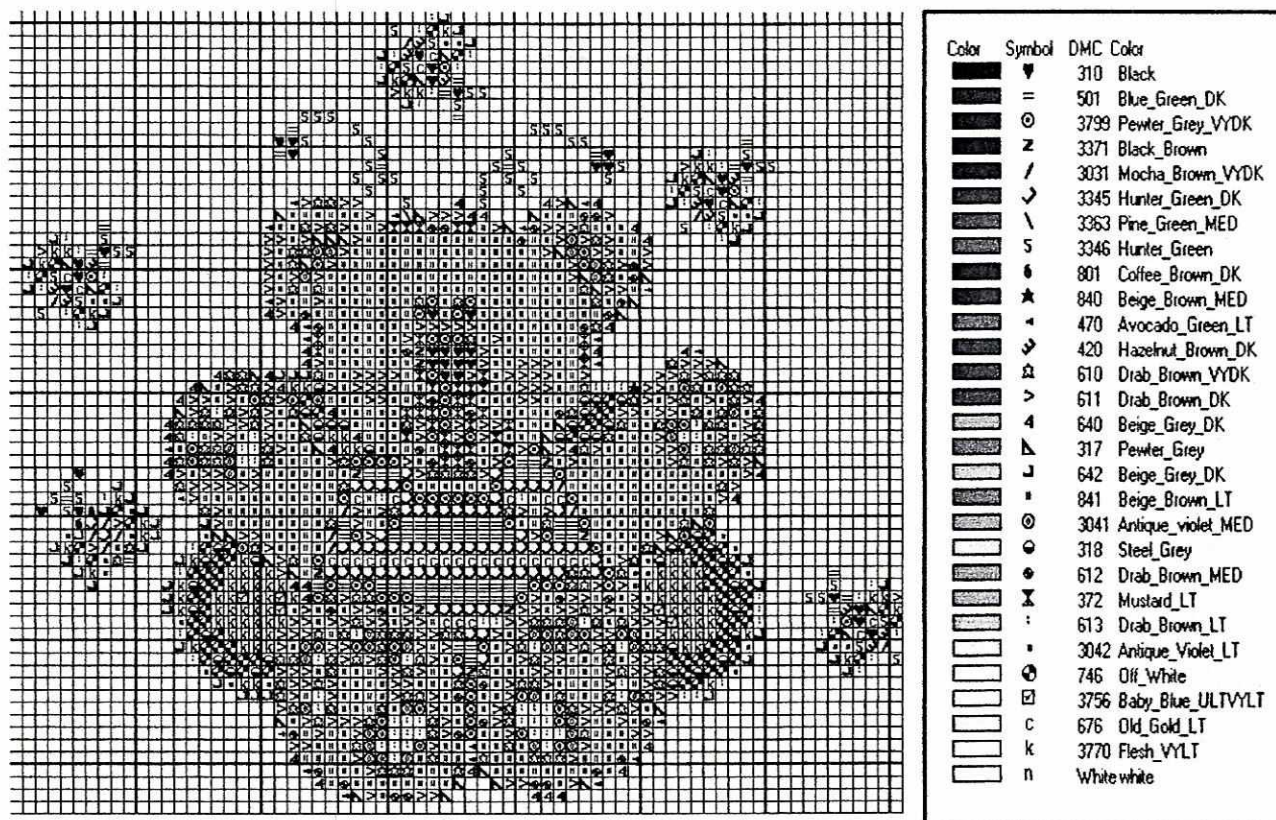


Fig 4.1 Patrón de símbolos generado por la aplicación

### C. Proceso de creación de patrones

Para la generación de patrones de bordado de cruceta a partir de imágenes, se requiere que la imagen pase por una serie de transformaciones. En esta sección se verá de forma general el proceso por el que se pasa para generar el patrón, a fin de tener una visión general de la aplicación, y a lo que se quería llegar. En las secciones subsiguientes, se explicarán y analizarán a detalle los algoritmos utilizados en cada una de las fases del proceso.

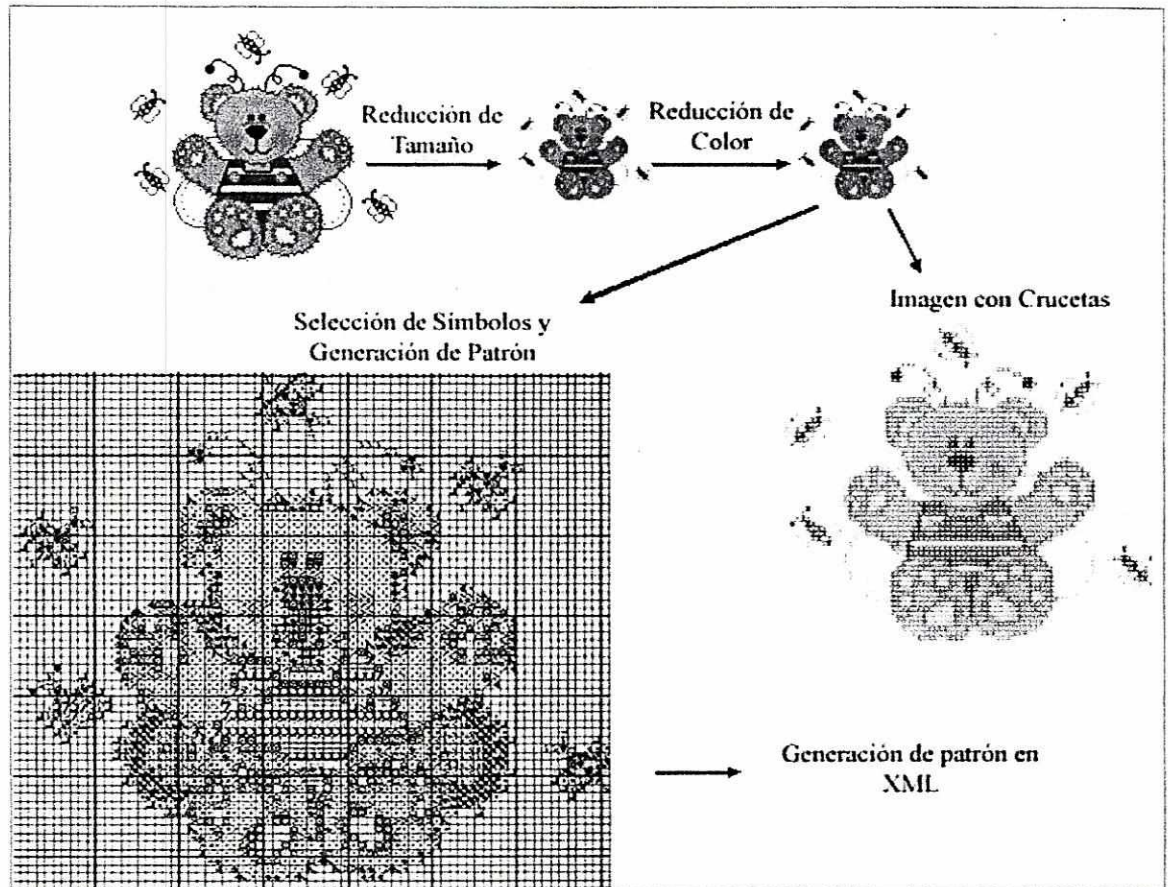


Fig 4.2 Proceso de creación del patrón

Para poder llegar a generar la salida final en esta aplicación, se lleva a cabo el siguiente proceso:

- Reducción de tamaño de la imagen: Esta fase del proceso busca disminuir la cantidad de píxeles que tiene la imagen. Ya que en el momento de generar el patrón de bordado, cada píxel de la imagen va a representar una cruz a bordar. Es importante que en el patrón se pueda generar en la menor cantidad de píxeles.
- Reducción de colores: Dado que la paleta de colores DMC, no cuenta con los 17 millones ( $2^{24}$ ) de colores que se pueden representar en un píxel de una imagen digital, es necesario reducir los colores a aquellos que se tienen disponibles. Además, las personas que bordan cuentan con un número limitado de colores. Entonces, esta aplicación tiene como objetivo, generar una imagen con los colores que se tienen. Durante el proceso de creación de patrones, esta fase se colocó luego de la reducción de tamaño, ya que los algoritmos utilizados para reducir tamaño, lo hacen promediando colores o haciendo cambios de color a medida que reducen la imagen.

- Imagen con crucetas: Antes de generar el patrón original, se puede crear una imagen de crucetas. Ésta muestra con cruces de colores cómo se vería la imagen bordada. A diferencia del patrón de símbolos ésta sirve para que el usuario del patrón decida si desea bordar una imagen así o no, ya que se puede apreciar como se vería ya bordada.
- Selección de símbolos y generación de patrón: Una vez que fue hecha la reducción de tamaño y de colores, se puede generar el patrón de bordado de cruceta. Primero, a cada color utilizado se le asigna un símbolo. Luego se seleccionan a partir de cada píxel de la imagen transformada, el color al que corresponden y por lo tanto su símbolo. Y luego se copia el símbolo en una nueva imagen. Así se va creando el patrón.
- Generación de patrones XML: Una vez se tiene el patrón de bordado ya hecho se puede crear un archivo en formato XML que almacene la información importante del patrón. En dicho XML se almacena la paleta de colores utilizados, y las posiciones de las cruces y los colores que se tienen.
- Transformación de formato XML para un formato similar al utilizado por las máquinas bordadoras Tajima: El XML generado se pasa por una hoja de estilo XSL, la cual la transforma, generando otro archivo XML que tiene un árbol con la información que utilizan las máquinas bordadoras Tajima.
- Bordado: Con el patrón con símbolos y la lista de colores, se puede realizar el bordado en tela cuadrículada. La persona que la borda sólo tiene que hacer un punto de cruz en la posición indicada en la cuadrícula con el color que corresponde al símbolo. Si se quiere bordar en máquinas, se debe utilizar una aplicación que le de formato al XML que tiene la información necesaria a un formato idéntico al que utilizan las máquinas. O lo ideal, que es que en el futuro estas máquinas reciban directamente el XML y con esto borden, ya que XML va a llegar a ser una especificación estándar.

## V. REDUCCIÓN DE TAMAÑO DE IMÁGENES

Las imágenes muy grandes requieren de mucho espacio de almacenamiento y de mucho tiempo para su despliegue. En muchas aplicaciones una versión pequeña de la imagen resulta ventajoso, pero en el caso de este proyecto el tener una imagen más pequeña resulta necesario.

### A. Cortar la imagen

Una de las maneras más simples de reducir el tamaño de una imagen es cortándola, seleccionando solamente la parte de la imagen que es necesaria, e ignorando lo demás. Este método funciona cuando toda la información importante de la imagen se encuentra en una región.

Es mejor que esta parte del proceso la realice la persona en vez de la máquina, ya que la persona es quien decide cuál es la parte importante de la imagen que se quiere bordar. El programa desarrollado no corta la imagen, ya que él usuario podría hacerlo desde cualquier aplicación para editar imágenes. Se desea que la imagen de entrada a la aplicación desarrollada sólo tenga el área que se quiere bordar.

### B. Salto de píxeles

Otra manera de reducir una imagen es ir saltando píxeles uniformemente a través de la imagen. Este método es rápido, pero produce irregularidades en la imagen. En una imagen que ya pasó por el proceso de dithering (capítulo VI), el ir eliminando píxeles de esta forma puede causar efectos no deseados, ya que los colores resultantes pueden ser totalmente diferentes a los buscados. También al saltar píxeles se puede eliminar líneas que son realmente importantes.

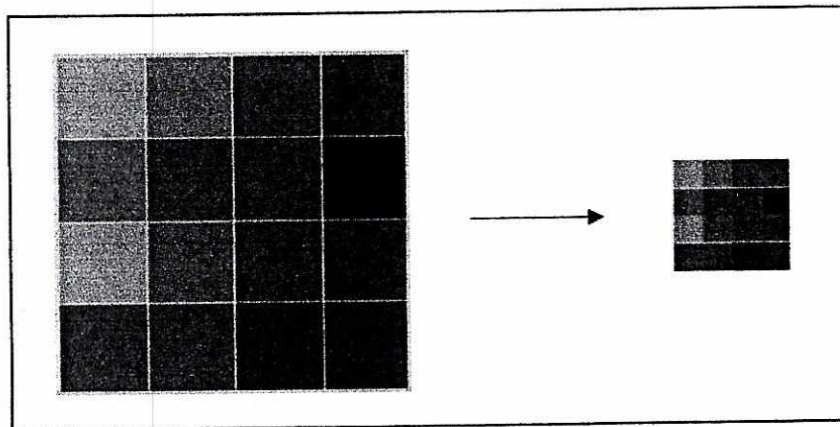


Fig 5.1 Imagen reducida con salto de píxeles. Se saltaron 2 de cada 3 píxeles

## C. Promediar píxeles

Para reducir los problemas que surgen al hacer un simple salto de píxeles, se puede realizar un promedio de los píxeles adyacentes en lugar de sólo dejar píxeles sin utilizar. Lo único malo en este algoritmo es que en ocasiones la imagen se vuelva borrosa.

1. Promediar píxeles. Este es uno de los algoritmos más simples; toma en cuenta la escala a la que se quiere reducir la imagen y cuantos píxeles hay que unir para formar un solo píxel de la imagen original. Luego se van tomando grupos adyacentes de esa cantidad de píxeles y se promedian, sumando todos los canales RGB (rojo, verde, azul) de los diferentes píxeles, y dividiéndolos dentro de la cantidad de píxeles que se promedió.

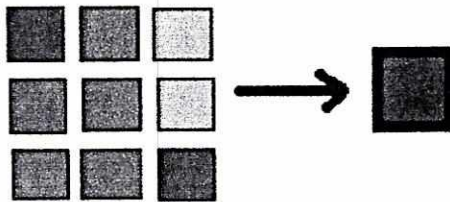


Fig 5.2 Ejemplo del promedio de píxeles. En este ejemplo cada cuadro representa un píxel.

### Algoritmo de Promedio de Píxeles

```
int anchocuadro=pixeles_por_punto;
int altocuadro=pixeles_por_punto;

int ancho = Imagen->Width/ anchocuadro;
int alto = Imagen->Height/ altocuadro;

int x1=0,y1=0;
for (y = 1; y< Imagen.Alto;y++)
{
    for (x=1; x<Imagen.ancho; x++)
    {
        Rect.Left = x * anchocuadro ;
        Rect.Right = Rect.Left + anchocuadro - 1;
        Rect.Top = y* altocuadro;
        Rect.Bottom = Rect.Top + altocuadro - 1;
        nuevo_color=promediar_color (Imagen, Rect);
        Imagen[x][y]=nuevo_color;
    }
}
```

Donde la función promediar\_color es la siguiente

```
pixeles = (AArea.Right - AArea.Left + 1) * (AArea.Bottom - AArea.Top + 1);
// Suma toda el área
for (y = AArea.Top; y<=AArea.Bottom; y++)
{
    for (x=AArea.Left; x<=AArea.Right; x++)
    {
        totalb = totalb + Imagen[x][y].Blue;
        totalg = totalg + Imagen[x][y].Green;
        totalr = totalr + Imagen[x][y].Red;
    }
}
Color.p.red = totalr / pixeles;
Color.p.green = totalg / pixeles;
Color.p.blue = totalb / pixeles;
```

2. Promediar píxeles utilizando matriz ponderada o de pesos. Este algoritmo es muy similar al anterior, con la diferencia de que en el momento de promediar los píxeles utiliza una convolución, multiplica los píxeles a promediar por una matriz. La matriz utilizada en este algoritmo fue:

Matriz ponderada[4][4]=

3	7	7	3
7	15	15	7
7	15	15	7
3	7	7	3

La suma de todos los números de la matriz da 128, por lo tanto luego de la convolución el número de color resultante se divide entre 128. Este algoritmo también se aplica sobre cada uno de los canales de color.

### Promedio de píxeles utilizando matriz de pesos

Donde la función `promediar_color` es la siguiente

```
totalr = 0;
totalg = 0;
totalb = 0;
int x=-1, y=-1;
for (y = Area.Top-1; y<=Area.Top+2; y++)
{
    for (x=Area.Left-1; x<=Area.Left+2; x++)
    {
        totalb = totalb + matriz_ponderada[x][y]*Image[x][y].Blue;
        totalg = totalg + matriz_ponderada[x][y]*Image[x][y].Green;
        totalr = totalr + matriz_ponderada[x][y]*Image[x][y].Red;
    }
}
Color.p.red = totalr /128 + 0.5;
Color.p.green = totalg /128 + 0.5 ;
Color.p.blue = totalb /128 + 0.5;
return Color;
```

Al utilizar este algoritmo, cuando se quiere reducir la imagen a más de cuatro veces el tamaño de la original, la matriz de convolución sigue siendo de 4x4. Entonces hay píxeles que no se toman en cuenta. Esto es el resultado de multiplicar los píxeles por cero. Si se quisiera hacer un algoritmo más eficaz se podría hacer una matriz de pesos dinámica, que fuera creciendo con el tamaño de la escala a la que se quiere reducir la imagen.

## D. Algoritmo del cociente para escalar la imagen

Este algoritmo se tomó de las funciones de un programa llamado ImageMagick, que fue creado por John Cristy en 1992. Esta es una adaptación del algoritmo utilizado en el programa pnmscale de NetPBM.

Este algoritmo recibe un bitmap como entrada y lo escala por los factores necesarios para lograr la medida deseada en la imagen de salida. Por medio de este algoritmo se puede ampliar las imágenes (si el factor de escalamiento es mayor de 1) o reducir las (si el factor de escalamientos es menor de 1).

Este algoritmo mezcla los colores de los píxeles adyacentes, para producir píxeles de salida que contienen información de múltiples píxeles. Esto implica que los colores de la imagen de salida, puede que no hayan estado en la de entrada.

Además, en esta reducción de tamaño cada dimensión se modifica independientemente sin consideración del cociente entre uno y otro.

El algoritmo utiliza diferentes métodos de modificar el tamaño de la imagen. Uno de ellos es el muestreo discreto, en lugar de promediar los píxeles, escoge el color que cubre la mayoría del área de los píxeles a promediar. La ventaja de esto es que la imagen final tiene la misma paleta de colores que la imagen inicial. La desventaja de este algoritmo es que distorsiona la imagen.

La calidad de imágenes no es mucho mejor en comparación con las obtenidas por el algoritmo de promedios utilizando pesos, aunque sí mejora un poco la claridad de la imagen. Pero, a pesar de ser un poco más complejo, se agregó el algoritmo a esta aplicación ya que permite modificar el tamaño de las imágenes a un número de columnas y filas específico. En el bordado de cruceta, esto puede llegar a ser importante, ya que se puede necesitar bordar un cuadro que ajuste en cierto marco o que ocupe el área de un adorno en particular. Éste es un algoritmo de radio o cociente simple.

## E. Análisis de los algoritmos en la aplicación

La aplicación permite que el usuario seleccione el algoritmo a utilizar. Sin embargo, a continuación se hará una comparación para ofrecer una mejor solución al usuario. En la gráfica que sigue se redujo el tamaño de la imagen a la mitad.

El algoritmo de salto de píxeles, aunque mantiene los colores más fijos, no es una solución viable, ya que elimina muchos píxeles importantes, y queda una imagen distorsionada.

Luego el promedio de píxeles con pesos, al igual que el algoritmo del cociente, trata de crear una imagen más uniforme, donde no se sientan los saltos entre píxeles. Pero, eso hace que se vean un poco más borrosas las imágenes. En ocasiones, en el bordado se puede escoger esta selección cuando se desea que la disminución de colores sea uniforme.

La ventaja del algoritmo del cociente es que permite seleccionar las medidas exactas del bordado, en cambio los otros algoritmos modifican el tamaño de la imagen uniformemente, en números enteros nada más, es decir, la reducen a la mitad, a tres veces el tamaño original, etc.

El que mejor resulta para la creación de patrones de bordado de cruceta es el promedio de píxeles. Éste, si bien puede causar algunas alteraciones en el color, da una imagen más clara, y presenta cambios de color entre los píxeles adyacentes, lo cual es útil para el bordado de cruceta. A este algoritmo, que es la solución propuesta, se le hizo una modificación a fin de mejorarlo. Se eliminó el color blanco del promedio en las áreas donde había muchos píxeles de color blanco. Esto busca que en los contornos no promedie el blanco del exterior con los colores de los bordes, ya que eso genera muchos píxeles en diferentes tonalidades de grises, los cuales no son necesarios.

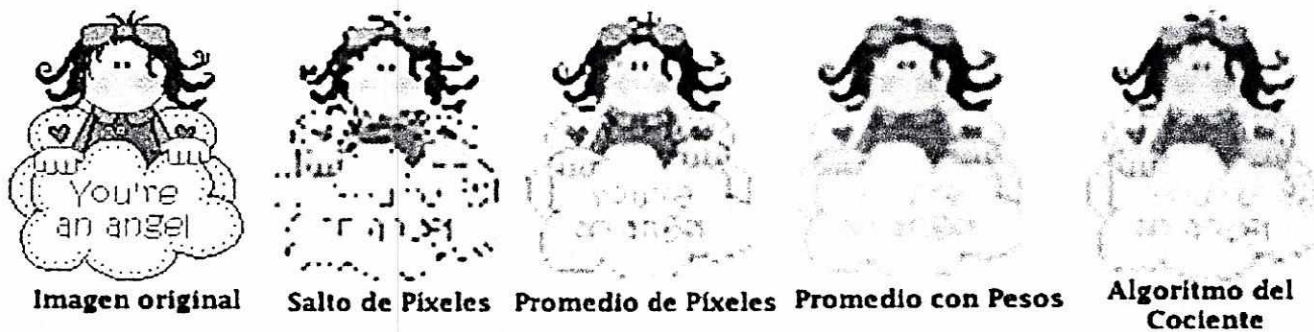


Fig 5.3 Comparación de reducción de tamaño utilizando diferentes algoritmos.

## VI. REDUCCIÓN DE COLORES

Tanto en la industria de bordado como en los bordados hechos a mano se tiene un número restringido de colores que se pueden utilizar. Por ejemplo, todas las combinaciones RGB para un píxel podrían ser hasta de 16 millones de colores, mientras que la lista de lustrinas DMC para bordado de cruceta no es mayor a 1000 colores diferentes. Así que si se quiere bordar imágenes es fundamental poder crear con los colores que se tienen una imagen que se asemeje a la original.

Si se toma como base la industria de bordado, las máquinas tienen diferentes agujas por cabeza. En cada aguja se colocan diferentes colores de hilo. Mientras mayor sea el número de agujas que tiene cada cabeza, mayor es el precio de las máquinas. Por eso se trata de utilizar pocos colores en los bordados, las máquinas varían de 6 a 15 agujas. Cada cambio de color en un bordado, tanto en máquinas como en bordados hechos a mano, representa pérdida de tiempo. Por eso el limitar los colores resulta de gran importancia.

No sólo en los bordados resulta importante la reducción de colores. La mayoría de dispositivos de salida (despliegue e impresión) no tiene la capacidad de reproducir imágenes en colores originales (*true colors*). Por ejemplo, los monitores están limitados a cierta cantidad de colores, como resultado de los diferentes niveles de iluminación que pueden combinarse utilizando tres puntos de fósforo. En muchas ocasiones el número de colores de una imagen va a sobrepasar la capacidad del dispositivo para desplegar la imagen; por ejemplo, puede ser necesario presentar una imagen de 256 colores en un sistema de despliegue de 16, o imprimir una fotografía escaneada en una impresora matricial.

En ocasiones se puede tener interés en procesar imágenes para desplegarlas en *hardware* de este tipo, que está basado en una paleta de colores. Por consecuencia los colores tienen que ser reducidos drásticamente a una paleta de colores limitados. Ejemplos de este tipo de *hardware*, podrían ser algunas impresoras que sólo cuentan con un número discreto de colores; pantallas publicitarias que despliegan utilizando sólo cierta cantidad de lámparas de colores diferentes, o en impresiones sobre playeras, tazas, etc.

Otra aplicación similar es aquella en que se convierten las imágenes para que puedan ser vistas en los diferentes navegadores de Internet. Las imágenes pasan por una reducción de colores a fin de proveer una imagen que tenga un formato que se pueda utilizar en Internet.

## A. Selección del mejor color

Para la selección del mejor color la clave está en la comparación del color original con el color que se quiere colocar en su lugar. Para esto lo que se hace es comparar cuánto varía un color del otro. La varianza entre colores se mide sumando las diferencias de cada uno de los canales de color (Red, Green, Blue) del color original y el nuevo color. Además, para calcular la varianza, hay que evaluar la proporción de los tres colores entre sí, ya que en el bordado es preferible bajar o subir la brillantez de un color que cambiarlo por otro color diferente, aunque éste sea más cercano al original.

### Comparación al color más parecido

```
int calcula_varianza (color_anterior , color_nuevo)
{ int rg,gb,br; //Diferencias entre rojo-verde, verde-azul y rojo-azul
  int diferencia_entre_colores = (abs(color_anterior.rojo-color_nuevo.rojo) +
    abs(color_anterior.azul-color_nuevo.azul) +
    abs(color_anterior.verde - color_nuevo.verde));

  rg=abs(abs(color_anterior.rojo-color_anterior.verde)-abs(color_nuevo.rojo-color_nuevo.verde));
  gb=abs(abs(color_anterior.verde- color_anterior.azul)-abs(color_nuevo.verde- color_nuevo.azul));
  br=abs(abs(color_anterior.azul- color_anterior.rojo)-abs(color_nuevo.azul- color_nuevo.rojo));
  varianza=diferencia_entre_colores+ (rg+gb+br)/5;
  return varianza;
}
```

En cuanto a la selección de colores la aplicación desarrollada tiene dos opciones que el usuario puede elegir.

1. Utilizar paleta de colores creada. En ésta el usuario puede agregar colores DMC a una paleta. Esto puede hacerlo añadiendo a la paleta los colores de hilo que tenga, o limitándola a los colores que quiera utilizar. Utilizando la paleta creada, se hace la reducción de colores, al más parecido, utilizando la varianza que se calcula, con el algoritmo mostrado anteriormente.

2. Selección de colores utilizando toda la gama de colores DMC. Otra opción que se tiene es utilizar toda la gama de colores DMC. Esta opción es un poco más tardada, pero como resultado calcula los colores más similares entre sí. Lo que hace esta opción es comparar los colores de los píxeles de la imagen original con todos los colores DMC, y ver cuántas veces se utilizaría cada píxel si se tuviera toda la paleta DMC. Luego, dependiendo de la cantidad máxima de hilos que el usuario quiera utilizar, el programa selecciona los colores de la paleta DMC que se utilizaron más veces durante la comparación. Y son esos los colores que agrega a la paleta, con la que se va a realizar la reducción.

## B. Dithering

Dithering es uno de los procesos que se utiliza cuando no se pueden desplegar ciertos colores de una imagen. Por ejemplo, si un gradiente se necesita desplegar en un sistema que no puede desplegar todos los colores de dicho gradiente, entonces la imagen necesita pasar por un proceso de dithering. Este proceso mezcla los colores que se tienen para aproximar los colores que no existen en la paleta de colores de la imagen. Por ejemplo, en las imágenes en los periódicos se utiliza mucho esta técnica. Al acercarse a una imagen en el periódico se observan muchos puntos negros y blancos, que al mezclarlos son los que crean una imagen con diferentes tonos de gris. Al aplicar dithering a las imágenes se toma en cuenta las propiedades de la percepción humana, ya que en el ojo al mezclarse los diferentes colores adyacentes, se observa otro color diferente.

El aplicar dithering a las imágenes mejora la apariencia de una imagen cuando son pocos los colores disponibles.

Existen dos tipos de dithering: el aleatorio y el regular u ordenado. El ordenado utiliza un trazo regular, como un patrón de ajedrez, y el aleatorio o *random* que, como su nombre lo indica, utiliza patrones aleatorios. El dithering aleatorio es el mejor para fotografías.

1. Floyd-Steinberg Dithering. Uno de los algoritmos de dithering es el de Floyd-Steinberg. Éste es uno de los algoritmos más lentos, pero es de los que proveen una mejor imagen. Este algoritmo es un ejemplo de las técnicas de difusión de error. Los algoritmos que utilizan difusión de error buscan para cada punto de la imagen el color más cercano disponible. Floyds-Steiberg busca el mejor reemplazo de color para cada píxel, pero busca eliminar los errores que introduce al no encontrar el color exacto.

La solución propuesta por el algoritmo es considerar el resultado de cambio de color en un píxel al hacer el cambio de color en sus vecinos. La clave de los algoritmos que utilizan la técnica de difusión de error es: el error resultante de cambiarle color a un píxel se mueve a sus vecinos y por consecuencia afecta su cambio de color

Lo primero que se realiza es definir el error. Este va a estar dado por la diferencia del color del píxel original y el píxel por el cual se está cambiando. Luego, existen muchas maneras diferentes de dividir el error, y muchas formas diferentes de recorrer la imagen. En este proyecto se utilizó de izquierda a derecha y de arriba hacia abajo.

$$e = I_{in}(x,y) - I_{out}(x,y)$$

Luego este error se divide entre los pixeles adyacentes, de la siguiente manera:

$$I_{in}(x+1,y) = I_{in}(x+1,y) + 7/16 e$$

$$I_{in}(x+1,y+1) = I_{in}(x+1,y+1) + 3/16 e$$

$$I_{in}(x,y+1) = I_{in}(x,y+1) + 5/16 e$$

$$I_{in}(x-1,y+1) = I_{in}(x-1,y+1) + 1/16 e$$

	x	7/16
1/16	5/16	3/16

Tabla 6.1 Difusión del error utilizando Floyd-Steinberg dithering



Fig 6.1 Comparación de dithering contra cambio de color simple.

#### Algoritmo de dithering

```

 $\alpha = 7/16$ 
 $\beta = 3/16$ 
 $\gamma = 5/16$ 
 $\delta = 1/16$ 
for (int y = 0; y < Original.Height-1; y++)
{
  x1=0;
  int pos=0;
  for (int x = 1; x < Original.Width-1; x++)
  { pos++;
    nuevo_color=threshold(Imagen.pixel(x,y));
    error = Imagen.Pixel(x,y) - nuevo_color;
    Imagen.Pixel(x+1,y) = Imagen.Pixel(x+1,y)+  $\alpha$ *error;
    Imagen.Pixel(x-1,y+1) = Imagen.Pixel(x-1,y)+  $\beta$  *error;
    Imagen.Pixel(x,y+1) = Imagen.Pixel(x+  $\gamma$  *error);
    Imagen.Pixel(x+1,y+1) = Imagen.Pixel(x+1,y+1)+  $\delta$ *error;
  }
}

```

## C. Análisis de algoritmos

En la imagen anterior se pudo observar cómo el utilizar dithering puede causar que una imagen sea más clara cuando se están utilizando pocos colores. Dithering resulta útil cuando la paleta de colores que se tiene son solamente diferentes tonos de grises. Esto se debe a que el error que se da por los cambios de colores se va esparciendo por la imagen.

Este algoritmo es más funcional cuando las imágenes que se tienen no tienen colores tan parejos, como las fotos o pinturas. Pero para la mayoría de imágenes que se quieren bordar, el algoritmo de dithering no le agrega valor a la imagen resultante. Esto se debe a que en el momento de bordar es más fácil tener imágenes con varias crucetas del mismo color juntas, ya que esto hace más fácil el bordado. En cambio el dithering va agregando píxeles de otros colores para aproximar los colores originales. Y el tener que bordar píxeles de muchos colores diferentes esparcidos por la imagen no es algo deseable en los bordados, ya que eso requiere hacer cortes de hilo, o pasar hilo en forma desordenada en la parte de atrás del bordado, además de hacer más difícil el control de los píxeles que se van cociendo. Son esas crucetas sueltas las que hacen que el trabajo de bordado lleve más tiempo.

Además, en la mayoría de ocasiones el dithering en las imágenes coloca píxeles de colores fuera de lugar, ya que en sí la percepción total de la imagen es la importante. En la imagen un píxel no es tan obvio, pero el colocar una cruceta de un color que esta fuera de lugar sí se puede percibir. Es por esas razones que en la aplicación, aunque se permite utilizar este algoritmo, no es recomendable. Se recomienda que se utilice el cambio simple de colores por el más parecido.

En cuanto a la selección de la paleta de colores a utilizar se recomienda utilizar la paleta establecida por el usuario si ya se tienen muchos colores de hilo, y no se quiere comprar nuevos hilos. Ya que ésta permite utilizar lo que ya se tiene.

Pero si al usuario no le importa el tener que comprar nuevos hilos, es recomendable dejar que el programa seleccione los colores más utilizados, usando todo el listado de colores DMC. Y que se limite en el programa el máximo de colores que se quiere tener en la imagen. Utilizando esto, a medida que se incrementa el número de colores, los colores de la imagen resultante se parecen más a los de la original.

El dejar que sea el programa el que seleccione la paleta tiene la ventaja que la lista de los colores de la paleta a utilizar se calcula utilizando la imagen original. Entonces aún luego de promediar píxeles la imagen conserva la mayoría de los colores originales, lo cual no ocurre cuando la reducción de color se hace utilizando cualquier otra paleta.

La desventaja del método que añade en la paleta los colores más utilizados de la lista DMC es que no siempre los colores más utilizados van a ser los más importantes. Puede que existan píxeles que se utilizan en pocas ocasiones, pero que le agregan un contraste especial a la imagen.

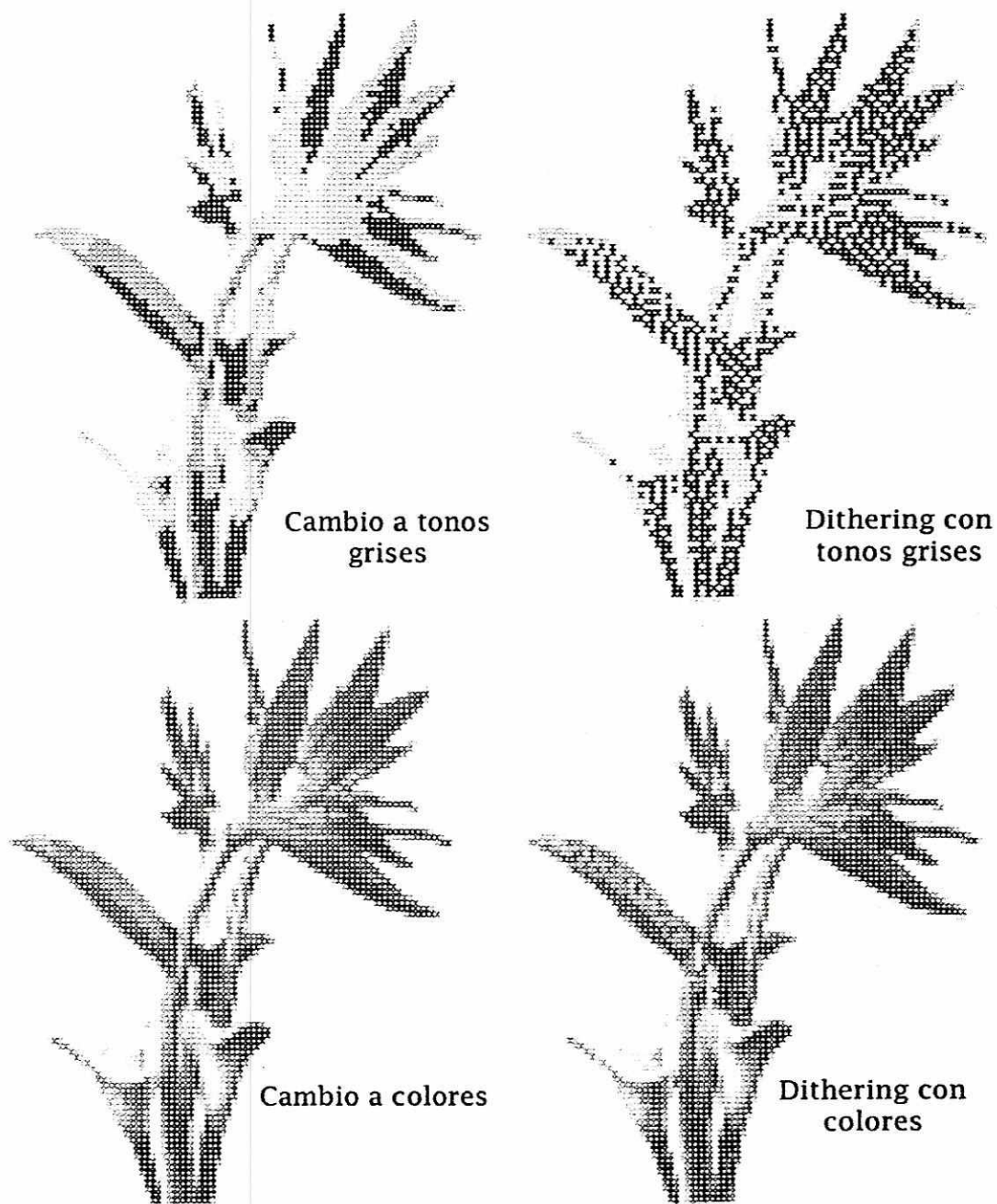


Fig 6.2 Comparación de cambios de colores. Con colores y tonos grises.

## VII. CONTORNOS

Los bordes o contornos se pueden definir como el cambio significativo de tonos de grises en una imagen.

Al detectar bordes basados en el gradiente se asume que los bordes tienen un gradiente alto. En los bordes se espera que haya un cambio rápido de intensidad. La magnitud de la tasa de cambio de intensidad es la intensidad del borde. Si se calcula el gradiente en regiones uniformes se termina con un vector de magnitud 0, lo que indica que allí no existen píxeles que definan el borde.

En imágenes naturales usualmente no se tienen bordes, ya que no se tiene una discontinuidad, como en las figuras.

La manera más simple de calcular los cambios de intensidad es aplicando un cambio de color. Si la magnitud del gradiente es mayor que el píxel que tuvo cambio de color, entonces el píxel es un borde. En este tipo de algoritmos la fuerza o intensidad del borde la da la magnitud del gradiente, y la dirección del borde está dada por el ángulo del gradiente.

En las funciones discretas no se puede calcular un gradiente, pero se puede estimar utilizando algunos operadores, por ejemplo el operador de Sobel.

Para detectar bordes y líneas en imágenes existen diferentes operadores. La mayoría de ellos utilizan máscaras de convolución, y se basan en operaciones diferenciales. Estos diferenciales se utilizan en cambio de color en los píxeles y sus direcciones. La mayoría de operadores se ve afectado por imágenes con distorsiones (ruidosas). Por eso antes de aplicar los operadores las imágenes deben filtrarse, si es posible. La idea principal de los operadores de detección de contornos o bordes se basa en comparar los píxeles con sus vecinos. Al comparar sus colores, si un píxel es muy similar al color de sus vecinos, entonces es poco probable que el píxel sea parte del borde; en cambio, al haber un cambio drástico, es muy probable que este píxel forme parte del borde. El único problema es que en las imágenes no siempre funciona así, a veces ocurren cambios suaves; entonces es difícil encontrar los bordes.

### A. Operador de Sobel

Lo que hace el operador de Sobel es detectar los bordes horizontales y los verticales de manera separada. Y luego se combinan ambas direcciones de bordes. Estos gradientes  $SX$  y  $SY$  se calculan utilizando máscaras de convolución.

Operadores de Sobel:

$$g_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad g_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

La magnitud y dirección del borde se calculan utilizando los resultados de aplicar las dos máscaras de Sobel.

$$\mathbf{g} = \begin{bmatrix} g_x \\ g_y \end{bmatrix}$$

$$g = \sqrt{g_x^2 + g_y^2}$$

$$\theta = \tan^{-1} \left( \frac{g_y}{g_x} \right)$$

Lo que hace el algoritmo detector de bordes de Sobel es utilizar una matriz de convolución simple, para crear una serie de gradientes. Matemáticamente se puede describir como la aplicación de la convolución  $K$  al grupo de píxeles  $p$ .

$$N(x, y) = \sum_{k=-1}^1 \sum_{j=-1}^1 K(j, k) p(x - j, y - k)$$



Fig 7.1 Ejemplos utilizando operadores de Sobel

## B. Operador de Laplace

El operador de Laplace utiliza la misma idea que el de Sobel, ya que compara cada píxel con sus píxeles vecinos. Pero este, a diferencia del de Sobel, sólo está formado por una máscara. Y en ésta los cambios en todas las direcciones tienen el mismo valor.

Si en una imagen hay un cambio de color bastante obvio, la primera derivada de la función tendrá un pico. Por lo tanto en la segunda derivada estos puntos son los ceros.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Lo que hace el operador de Laplace para encontrar los bordes en las imágenes es buscar dónde se encuentran los ceros de la segunda derivada de la imagen. El calcular la segunda derivada de la imagen hace que ésta sea muy sensible al ruido.

El objetivo de la detección de los bordes es localizar las regiones donde la intensidad cambia rápidamente. Para funciones de una dimensión, este gradiente se considera grande cuando la primera derivada es mayor que el cambio de color. Es decir, cuando la primera derivada alcanza un máximo, que es cuando la segunda derivada es 0.

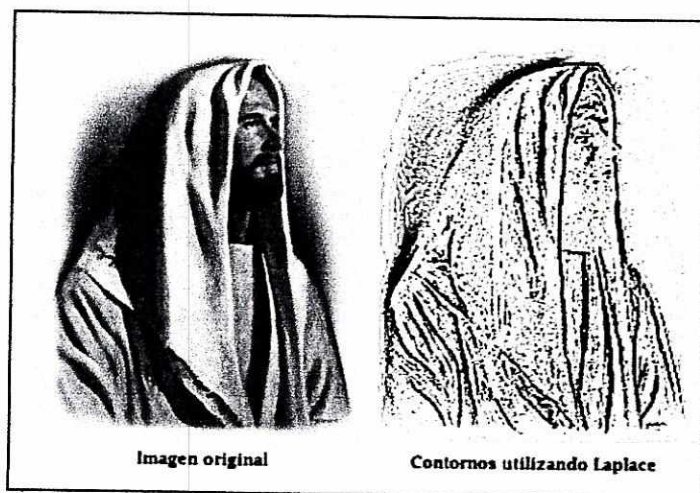


Fig 7.2 Imagen utilizando operador de Laplace

## C. Análisis de algoritmos

Ambos algoritmos expresan claramente los contornos de la imagen. La ventaja de Sobel, sobre Laplace es que Sobel es menos sensible al ruido, entonces sólo encuentra los contornos más importantes.

Una desventaja de ambos es que cuando se tienen imágenes con bordes bien definidos y que son anchos, entonces calcula como si hubiera un contorno en cada lado del borde.

En bordado de cruceta, los contornos juegan un papel muy importante, aunque estos no se realicen con punto de cruz. Estos se hacen con pespunte. En los bordados, los contornos hacen resaltar partes importantes de la imagen, e identifican los lugares importantes de cambio. Estos deben ser muy finos. Y no se quiere tener bordes en todos los cambios de intensidad de la imagen. Por esto, no se considera que ninguno de estos algoritmos se adapte a las necesidades del bordado de cruceta.

## VIII. COMPARACIÓN DE IMÁGENES

Dependiendo de la imagen, el error resultante de la reducción de error puede ser obvio o invisible. Las imágenes con altas frecuencias espaciales (tales como el cabello o grama) mostrarán mucho menos error que imágenes con menores frecuencias, como por ejemplo imágenes con áreas suavemente sombreadas (como los rostros). Esto se debe a que los contornos que surgen del proceso de cambio de color son encubiertos por las altas frecuencias de la imagen.

Para medir la diferencia entre la imagen original y la imagen transformada se pueden utilizar diferentes valores. Uno de estos es el error total, que es la suma de todas las diferencias entre los píxeles originales y sus píxeles correspondientes en la imagen modificada. También se puede utilizar el promedio de los mínimos cuadrados.

Lo ideal sería que este error de alguna manera se basara en la percepción, ya que al final es el ojo humano el que juzga la calidad de los bordados.

### A. Error del promedio de los cuadrados (MSE)

Es una de las mediciones más utilizadas para comparación de imágenes. Este error lo que hace es calcular el promedio de las diferencias entre los datos de la imagen inicial y aquellos de la imagen ya transformada.

Esta medida de control se utiliza para probar calidad, también se conoce como promedio de la desviación de los cuadrados o *mean square deviation*.

$$\text{MSE} = \frac{1}{m} \sum_{i=1}^m (x_i - T)^2$$

En donde  $x_i$  es el valor  $i$  del grupo de valores, y  $T$  es el valor deseado.

El utilizar este error tiene algunas desventajas: dado que el MSE depende de la unidad de medida que se está utilizando, no puede ser promediado de una forma significativa, para poder compararlos con otros errores MSE. Por ejemplo, para comparar imágenes, la medida está dada sobre los canales de color RGB, entonces la medida del error está relacionada con las diferencias RGB, donde el máximo puede ser de 255 al cuadrado. Si se quiere que sea una medida de error más general se debe de normalizar.

En la aplicación desarrollada, para imágenes del mismo tamaño, el MSE se calculó sumando los cuadrados de las diferencias entre el píxel en la imagen original y su correspondiente en la imagen transformada.

Pero, dado que la mayoría de imágenes también había pasado por un proceso de transformación de tamaño, en éstas se calculó el promedio general de la imagen. Y luego se calcularon las diferencias de cada píxel de la imagen transformada con el promedio.

## B. Normal de los promedios de los cuadrados (NMSE)

El error normalizado (NMSE) puede ser utilizado para comparar imágenes. En general, mientras más se acerca el valor del error a 0, la imagen transformada se parece más a la imagen fuente. Este error toma en cuenta cada píxel en la imagen. Normaliza el error en el rango entre 0 y 1. La ventaja de este error es que es independiente del rango de valores RGB de la imagen.

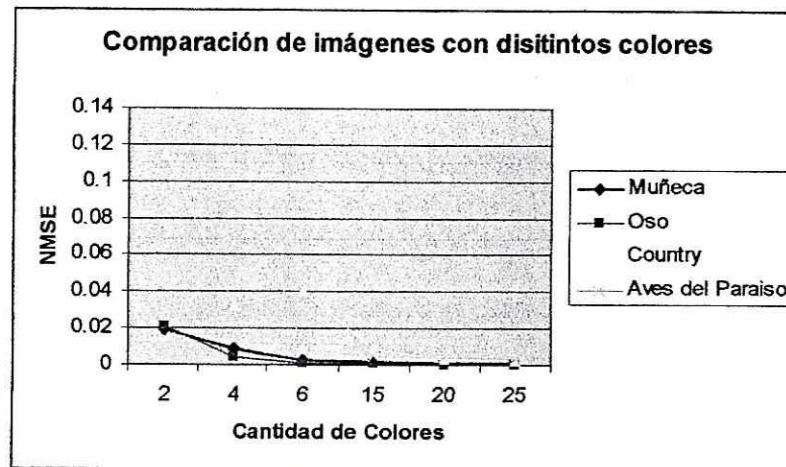


Fig 8.1 Gráfica de comparación del NMSE para imágenes con distinta cantidad de colores

De esta gráfica se puede inferir que en una misma imagen, a medida que se le van agregando más colores, el error al compararla con la imagen original va siendo menor. Esto es cierto cuando los colores que se están agregando a la paleta están entre los más utilizados, al comparar la imagen original contra todo el listado de colores DMC.

También se utilizó el NMSE para comparar una misma imagen bajo los diferentes algoritmos.

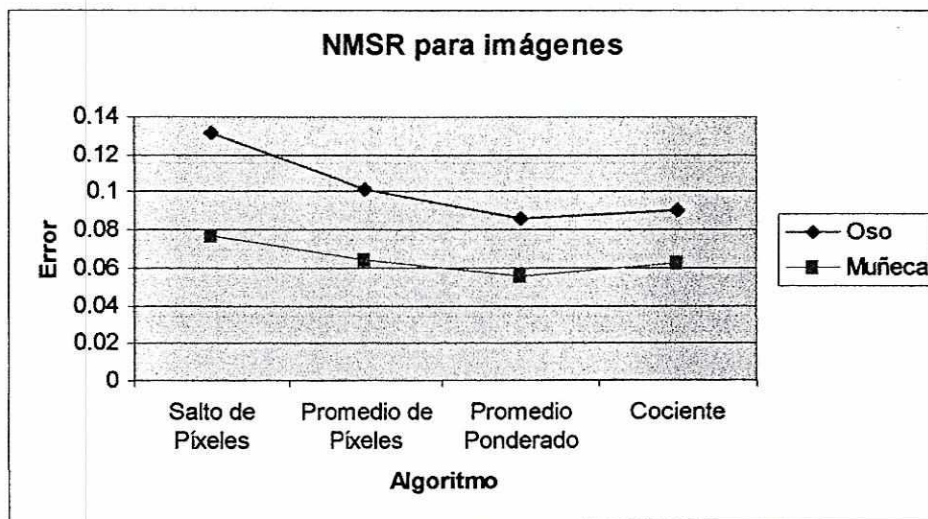


Fig 8.2 Comparación del NMSE para imágenes utilizando distintos algoritmos

De esta se deduce que el mejor algoritmo, en el cual la imagen termina más parecida a la imagen original, es la reducción de tamaño que utilizó promedios ponderados, seguida por el algoritmo del cociente. Esto se debe a que ambos algoritmos tratan de que el promedio se mejore y de hacer más pareja la imagen.

### C. Percepción visual

Si bien las medidas numéricas, como los datos vistos arriba, logran crear una relación de la imagen transformada con relación a la original y calcular la imagen más similar, ésta imagen no es necesariamente equivalente a la imagen que es más agradable para los humanos.

En el bordado de cruceta uno de los factores más importantes es que a las personas que lo van a observar y a bordar les guste la imagen. Por eso, éste factor se tomó en cuenta durante este proceso. Se crearon una serie de imágenes con las diferentes combinaciones de transformación de color y de tamaño, y se dieron a diferentes personas que bordan cruceta, para que emitieran una opinión sobre cuál era para ellos la mejor imagen (Apéndice B).

A partir de las respuestas obtenidas por ellos se obtuvieron los siguientes resultados:

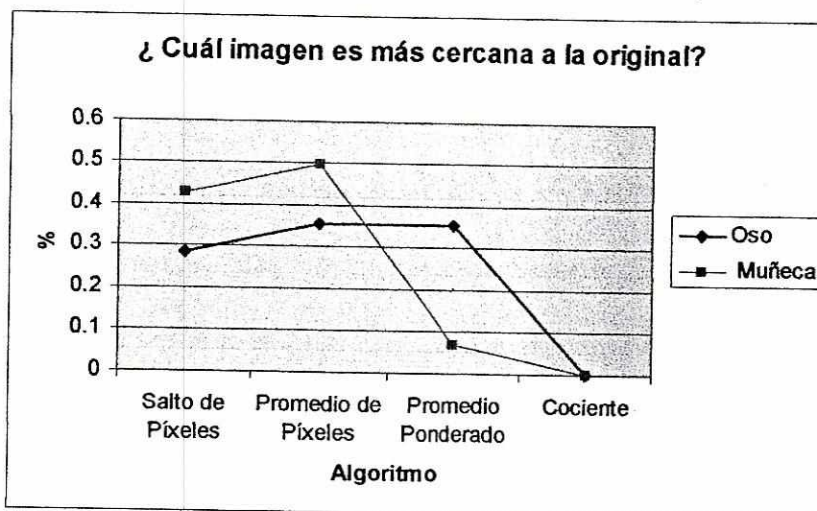


Fig 8.3 Comparación de imágenes más cercanas a la original

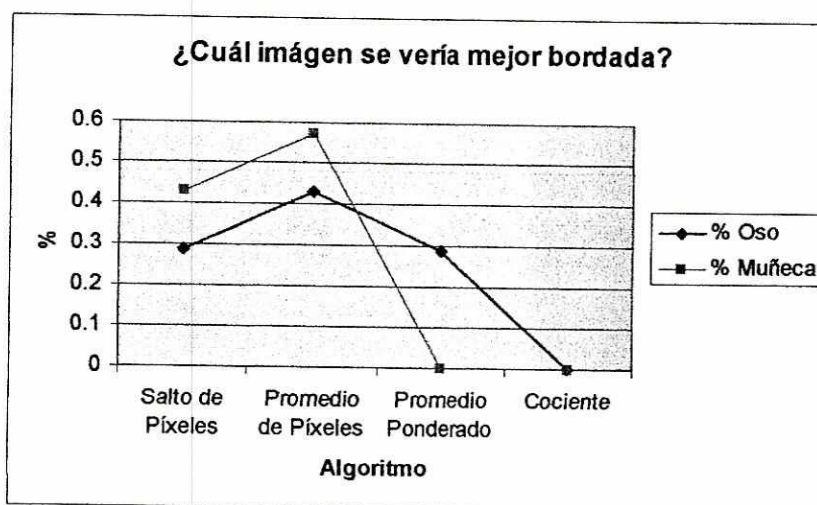


Fig 8.4 Comparación de imágenes que se verían mejor bordadas

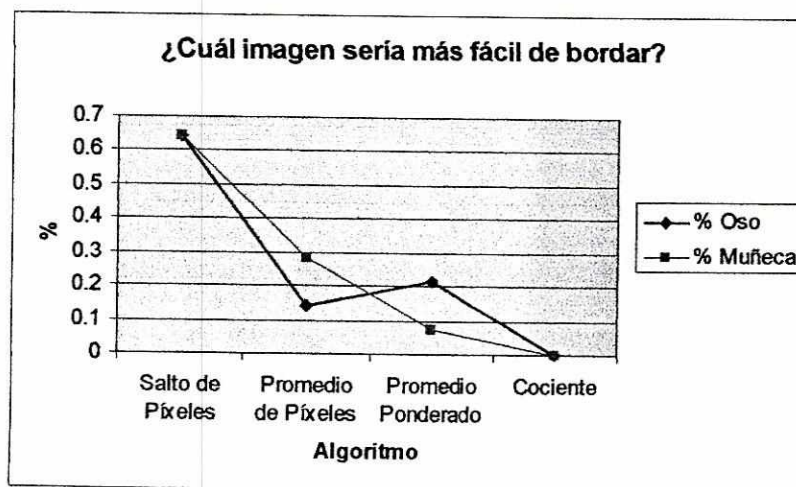


Fig 8.4 Comparación de imágenes más fáciles de bordar

En las gráficas se observa que aunque la imagen sea más parecida a la original, eso no las hace mejores a la vista de los humanos. Por ejemplo, el algoritmo de salto de píxeles está mejor calificado que el algoritmo de promedio ponderado, que es el que tiene en la realidad la imagen más cercana, a pesar de que las imágenes de salto de píxeles que se calificaron presentan muchas deformaciones obvias.

Las imágenes que mejor se ven son las que utilizan el algoritmo de promedio de píxeles; y además es el que las personas que bordan cruceta piensan que se varían mejor en cruceta. Pero las más fáciles de bordar son las que tienen saltos de píxeles, ya que los colores no son tan parejos, y eso permite al bordar ver las diferencias de colores y guiarse por ellos. También el promedio ponderado sería fácil de bordar, ya que por ser más pareja la imagen se utiliza más del mismo color.

Ni el algoritmo del cociente, ni el del cociente están entre los que se verían mejor bordados en cruceta.

## IX. INTERFASE CON MÁQUINAS BORDADORAS

Las máquinas bordadoras se utilizan en la industria para llevar a cabo trabajo repetitivo al por mayor. Un ejemplo de estas máquinas son las bordadoras Tajima, que son una de las marcas más conocidas internacionalmente. En la actualidad para poder bordar una imagen en máquinas Tajima, en la computadora, por medio de software especializado se asigna en qué áreas de la imagen se quiere que haya puntadas de cada tipo. Luego de seleccionar las áreas y el tipo de puntada, el programa genera un archivo que tiene el formato que utilizan como entrada las máquinas bordadoras.

Para poder desarrollar un bordado con crucetas utilizando estas máquinas, al momento de digitalizar la imagen, habría que colocar cada puntada de cada cruz independientemente, teniendo cuidado de que se mantenga la perfección de cada cruceta.

El trabajo de digitalizar la imagen se simplificaría si se pudiera hacer automáticamente, con sólo seleccionar la imagen. La aplicación desarrollada en este proyecto tiene como uno de los propósitos generar, a partir de las imágenes, patrones que se puedan interpretar en un futuro por medio de máquinas bordadoras. Para este propósito se generó a partir del patrón de símbolos un archivo XML que contiene toda la información importante del patrón. Se utilizó XML, ya que ese es un formato que se está utilizando para compartir información, y que se está convirtiendo en el estándar. Se espera que en el futuro cercano las máquinas bordadoras reciban como entrada XML. Luego, por medio de una transformada XSL, la información del patrón se transformó de una manera que tenga toda aquella información que es necesaria para las máquinas bordadoras Tajima.

### A. Formato XML

El *eXtensible Markup Language* (XML) es un método universal para representar información que está diseñado para ser transportado por *Internet*. El formato XML es un método que busca separar los datos de la presentación, y mostrarlos de una manera estándar que describa un lenguaje fácil de entender para las computadoras y para los humanos. El *World Wide Web Consortium* (W3C) fue quién creó una recomendación oficial del XML.

XML busca ser el lenguaje que las compañías utilizarán para intercambio de datos. La idea es que de una manera fácil se pueda enviar información de una base de datos de una compañía a los clientes, o a otras compañías, a través del browser.

La palabra *Markup*, se refiere a las instrucciones que pueden decirle a la aplicación qué hacer con la información.

También XML, como lo implica su nombre, tiene una estructura que se puede extender, esto quiere decir que se pueden ir creando etiquetas diferentes a medida que se está trabajando con este formato. A diferencia del *Hypertext Markup Language* (HTML) sus etiquetas ya están establecidas, en cambio en el XML no se quiere limitar a cierto número de etiquetas, y es difícil pensar en un grupo de etiquetas que pudieran describir todos los diferentes tipos de información, por eso se hace más fácil elaborar un sistema para crear etiquetas.

La idea del XML es separar los datos de la presentación, así una vez que se conoce la estructura de los datos, ya se pueden llevar a cabo más cosas con ellos. Una buena analogía para comparar la funcionalidad de XML y el HTML es compararlos como que el XML fuera el plano, mientras que el HTML es la foto. Con la foto no se pueden hacer muchos cambios, mientras que con el plano se pueden generar nuevas vistas, hacer nuevas modificaciones, etc.

El XML es una forma de expresión que ha sido estandarizada por reglas gramaticales, como cualquier otro lenguaje.

#### 1. Documentos XML. Algunas de las características de XML son las siguientes:

- Los documentos XML son sensitivos a la letra (mayúsculas y minúsculas)
- Cada etiqueta que se abre se debe cerrar. Ej : <Nombre></Nombre>
- No pueden haber etiquetas que se traslapen. Esta regla funciona como la regla de los paréntesis, no se puede concluir una etiqueta sin haber concluido primero todas las etiquetas empezadas luego de esta.
- Sí se pueden utilizar etiquetas anidadas. Y se anima a utilizarlas desde una perspectiva de diseño elegante.

Un documento XML, puede contener las siguientes marcas:

- Inicios de elementos y fin de elementos: Son los bloques de construcción del XML, estos pueden contener la información o definir la estructura. Cada elemento inicia con una etiqueta de inicio, y termina con una etiqueta de finalización. Las etiquetas de inicio no pueden tener más de una palabra, y van rodeadas por los caracteres <>. La única diferencia entre la etiqueta de inicio y la de finalización, es que esta última, luego del < se sigue por un /.

Los elementos se pueden utilizar para definir jerarquía, esto se hace creando etiquetas anidadas. Ejemplo:

```

<CONTACTO>
  <NOMBRE>Pedro Picapiedra</NOMBRE>
  <TELEFONO>
    <AREA>502</AREA>
    <NUMERO>6349006</NUMERO>
  </TELEFONO>
</CONTACTO>

```

- Atributos: Estos son información utilizada para describir elementos. Estos atributos se encuentran dentro de la etiqueta de inicio, luego del nombre del elemento.

Ejemplo: `<tabla borde="5" tamaño="2">`

- Comentarios: Estos se utilizan para escribir recordatorios para el programador, no para ser desplegados. Estos se escriben de igual manera que en HTML.

Ejemplo: `<!-- Este es un comentario -->`

- Referencias a entidades: Estos son punteros a entidades. En XML las entidades son unidades de texto que pueden ser desde un carácter hasta documentos completos. Esto hace que no sea necesario integrar todo el XML en un solo documento, sino que se puedan integrar varios. Esto también se utiliza cuando se tiene que escribir caracteres que podrían confundir al parser, entonces se hacen referencias a entidades.

Ejemplo: `<comparación> 3 < 5 </comparación>` se cambia a `<comparación> 3 &lt 5 </comparación>`, ya que en la primera opción el carácter `<` confundiría al parser.

- Instrucciones de procesamiento

Esta información es específica para la aplicación o su procesador. Para el parser por ejemplo. Se escribe dentro de las cadenas `<? y ?>`.

Ejemplo: `<? xml version="1.0" ?>`

- Secciones de datos (CDATA) : Esta es una manera de burlar al parser cuando aparecen caracteres que lo confundirían, lo que se hace es envolver los caracteres donde se daría un error de parser dentro de un CDATA. Ejemplo: `<comparación> <![CDATA[3<5]]> </comparación>`

- Document Type Definition (DTD): Define la estructura lógica del documento.

a. Document Type Definition (DTD). Los DTD son una serie de expresiones o reglas que definen la estructura lógica de un documento de XML. Esta es una parte opcional de un sistema

XML. Lo que se hace es crear un archivo con las reglas y luego asociarlo al documento que debe cumplirlas.

La ventaja de utilizar DTD es que grupos independientes de personas pueden acordar utilizar un DTD para intercambiar datos. Así mismo se pueden utilizar DTD para verificar que los XML recibidos son válidos.

Ejemplos de cómo se definen DTD:

DTD externa:

```
<!DOCTYPE archivo_que_debe_cumplirlas SYSTEM "reglas.dtd">
```

DTD interna:

```
<!DOCTYPE root_element
[<!-- Definiciones internas van acá -->
]
```

Declaración de elementos:

En un DTD se declaran los elementos de la siguiente forma:

```
<!ELEMENT nombre_del_elemento categoría>
ó
<!ELEMENT nombre_del_elemento (contenido_del_elemento)>
```

Elementos vacíos: Los elementos vacíos se declaran con la palabra EMPTY:

```
<!ELEMENT nombre de elemento EMPTY>
```

Elementos con datos de caracteres: Los elementos que son solamente caracteres se declaran con #PCDATA dentro de paréntesis.

```
<!ELEMENT nombre elemento (#PCDATA)>
```

Elementos con cualquier contenido: Se declaran con la palabra clave: ANY

```
<!ELEMENT nombre del elemento ANY>
```

Elementos con hijos: Los elementos que tienen uno o más hijos se definen colocando el nombre de los hijos dentro de paréntesis:

Elementos que se repiten una sola vez:

```
<!ELEMENT nombre del elemento (hijo con una sola repetición)>
```

Elementos que se repiten al menos una vez:

```
<!ELEMENT nombre elemento (hijo +)>
```

Elementos que se pueden repetir cero o más veces:

```
<!ELEMENT note (message*)>
```

Elementos que se repiten cero o una vez:

```
<!ELEMENT nombre del elemento (hijo?)>
```

Elementos que pueden darse o uno o el otro:

```
<!ELEMENT nombre del elemento ((elemento1 | elemento2))>
```

Declaración de atributos:

La declaración de atributos tiene la siguiente sintaxis:

```
<!ATTLIST nombre_del_elemento nombre_de_atributo tipo
valor default>>
```

Donde el tipo puede ser:

Valor	Explicación
CDATA	El valor son datos de caracteres
(en1 en2  en3)	El valor debe ser uno de la lista
ID	El valor tiene una ID única
IDREF	El valor es el ID de otro elemento
IDREFS	El valor es una lista de otros ids
NMTOKEN	El valor es un nombre XML valido
NMTOKENS	El valor es una lista de nombres XML
ENTITY	El valor es una entidad
ENTITIES	El valor es una lista de entidades
NOTATION	El valor es el nombre de una notación
Xml:	El valor es un valor de xml predefinido

Xml:	El valor es un valor de xml predefinido
------	---

Tabla 9.1 Valores del atributo en XML

El valor puede tener los siguientes valores:

Valor	Explicación
Valor	El valor default del atributo
#REQUIRED	Indica que el atributo debe ser incluido en el elemento
#IMPLIED	El atributo no tiene que ser incluido
#FIXED valor	El valor del atributo es fijo

Tabla 9.2 Tipos del atributo en XML.

2. DTD del patrón. A partir de la imagen ya transformada, y de la generación del patrón de símbolos, la aplicación desarrollada puede generar un archivo XML, el cual tiene toda la información importante del patrón. Este archivo incluye la información de cada color utilizado en el patrón (su número DMC, su nombre, y los colores RGB por los que está formado). Y además incluye la información de cada cruceta de color, para ellas se especifica sus coordenadas X, Y, el número de color de la cruceta, y el símbolo.

También incluye dos parámetros más, que son el total de saltos y de cambios de color que se realizarían si la imagen se recorriera de izquierda a derecha, y de arriba hacia abajo.

El DTD del archivo generado es el siguiente:

```
<!DOCTYPE PATRON [
<ELEMENT PATRON
(NOMBRE,COLORES,MAXX,MAXY,PUNTADAS,TOTAL_SALTOS,TOTAL_CAMBIOS_DE_CO
LOR)>
<ELEMENT NOMBRE ANY>
<ELEMENT COLORES (COLOR)*>
<ELEMENT COLOR EMPTY>
<ATTLIST COLOR
    DMC CDATA #REQUIRED
    NOMBRE CDATA #REQUIRED
    SIMBOLO CDATA #REQUIRED
    RED CDATA #REQUIRED
    GREEN CDATA #REQUIRED
    BLUE CDATA #REQUIRED >
<ELEMENT MAXX ANY>
<ELEMENT MAXY ANY>
<ELEMENT PUNTADAS (CRUZ)*>
<ATTLIST PUNTADAS
    NUM_COLOR CDATA #REQUIRED
    SIMBOLO CDATA #REQUIRED
    X CDATA #REQUIRED
    Y CDATA #REQUIRED >
<ELEMENT TOTAL_SALTOS ANY>
<ELEMENT TOTAL_CAMBIOS_DE_COLOR ANY>]>
```

## B. Lenguaje XSL

XSL es un lenguaje derivado de XML para transformar documentos desde un vocabulario XML a otro XML. XSL es una recomendación para definir la transformación de un documento XML y su presentación. Las hojas de estilo XSL son documentos XML con sus propias etiquetas.

### 1. Componentes del XSL

a. XML Path Language (XPath): Es un lenguaje de expresiones utilizadas por XSLT para acceder o referirse a partes de un documento XML. Este se utiliza dentro de las transformaciones para seleccionar los elementos que se van a transformar, procesar los elementos condicionalmente y para escribir expresiones a fin de generar valores de texto.

Algunos ejemplos de cómo se utiliza el XPATH son:

Accesa a nodo raiz  
`<xsl:apply-templates select="/" />`

Accesa cada nodo dentro de las etiquetas PUNTADAS que están dentro de etiqueta PATRON  
`<xsl:apply-templates select="PATRON/PUNTADAS" />`

b. XSL Formatting Objects (XSL-FO): Es un vocabulario XML para especificar la semántica.

c. XML Stylesheet Transformation Language (XSLT): XSL utiliza XSLT para el proceso de transformación de los objetos, para que cumplan con el formato del dispositivo final. XSLT contiene reglas dadas por el autor del documento para transformar el XML original, o árbol fuente, en un árbol de resultado.

Cada regla del XSL se compone de un patrón (*pattern*) y una acción o plantilla (*template*). Se hace corresponder los elementos del árbol fuente según un patrón, y luego se genera el resultado colocando los elementos del original en una plantilla especificada.

La forma de programar en XSL es diferente a los lenguajes imperativos que son los más utilizados, es más parecido a los lenguajes funcionales. XSL le dice al procesador qué hacer, no cómo llevarlo a cabo. Una de las características es que en XSL una instrucción debe hacer lo mismo sin importar el camino que se utilice para llegar a ella. Esto elimina que haya efectos secundarios. Además en XSLT no existen variables a las que una vez asignado un valor se les pueda cambiar nombre.

En XSL y XML se refieren mucho a los árboles, en XSL el documento original se traduce a un árbol, donde existen una raíz (que es el elemento raíz) y los hijos de cada nodo que son los elementos contenidos en él.

## 2. Modelos de uso del XSL

- Modelo por plantillas: Si el XML que se recibe como origen consiste de muchas estructuras repetitivas que se quiere que tengan el mismo formato, entonces se utiliza este modelo.
- Modelo por datos: A diferencia del modelo por plantillas, si la información a recibir consiste en datos irregulares, entonces se utiliza este modelo. Este separa cada segmento de información para tratarla independientemente.

Se pueden combinar ambos modelos cuando hay ciertos segmentos de información que quieren tratarse por separado, y otros que son repetitivos.

3. Procesamiento de hojas de estilo. Un procesador de XSL acepta documentos o datos en XML y una hoja de estilo XSL, y produce lo que el diseñador esperaba presentar a partir del árbol origen en XML. Para poder llegar a presentar el contenido se deben tomar en cuenta dos partes: la primera que es construir un árbol resultante a partir de la interpretación del árbol origen, y la otra parte es producirlos en una forma que sea conveniente para la presentación, ya sea de despliegue u otro medio, por ejemplo el formato que reciben las máquinas bordadoras.

El primer aspecto se llama transformación de árbol y el segundo aspecto es el formateo. La segunda parte puede llevarla a cabo una aplicación encargada de formatear, o puede ser algo que ya esté dentro del browser, o máquina que recibe el XML.

Las transformaciones de árbol permiten que la estructura del árbol origen sea totalmente diferente a la estructura del árbol resultante. En esta transformación también se agrega la información necesaria para darle formato al árbol resultante. Esta información se logra al incluir objetos de formateo, logrando así que los nodos del árbol resultante sean nodos de formateo. En XSL son los objetos de formateo y sus propiedades las que permiten llevar a cabo esto.

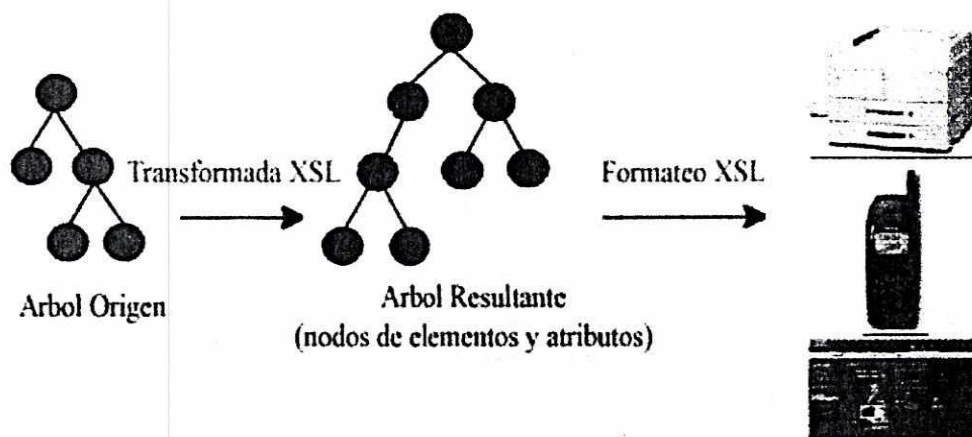


Fig 9.1 Procesamiento de XML en 2 partes: Transformada XSL y Formateo XSL

## C. FORMATO .DST

1. Máquinas de bordar marca Tajima. La empresa Tajima se dedicada a la producción de máquinas bordadoras. Sus máquinas juegan un papel importante en el desarrollo de bordados a nivel internacional, éstas se exportan a más de 100 países en el mundo.

Estas máquinas tienen varias cabezas, cada una de ellas puede tener de 6 a 15 agujas, en cada una de las cuales se coloca un hilo de color diferente.

Las máquinas Tajima utilizan un formato .dst

### 2. Gramática del formato DST

a. El encabezado: Los archivos .dst tienen un encabezado con información sobre el diseño. El encabezado es de un largo de 125 bytes, y está relleno de caracteres 0x20, para llegar a un total de 512 bytes.

Todas las entradas del encabezado están formadas por dos caracteres ASCII seguidas del signo de dos puntos. Luego, está seguido del valor y un cambio de línea.

La primera entrada es 'LA' que es el nombre del diseño, este no tiene información del directorio en que se encuentra el archivo o la extensión de éste. El nombre puede tener un máximo de ocho caracteres. Pero está en un espacio de 16 caracteres en total. Los que no utiliza el nombre quedan en blanco.

La siguiente entrada es 'ST', ese es el total de puntadas del bordado. Este es un número de siete dígitos relleno con ceros que anteceden al número. Este número incluye los cambios de color, los saltos, puntadas, y cualquier registro especial.

Luego la entrada 'CO' o colores, es un número de tres dígitos también relleno con ceros que anteceden al número. Este número es la cantidad de cambios de color que se dan en el archivo.

El siguiente es '+X', esta entrada es la distancia máxima, en centímetros, que el bordado se va a mover de la posición inicial hacia el X positivo. Este es un número de cinco dígitos relleno con ceros también.

Luego la entrada '-X', es la distancia máxima que el bordado se va a mover en X negativo. Y así mismo las siguientes dos entradas, '+Y' y '-Y' que son en el eje y. Todas son de cinco dígitos también.

Después las entradas 'AX' y 'AY' expresan las coordenadas relativas desde el último punto al de inicio expresado en décimas de milímetro. Si el punto de finalización es el mismo que el punto de inicio 'AX' y 'AY' deberían ser (0,0).

'MX' y 'MY' se utilizan para expresar las coordenadas del último punto del archivo previo. Esto se utiliza para diseños multivolumen. Cuando es un solo archivo, estas entradas deben ser '+ 0'.

Y la última entrada del encabezado 'PD' también almacena información para diseños multivolumen. En diseños de un solo archivo se coloca \*\*\*\*\*.

b. Registros de puntadas. Cada registro consiste en tres bytes, cada uno de los cuales contiene información. Los bits más altos de cada byte representan la coordenada en Y, y los bits bajos corresponden a X. El último byte, además de tener información de X y Y contiene información que define la función de ese registro.

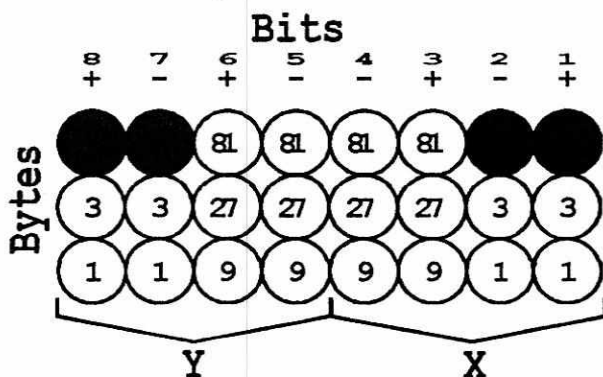


Fig 9.2 Bytes representantes de cada registro de puntada del formato .dst

Códigos de las funciones: Los códigos de las funciones se encuentran en el último byte. Los bits uno y dos siempre están encendidos (en 1). Con excepción de el registro que indica el final del diseño, ya que en ese los bytes uno y dos se ponen en 0. Los códigos son los siguientes:

- Puntada Normal	00000011	0x03
- Salto	10000011	0x83
- Parada/Cambio de Color	11000011	0xC3
- Fin del Diseño	11110011	0xF3

Cada registro de tres bytes puede ser decodificado de la siguiente manera:

Byte nnn+0:	Byte nnn+1:	Byte nnn+2:
Bit 7 : y +1	Bit 7 : y + 3	Bit 7 : salto
Bit 6 : y - 1	Bit 6 : y - 3	Bit 6 : cambio de color
Bit 5 : y +9	Bit 5 : y +27	Bit 5 : y + 81
Bit 4 : y - 9	Bit 4 : y - 27	Bit 4 : y - 81
Bit 3 : x - 9	Bit 3 : x - 27	Bit 3 : x - 81
Bit 2 : x +9	Bit 2 : x + 27	Bit 2 : x + 81
Bit 1 : x - 1	Bit 1 : x - 3	Bit 1 : ] siempre es 1
Bit 0 : x +1	Bit 0 : x + 3	Bit 0 : ] siempre es 1

Por ejemplo, si se tienen los siguientes tres bytes: 10101000 00010010 00100011, se entiende de la siguiente forma. Es una puntada normal, ya que tiene es la única función de código que se cumple en el tercer byte. Y esta puntada se moverá 64 en eje Y y -12 en el eje X.

3. XSL para transformar al formato .DST. Ya que se entendieron los datos por los que están compuestos los archivos con formato .dst, se puede explicar las conversiones que utilizó el XSL para convertir el XML, que contiene toda la información del patrón, a otro XML, que cuenta con la información requerida por las máquinas bordadoras Tajima.

Los archivos .dst no almacenan la información por crucetas, sino por puntadas simples. Por eso cada una de las cruces del formato original se tradujo a cuatro puntadas (1-2, 2-3, 3-4 y 4-5) en el formato XML para las máquinas Tajima.

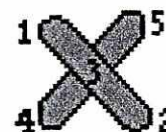


Fig 9.3 Manera de bordar cada cruz con puntadas simples

Se terminó en la posición 5, que es donde idealmente se iniciaría la puntada siguiente.

También se verificó cuándo hay cambios de color entre una cruz y la siguiente. Si existía un cambio de color se agregó un registro de cambio de color al nuevo formato XML. Además, si la posición final de una cruz era diferente a la posición inicial de la siguiente se agregaba un registro de salto.

El archivo final que se generó contiene la información que podría utilizar una bordadora Tajima. No está formateado exactamente de la manera que estaría en un archivo .dst, ya que los XML tienen como propósito separar los datos de la presentación. Así que contiene la información que podría utilizar, pero no en ese formato.

### DTD DE XML CON PARÁMETROS PARA .DST

```

<!DOCTYPE DST [<!ELEMENT DST (LA,CO,ST,PX, NX, PY, NY,AX, AY, MX, MY, PD, JUMP*,
CAMBIOCOLOR*, PUNTADA*) >
<!ELEMENT LA #PCDATA>
<!ELEMENT CO #PCDATA >
<!ELEMENT ST #PCDATA >
<!ELEMENT PX #PCDATA >
<!ELEMENT NX #PCDATA >
<!ELEMENT PY #PCDATA >
<!ELEMENT NY #PCDATA >
<!ELEMENT AX #PCDATA >
<!ELEMENT AY #PCDATA >
<!ELEMENT MX #PCDATA >
<!ELEMENT MY #PCDATA >
<!ELEMENT PD #PCDATA >
<!ELEMENT JUMP EMPTY>
<!ELEMENT CAMBIOCOLOR EMPTY>
<!ELEMENT PUNTADA EMPTY>
<!ATTLIST PUNTADA
      X CDATA #REQUIRED
      Y CDATA #REQUIRED >
<!ATTLIST CAMBIOCOLOR
      X CDATA #REQUIRED
      Y CDATA #REQUIRED >
<!ATTLIST JUMP
      X CDATA #REQUIRED
      Y CDATA #REQUIRED >
]>

```

## X. CONCLUSIONES Y RECOMENDACIONES

En el área de bordados juega un papel muy importante la percepción visual. Por eso, aunque la imagen transformada con menor error fue la que utilizó el algoritmo de promedio ponderado para la reducción de tamaño, este algoritmo no es el que se recomienda para los patrones de bordado de cruceta. Se recomienda el algoritmo de promedio de píxeles simple, ya que éste es el que causa mejor impresión visual, manteniendo diferencia entre píxeles, pero sin distorsionar la imagen final.

Mientras mayor sea la escala a la que se reduzca la imagen, mayor es la cantidad de colores importantes de la imagen que se pierden. Es mejor conseguir imágenes de baja resolución, para que así al tener que reducirles el tamaño no se pierda tanta información.

Es difícil obtener una tabla de conversión exacta de los colores de los hilos DMC a sus respectivos RGB y eso afecta los patrones. Ya que aunque en los cálculos y en la pantalla resulten agradables a la vista, ya al trasladarlos al bordado real los colores y sus combinaciones resultan diferentes.

Este proyecto logró obtener los resultados deseados, generar patrones de bordado de cruceta que sean entendibles al usuario.

Además por medio de este proyecto se motivó el estudio de algoritmos de manipulación de imágenes, que pueden ser de utilidad en muchas áreas, no solo en la generación de patrones de bordado de cruceta.

A partir de este proyecto, surgieron otras dos áreas que podrían profundizarse en futuros trabajos de graduación:

- La primera es el área de bordes; en esta debe calcular cuáles son los contornos importantes que deberían agregársele a los patrones, ya que los bordes, aunque no son puntadas de cruceta, juegan un papel importante para realzar las partes más importantes del bordado.
- La otra área es optimizar la transición del XML del patrón al XML de las máquinas bordadoras. Ya que bordar las cruces de derecha a izquierda y de arriba abajo no es la mejor solución. Se debería encontrar una manera de minimizar los cortes de hilo y los saltos.

## XI. APÉNDICES

### A. Glosario de términos

**Aída:** (Tela Aída) Tela de algodón creada especialmente para el punto de cruz o cruceta. Su trama regular delimita cada uno de los puntos. Existen diferentes tramas: 8, 11, 14, 16 y 18. Estas cifras expresan el count, que es el número de puntos por pulgada o. La más utilizada es de 14 counts.

**Algoritmo:** Procedimiento o fórmula para resolver un problema.

**Atributo:** Propiedades de los elementos.

**Backstitch o pespunte:** Es una puntada en la que el inicio de esa puntada se traslapa con el final de la anterior. Esta se puede utilizar para trazar líneas o bordes.

**Bit:** Unidad de medida de información, esta información puede tener dos estados 1 o 0.

**Bitmap:** Es un archivo o estructura de imagen que representa bit por bit la imagen que se despliega en el monitor. Los bitmaps pueden representar también gráficas de color, pero para eso se necesita más de un bit para representar cada canal de color.

**Bit depth:** El número de bits utilizados para representar cada píxel en la imagen, determinando su rango de color o tonalidades.

**Browser:** Un programa que permite al usuario encontrar e interactuar con el material de Internet.

**Convolución:** Es una función matemática sencilla que es el fundamento en muchos operadores de procesamiento de imágenes. Provee una manera de multiplicar dos arreglos de números, generalmente de tamaños distintos pero con la misma dimensionalidad, para producir un tercer arreglo de números de la misma dimensión.

**Contorno:** Los bordes o contornos se pueden definir como el cambio significativo de tonos de grises en una imagen.

**Count:** Puntos por pulgada. En cruceta se refiere a cuántas cruces se pueden hacer por pulgada de tela.

**Cruceta:** Es un tipo de puntada formada por dos puntadas simples, la cual forma una x. Esta tradicionalmente se hace de la esquina inferior izquierda hacia la esquina superior derecha, y luego de la esquina inferior derecha a la superior izquierda.

**CMY:** Acrónimo para Cyan, Magenta, and Yellow (cyan, magenta y amarillo) - los colores se substraen para formar el color.

**Dithering:** Es el proceso para simular colores que no están disponibles en una gráfica. Esto se logra utilizando patrones de dos o más colores de los que se tienen disponibles.

**DMC:** Una marca de hilos para bordar. Tienen un tipo de hilo utilizado para bordar cruceta. Estos tienen una gama de color bastante extensa. Al referirse a colores DMC se está hablando de los colores de hilo que DMC tiene disponibles.

**DTD:** Document Type Definition- Es un archivo que define los elementos y la estructura de un archivo XML.

**Elemento:** Una de las partes que componen una entidad. Cada uno de los objetos que forman los archivos XML.

**Espacio de color:** Un método o lenguaje particular para representar colores.

**Etiqueta:** Una etiqueta es un comando de formateo el cual describe cómo las páginas se desplegarán por el browser. Cada etiqueta consiste en un comando descriptivo que está contenido entre los signos "<" y ">".

**Gradiente:** La pendiente con la que aumenta o disminuye algo. El cambio de algún valor por unidad de distancia.

**Laplace:** El operador de Laplace encuentra los bordes en las imágenes, buscando en dónde se encuentran los ceros de la segunda derivada de la imagen.

**Lenguaje funcional o declarativo:** Aquellos que están compuesto de funciones. No tienen efectos colaterales, es decir, que sus parámetros no pueden cambiar mientras se ejecuta. Los resultados a obtener pueden verificarse.

- Mean square error:** Este error está dado por el cuadrado de todos los errores elevados al cuadrado. Se utiliza para calcular varianzas.
- Normalizar:** Un grupo de datos numéricos es normalizado al sustraer a todos los valores el valor mínimo y dividirlos por el rango de los datos. Esto logra hacer que todos los números queden en el rango entre 0 y 1.
- Paleta:** Es un rango de colores característico de algo o alguien. Un subconjunto de colores que puede ser desplegado.
- Parser:** Una función que reconoce las frases válidas de un lenguaje, analizando su estructura sintáctica.
- Patrón:** Modelo que se utiliza para poder realizar algo.
- Píxel:** Se le llama así al componente más pequeño de una imagen al cual se le puede asignar un color.
- Raster:** (Imágenes Raster) También se conocen con este nombre a las imágenes bitmaps, que son las que están representadas por una secuencia de píxeles o puntos.
- RGB:** Acrónimo para Red, Green, Blue (Rojo, verde y azul), que es el modelo para describir los colores emitidos por la luz. Es el modelo aditivo básico para el despliegue de color.
- Semántica:** Parte de la estructura del lenguaje que se relaciona con el significado.
- Sintaxis:** El orden o arreglo gramatical de las palabras.
- Sobel:** El operador de Sobel realiza una medida del gradiente en 2 dimensiones, y enfatiza las regiones con gradiente que corresponde a los bordes. Se utiliza para encontrar la magnitud del gradiente absoluto de cada punto en una imagen en tonos de grises.
- Trama:** Las telas están formadas por hilos que se cruzan entre sí, unos que van horizontales y otros verticales. Se le llama trama a los hilos horizontales de la tela.
- Urdimbre:** Las telas están formadas por hilos que se cruzan entre sí, unos que van horizontales y otros verticales. Se le llama urdimbre a los hilos verticales de la tela.
- Vector:** (Imagen tipo vector) Esta imagen está formada por puntos y líneas que enlazan dichos puntos.

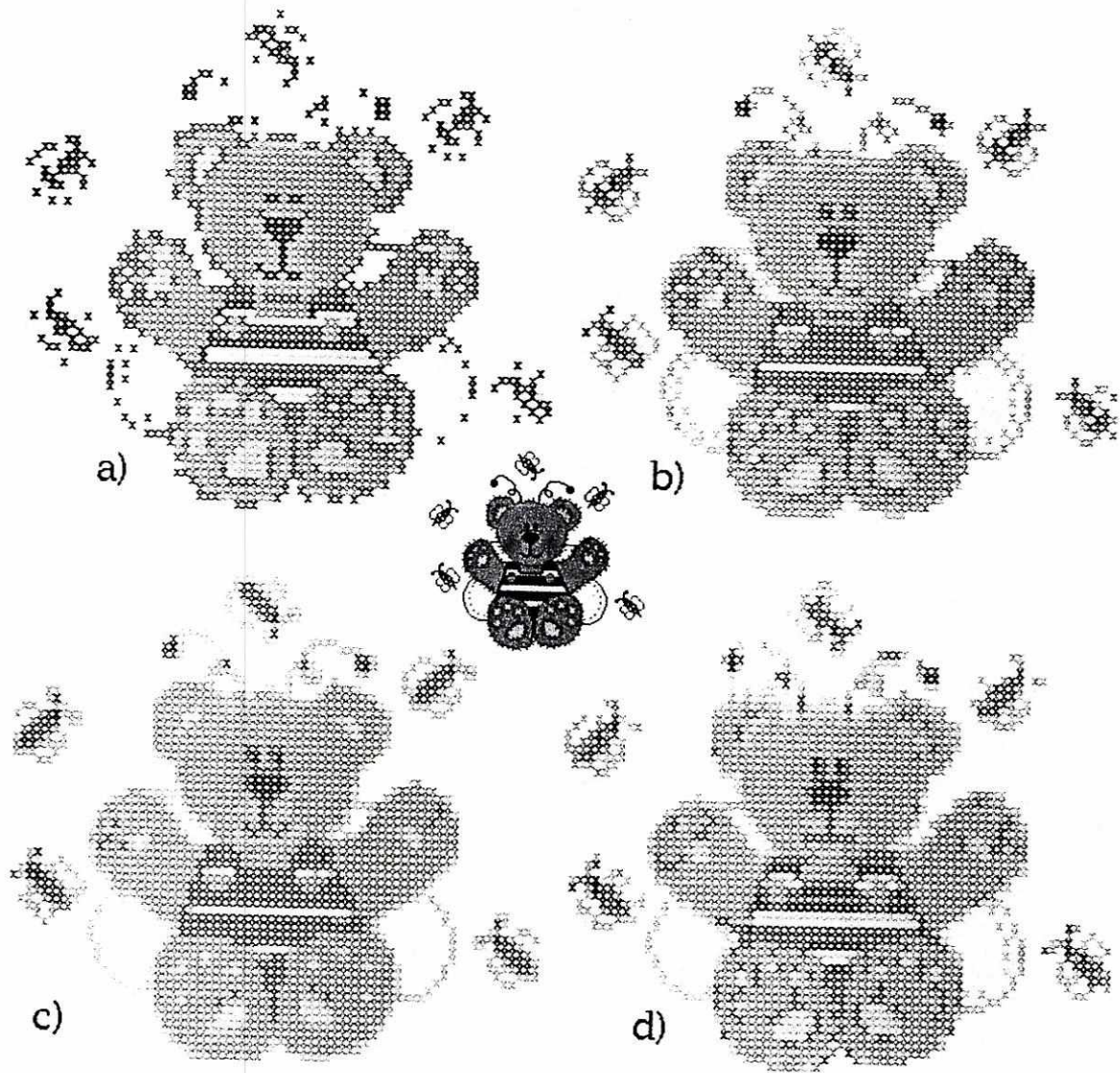
**XML:** Extensible Markup Language – Es una manera de crear formatos estándares. Este lenguaje permite que los diseñadores creen sus propias etiquetas personalizadas, permitiendo así la definición, transmisión y validación de datos entre aplicaciones y organizaciones.

**XPath:** XML Path Language -- Es un lenguaje de expresiones utilizadas por XSLT para acceder o referirse a partes de un documento.

**XSL:** Extensible StyleSheet Language, es un lenguaje que sirve para transformar documentos en XML a otro formato, que podría ser HTML, PDF, o aún otro XML.

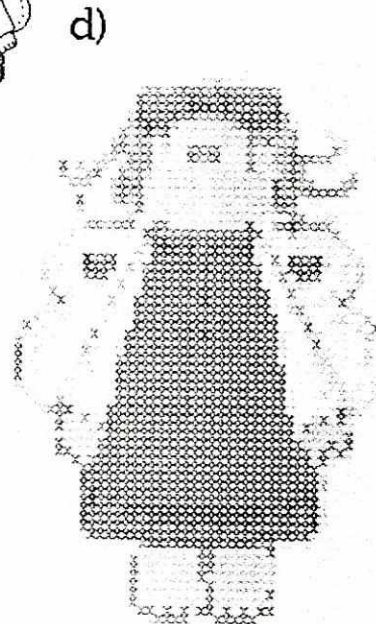
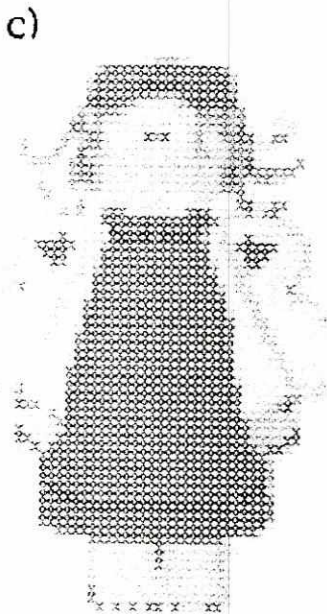
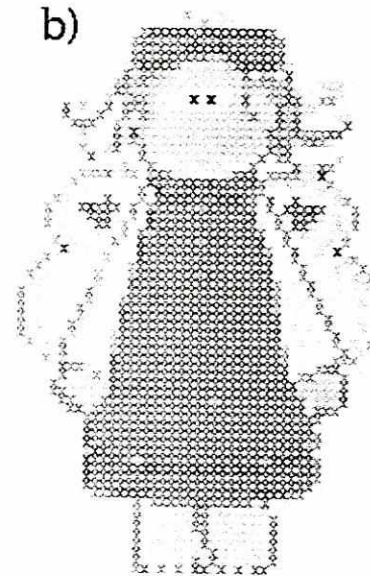
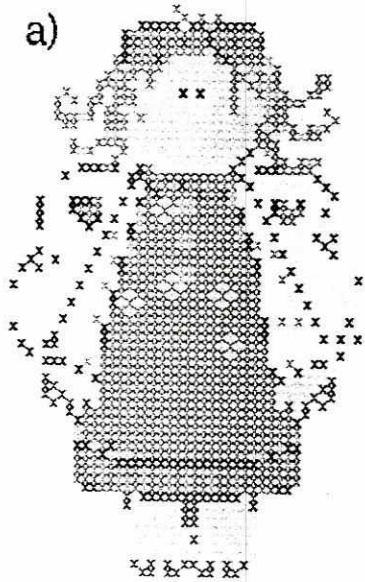
**XSLT:** Transformación XSL -- Es un lenguaje para transformar documentos XML en otros documentos XML. XSLT está diseñado para utilizarse como parte del XSL.

## B. ENCUESTA



1. ¿Cuál imagen se parece más a la original?
2. ¿Cuál imagen se vería mejor bordada?
3. ¿Cuál imagen sería más fácil de bordar?

1. ¿Cuál imagen se parece más a la original?
2. ¿Cuál imagen se vería mejor bordada?
3. ¿Cuál imagen sería más fácil de bordar?



## C. Selección de la muestra

### Términos en el estudio estadístico

**Población:** Conjunto de todos los individuos que son objeto del estudio. Se llama población al conjunto de todas las observaciones o medidas con una determinada característica que constituyen el objetivo de un estudio. Cada una de las observaciones de esta población se expresa normalmente en forma de números, y los individuos de la población son estos números.

**Muestra:** Parte de la población en la que miden las características estudiadas.

**Muestreo:** Proceso seguido para la extracción de una muestra.

**Encuesta:** Proceso de obtener información de la muestra.

**Parámetro:** Son las medidas o datos que se obtienen sobre la distribución de probabilidades de la población, tales como la media, la varianza, la proporción, etc.

**Estadístico:** Los datos o medidas que se obtienen sobre una muestra y por lo tanto una estimación de los parámetros.

**Error muestral:** De estimación o standard. Es la diferencia entre un estadístico y su parámetro correspondiente. Es una medida de la variabilidad de las estimaciones de muestras repetidas en torno al valor de la población, nos da una noción clara de hasta dónde y con qué probabilidad una estimación basada en una muestra se aleja del valor que se hubiera obtenido por medio de un censo completo. Siempre se comete un error, pero la naturaleza de la investigación nos indicará hasta qué medida podemos cometerlo (los resultados se someten a error muestral e intervalos de confianza que varían muestra a muestra). Varía según se calcule al principio o al final. Un estadístico será más preciso en cuanto y tanto su error es más pequeño. Podríamos decir que es la desviación de la distribución muestral de un estadístico y su fiabilidad. (Muestra y Tamaño de muestra, 2004)

### Métodos de muestreo

- Muestreo no probabilístico: no se usa el azar, sino el criterio del investigador.

- Muestreo probabilístico o aleatorio:

⇒ Muestreo aleatorio simple: se asigna un número a cada uno de los individuos de la población, y seguidamente se van eligiendo al azar los componentes de la muestra. La elección de un

individuo no debe afectar a la del siguiente, por tanto debe reemplazarse el n°, una vez extraído.

- ⇒ Muestreo sistemático: se ordenan previamente los individuos de la población, después se elige uno al azar y a continuación, a intervalos constantes, se eligen todos los demás hasta completar la muestra.
- ⇒ Muestreo estratificado: se divide la población total en clases homogéneas (estratos). La muestra se escoge aleatoriamente en número proporcional al de los componentes de cada estrato (Muestra y Tamaño de muestra, 2004).

### Tamaño de la muestra

Para determinar el tamaño de la muestra cuando los datos son cualitativos, es decir para el análisis de fenómenos sociales o cuando se quiere verificar la ausencia o presencia del fenómeno a estudiar, se recomienda la utilización de la siguiente fórmula:

$$n = \frac{n'}{1 + n'/N}$$

Siendo

$$n' = \frac{s^2}{\sigma^2}$$

donde:

$\sigma^2$  es la varianza de la población respecto a determinadas variables.

$s^2$  es la varianza de la muestra, la cual podrá determinarse en términos de probabilidad como  $s^2 = p(1-p)$

$se$  es error estándar que está dado por la diferencia entre  $(\mu - \bar{x})$  la media poblacional y la media muestral.

$(se)^2$  es el error estándar al cuadrado, que nos servirá para determinar  $\sigma^2$ ,

por lo que  $\sigma^2 = (se)^2$  es la varianza poblacional.

Para llevar a cabo este trabajo se utilizó como población, un grupo de discusión de bordado de cruceta de Internet que tiene 60 miembros que lo conforman. Se utilizó un error estándar de 0.1 al 75% de confiabilidad.

$$se^2 = (0,1)^2 = 0.01$$

$$s^2 = p(1-p) = 0.75(1-0.75) = 0.1875$$

$$n' = s^2 / se^2 = 0.1875 / 0.01 = 18.75$$

$$n = n' / (1 + n'/N) = 18.75 / (1 + 18.75/60) = 14.28$$

De esto resultó que el tamaño de la muestra a utilizar es de 14 personas. En este trabajo se utilizó un muestreo no probabilístico. Si bien, toda la muestra tenía igual posibilidad de ser incluida en la selección, ya que se colocó la encuesta en el forum, se sabía que las personas que participan activamente eran quienes iban a responder. Esto era algo deseable, ya que son éstas personas las que en la actualidad están más involucradas en el bordado de cruceta.

#### D. Comparación con otros programas

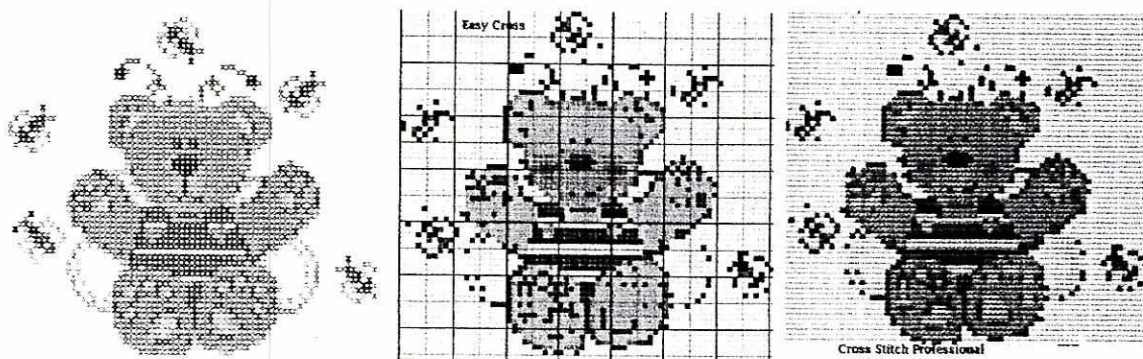


Fig 11.1 Comparación con otros patrones generados por otros programas

## E. Ejemplos

Color	Symbol	DMC	Color
■	=	824	Blue_VYDK
■	/	535	Ash_Grey_VY_LT
■	↑	319	Pistachio_Green
■	×	163	Green
■	7	792	Cornflower_Blue_DK
■	◇	3345	Hunter_Green_DK
■	◇	562	Jade_MED
■	×	930	Antique_Blue_DK
■	×	798	Delft_DK
■	^	826	Blue_MED
■	^	3363	Pine_Green_MED
■	^	3346	Hunter_Green
■	^	3347	Yellow_Green_MED
■	^	169	Pewter_Grey
■	^	470	Avocado_Green_LT
■	^	989	Forest_Green
■	^	502	Blue_Green
■	^	320	Pistachio_Green
■	^	341	Blue_Violet_LT
■	^	932	Aquamarine
■	^	793	Cornflower_Blue_MED
■	^	610	Drab_Brown_VYDK
■	^	611	Drab_Brown_DK
■	^	553	Violet
■	^	340	Blue_Violet_MED
■	^	640	Beige_Grey_DK
■	^	317	Pewter_Grey
■	^	166	Lime_Green
■	^	642	Beige_Grey_DK
■	^	368	Pistachio_Green_LT
■	^	932	Antique_Blue_LT
■	^	563	Jade_LT
■	^	841	Beige_Brown_LT
■	^	3041	Antique_violet_MED
■	^	809	Delft
■	^	3752	Antique_Blue_VYLT
■	^	503	Blue_Green_MED
■	^	318	Steel_Grey
■	^	927	Grey_Green_LT
■	^	612	Drab_Brown_MED
■	^	372	Mustard_LT
■	^	966	Baby_Green_MED
■	^	964	Sea_Green_LT
■	^	223	Shell_Pink_LT
■	^	613	Drab_Brown_LT
■	^	316	Antique_Mauve_MED
■	^	415	Pearl_Grey
■	^	644	Beige_Grey_MED
■	^	3042	Antique_Violet_LT
■	^	3747	Blue_Violet_VYLT
■	^	762	Pead_Grey_VYLT
■	^	3033	Mocha_Brown_VYLT
■	^	211	Lavender_LT
■	^	369	Pistachio_Green_LT
■	^	739	Tan_ULI_VYLT
■	^	746	Off_White
■	^	3756	Baby_Blue_ULI_VYLT
■	^	402	Mahogany_VY
■	^	3770	Flesh_VYLT

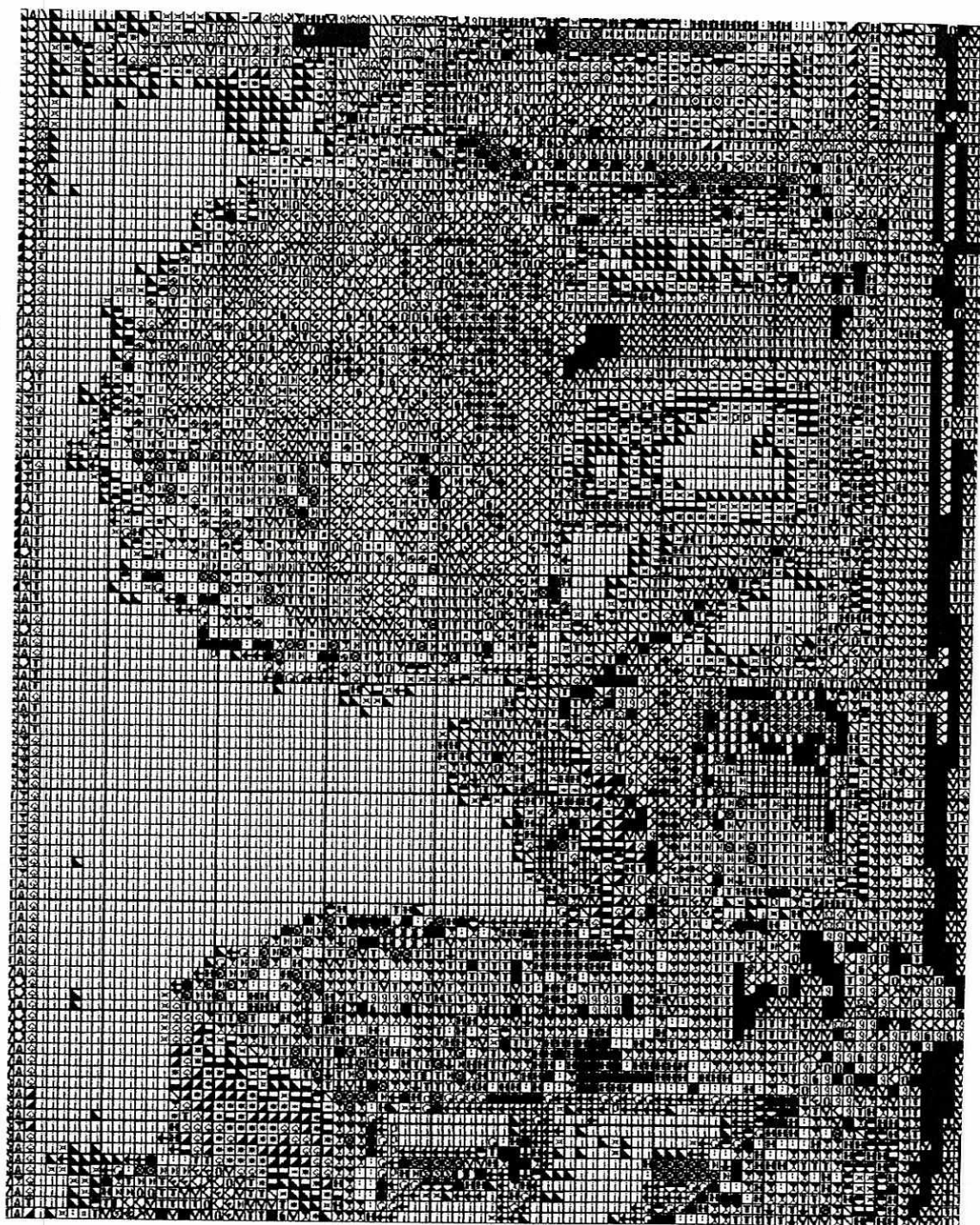


Fig 11.2 Ejemplo de patrón generado (Country)

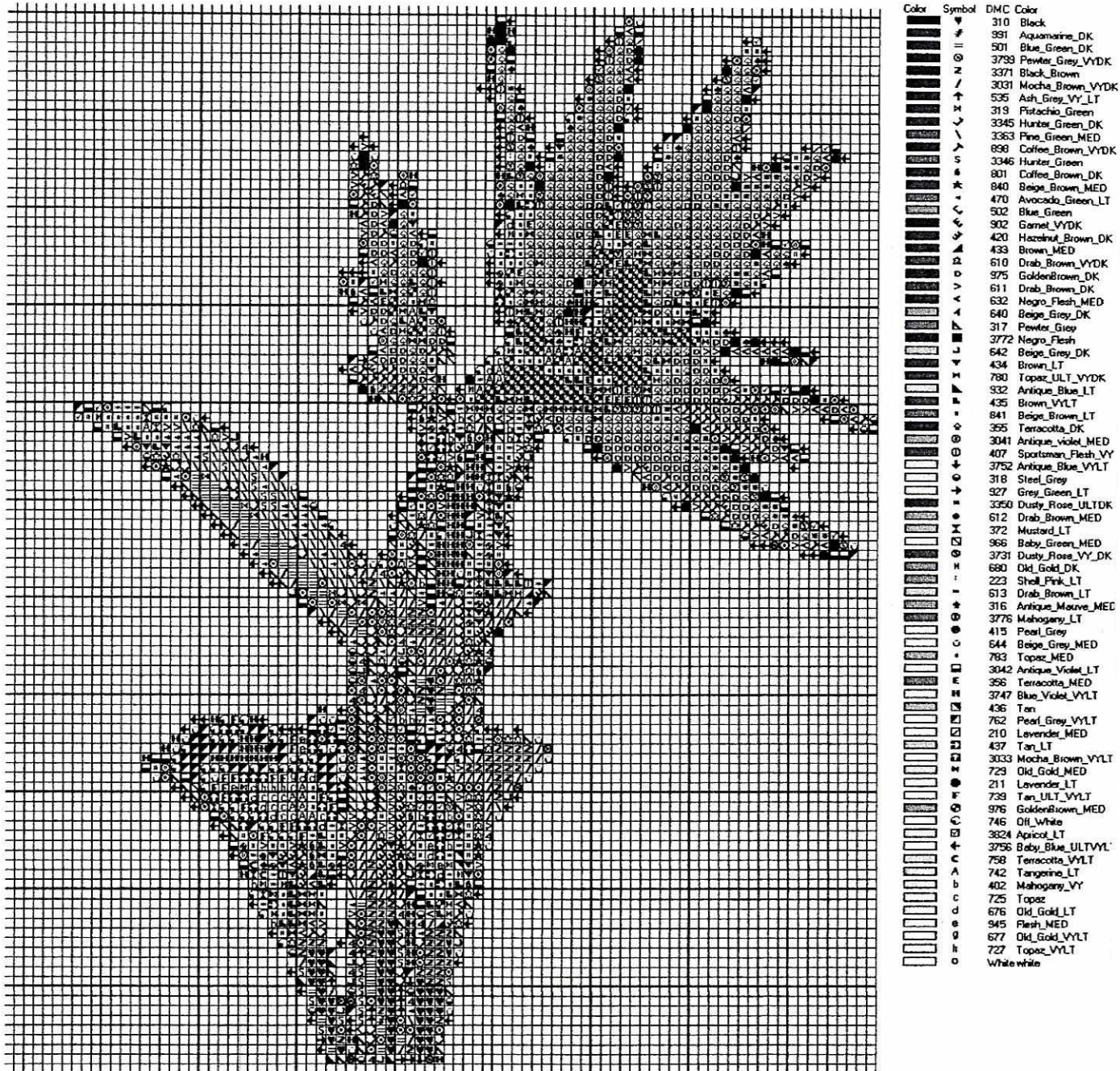


Fig 11.3 Ejemplo de patrón generado (Ave del Paraíso)



Fig Ejemplo de patrón ya bordado (Country)



Fig 11.4 Ejemplo de patrón ya bordado (Ave del paraíso)

Por:  
**Silvia Cabrera**

# Little Stitch

## Manual del Usuario

UV6

Agosto, 2004

### Indice

Introducción	1
Instalación	1
Creando tus propios patrones	2
Seleccionando los hilos	2
Reducción de Color	2
Reducción de Tamaño	3
Contornos	3
Imprimiendo el patrón	3
Generando XML	4
Guardando el patrón	4

### Little Stitch ....

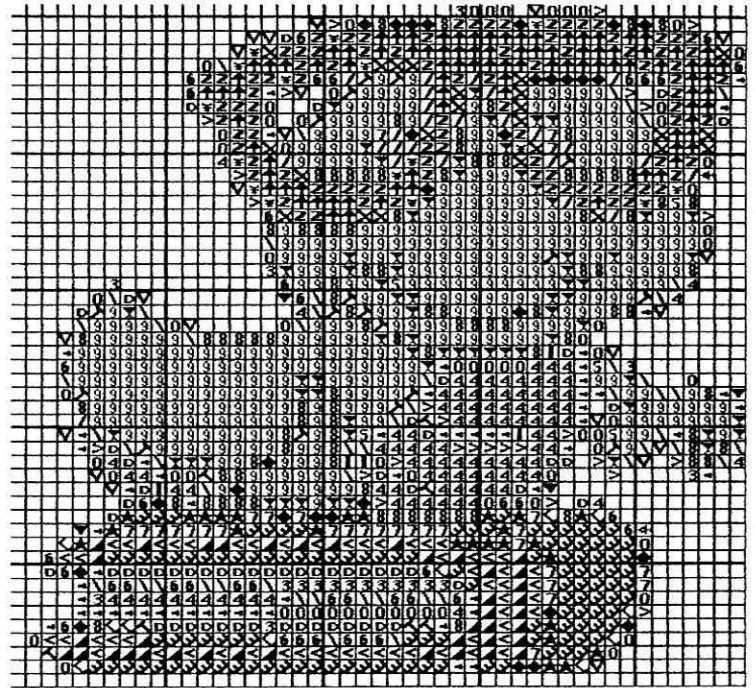
- A partir de cualquier bitmap obtiene un patrón de bordado de cruzeta
- Utiliza los colores DMC que tú desees.
- Reduce tu imagen a cuantas puntadas necesites.
- Imprime el patrón de símbolos, para que tu disfrutes bordando cruzetas y realizando arte.

## Introducción

*Little Stitch* es un programa diseñado para que puedas crear tus propios patrones de bordado de cruzeta

Con *Little Stitch* podrás hacer transformaciones a cualquier imagen .bmp para adaptarla a los colores de hilo que tienes disponibles, y al tamaño de bordado que tú desees.

Si tienes alguna imagen que te gusta mucho, de la cual no tienes el patrón para bordarla, puedes utilizar *Little Stitch* para obtener dicho patrón de símbolos y así empezar a bordar.



Cree sus propios patrones de bordado de cruzeta con Little Stitch

## Instrucciones de instalación

Para instalar *Little Stitch* en tu computadora haz lo siguiente:

- Inserta el CD en tu computadora.
- Del CD copia el folder "*Little Stitch*" al disco duro de tu computadora.
- Para poder correr el programa busca el lugar donde copiaste el folder, Dentro del folder "*Little Stitch*" dá doble clic sobre el programa: "*LittleStitch.exe*"
- El programa empezará a ejecutarse en ese momento.

# Creando tus propios patrones

Para poder crear tus propios patrones ejecuta el programa LittleStitch. Ya en el programa, en el menú superior, selecciona la opción **File** y luego selecciona la opción **Acquire Image**.

En la pantalla aparecerá una ventana donde debes seleccionar la imagen a convertir. Selecciónala y luego haz click en **Open**.

Luego aparecerá un recuadro con las diferentes combinaciones que puedes seleccionar para crear el patrón. Entre estos puedes seleccionar las opciones de hacerle reducción de tamaño, de color, y la manera de calcular los contornos.

Al seleccionar cualquier método de reduc-



ción de color el programa pondrá de manera predefinida la paleta de colores DMC.

Ya que te terminaste la selección haz click en **Go**. Y el programa empezará su trabajo.

Luego te aparecerán varias opciones que puedes seleccionar para obtener diferentes vistas de la imagen:

1. **Transformed Image:** Te permite ver la imagen transformada.
2. **Generate Cross Pattern:** Crea vistas de como se vería el patrón bordado en puntos de cruz. Para ayudarte a saber si la imagen te gusta, antes de bordarla.
3. **Generate Symbols Pattern:** Genera el patrón de símbolos y la lista de colores.
4. **Edges:** Muestra los contornos de la imagen.

## Seleccionando los hilos a utilizar

La paletas de colores utilizadas por *Little Stitch* están basadas en los colores definidos por la empresa DMC, para los hilos de algodón Mouline, Art 117.

*Little Stitch* te da diferentes opciones para que elijas qué colores de hilo quieres en tu patrón.

Si quieres utilizar los colores de hilo que ya tienes disponibles, antes de importar la imagen, en el menú superior selecciona **"Palette"**, luego **"Add Color"**. Allí podrás ingresar los colores DMC que tienes en tu colección de hilos. Luego, allí mismo en **Palette**, puedes guardar tu paleta de colores para futuros patrones en **Save Palette**.

Y la puedes seleccionar utilizando la opción **Load Palette**.

También puedes crear paletas de color, con los colores específicos que quieres que tenga tu bordado. Selecciona solo esos números de color y luego importa la imagen.

Color	Symbol	DMC Color
█	▼	500 Blue_Green_VYDK
█	♣	934 Black_Avocado_Green
█	×	779 Brown
█	=	3031 Mocha_Brown_VYDK
█	♣	838 Beige_Baom_VYDK
█	⊙	3862 Mocha_Beige_DK
█	z	3011 Khaki_Green_DK
█	/	3863 Mocha_Beige_MED
█	↑	320 Pistachio_Green
█	⊞	3364 Pine_Green
█	7	3012 Khaki_Green_MED
█	↓	523 Fern_Green_LT
█	⊞	3053 Green_Grey_187
█	⊞	3032 Mocha_Brown_MED
█	×	3864 Mocha_Beige_LT
█	A	644 Beige_Grey_MED
█	\	3820 Straw_DK
█	^	739 Tan_ULT_VYLT
█	S	712 Cream
█	♠	3865 Winter_White
█	*	White white

Paleta de Colores

"Y si no quieres comprar más hilos *Little Stitch* te permite usar en los patrones los colores de hilos que ya tienes."

Si no te importa comprar más hilos, tu patrón puede crearse utilizando los mejores colores de toda la gama de colores de hilos DMC. Para eso, en el momento de importar la imagen, debes hacer clic en el cuadro "With DMC Palette". Y escribe el número máximo de colores que quieres que tenga el patrón.

## Reducción de colores

Entre las reducciones de colores que *Little Stitch* te permite hacer están:

**Color Change:** Esta opción es un cambio simple de cada color al color más parecido de los que se encuentran en la paleta de colores que se utilizarán.

**Color Change with Dithering:** Esta técnica mezcla los colores existentes para simular los colores de la imagen que no existen en los hilos. Esta opción permite una mejor impresión visual para imágenes de tonos grises. Pero las hace mucho más difíciles de bordar.

# Reducción de Tamaño

Al momento de importar la imagen, *Little Stitch* te permite seleccionar diferentes maneras de reducción de tamaño:

**Píxel Skipping: (Salto de Píxeles)** Hace la reducción saltando líneas de píxeles, tanto de manera horizontal, como vertical. Mantiene los colores de la imagen original. Es ideal para imágenes sin detalles pequeños.

"Juega con los diferentes tipos de reducción de tamaño, habrá alguno que se adapta mejor a tu imagen."

**Weighted Average: (Promedio Ponderado)** Realiza un promedio con una tabla de pesos. La imagen resultante es más pareja, sin mucho cambio de color o tonalidades. Si ese es el efecto que quieres lograr en el bordado, este es el recomendado.

**Resize:** Trabaja con el número máximo de puntadas verticales y horizontales. Si necesitas un bordado de medidas específicas, este es el algoritmo que debes utilizar.

**Expand:** Agranda la imagen. Si tienes una imagen muy pequeña que quieres agrandar puedes utilizar esta opción.

Todos los algoritmos, a excepción del **Expand** utilizan la casilla de **Scale (Escala)** que se encuentra en la parte superior al momento de importar la imagen. Si quieres que la imagen se mantenga del mismo tamaño de la imagen original la escala debe ser 1. Si la escala es 2 se reduce o agranda a la mitad de acuerdo a la opción elegida, y así sucesivamente.

**Resize** utiliza las casillas de "Horizontal Stitches" y "Vertical Stitches" (Puntadas Horizontales y Verticales). El número predeterminado de puntadas es el de la imagen. En las casillas de puntadas horizontales y verticales debes colocar el número máximo de puntadas que deseas que tenga tu bordado.

**Píxel Average: (Promedio de Píxeles)** Esta opción reduce la ima-

# Contornos

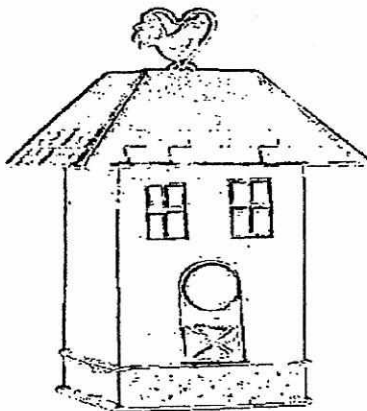
Los contornos te ayudarán a tener una idea de donde puedes agregarle bordes de punto atrás y así hacer que resalte tu imagen.

Al momento de importar la imagen, *Little Stitch* te da las siguientes opciones de contorno:

**Sobel B&W:** Este es el que provee bordes más finos. Te da una imagen de los posibles lugares donde se podría bordar punto atrás.

**Sobel GrayScale:** Transforma la imagen en una escala de grises que permite visualizar los contornos.

**Laplace B&W:** Crea bordes un poco más gruesos. Pero captura mayor cantidad de cambios en la imagen, y por lo tanto coloca más bordes.



**Laplace GrayScale:** Es una imagen en tonos de grises, donde se muestran los contornos.

Al momento de importar la imagen puedes seleccionar que tipo de borde quieres ver.

*Los bordes te ayudan a saber donde podrías agregar punto atrás para resaltar la imagen.*

# Imprimiendo el patrón

Ya que tienes el patrón que deseas bordar puedes imprimirlo. Para eso debes ir al menú superior, y seleccionar **File**, luego debes seleccionar **Print**.

Allí tienes 3 opciones:

- La primera es imprimir la imagen original (**Image**).
- La segunda es imprimir la vista

de la imagen en puntadas de cruceta (**Cross Stitch**)

- Y la tercera imprime el patrón de símbolos (**Symbols Pattern**) y su respectiva tabla de símbolos y colores

El patrón de símbolos impreso es una cuadrícula que tiene líneas resaltadas cada 10 cuadritos. Esto te ayudará a facilitar la cuenta de cuadritos durante el bordado.

  
**Little Stitch**

Trabajo de Graduación  
Licenciatura en Ciencias de la Computación

Agosto 2004

Por:  
**Silvia Cabrera**

UVG

## Guardando el patrón, y a bordar !!

Para finalizar, no olvides guardar el patrón que creaste. Este puede ser de utilidad en caso de que pierdas alguna de tus impresiones. O que quieras repetirlo nuevamente en el futuro.

Para guardarlo debes ir a **File** en el menú principal y luego ir a **Save**; allí al seleccionar **All** el programa almacenará todo en una sola carpeta: la imagen original, la imagen transformada, la vista de la imagen en punto de cruz, el patrón de símbolos y la paleta de colores.

Si en el futuro necesitas el patrón, en **File** seleccionas **Open Pattern**, y allí te preguntará dónde lo almacenaste. Al abrirlo tendrá toda la información de tu patrón cargada en el programa nuevamente.

En **Save** puedes almacenar todo por separado, si necesitaras hacerlo.

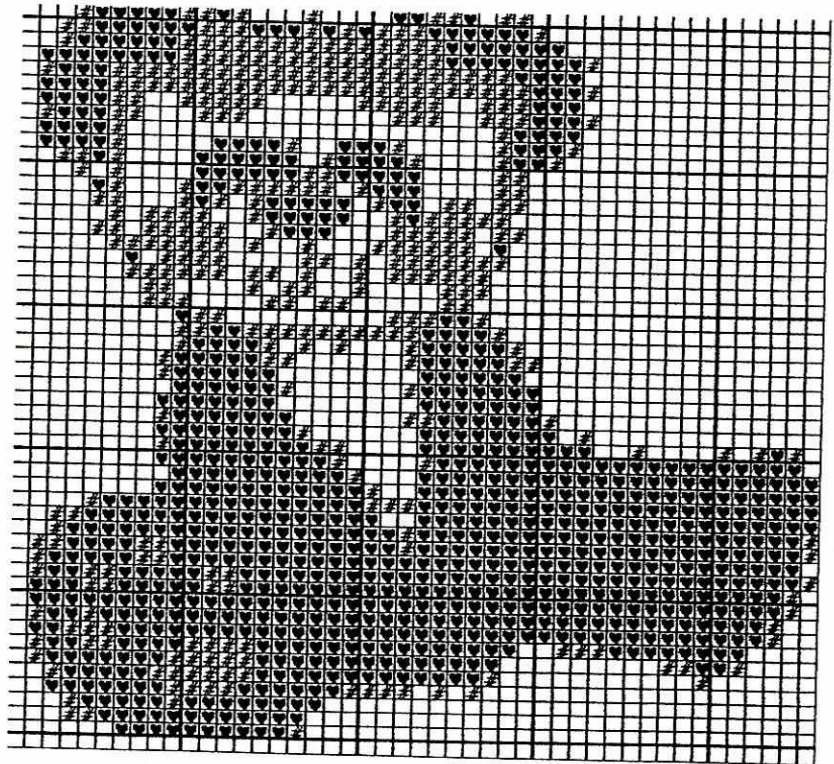
Ya que te aseguraste de almacenarlo estás listo para la mejor parte ...

**A bordar se ha dicho !!**

## XML para máquinas bordadoras

*Little Stitch* va un paso más allá de cualquier programa para generar patrones de bordado de cruceta; éste también genera los patrones en XML (eXtensible Markup Language) que es un lenguaje que se está convirtiendo en el estándar para compartir información. En un futuro cercano las máquinas bordadoras ya aceptarán XML, y los patrones hechos en *Little Stitch* serán fácilmente adaptables a estas máquinas.

Para generar el XML del patrón, luego de haber generado el patrón de símbolos, haz clic en **File** en el menú superior, y luego en **Generate XML**, esto te dará como resultado un archivo con toda la información del patrón: los colores de hilo y sus símbolos, cada cruz, su posición y su color.



## XII. BIBLIOGRAFÍA

- Los Materiales* [en línea] <<http://es.geocities.com/crucetaypunto/materiales.html>> [Consultada: 29 de Junio 2004]
- Leck, Cherie Mack. *Learn how to Stitch a simple heart Pattern* [en línea] <<http://www.craftdesigns4you.com/learntostitch.htm>> [Consultada 29 de Junio 2004]
- Tom Pudding Designs. *Ethnic Cross Stitch* [en línea]. <[http://www.tompudding.co.uk/ethnic\\_cross\\_stitch.htm](http://www.tompudding.co.uk/ethnic_cross_stitch.htm)> [Consultada 29 de Junio 2004]
- Visgraf Lab. *Floyd-Steinberg Dithering* [en línea] <[http://www.visgraf.impa.br/Courses/ip00/proj/Dithering1/floyd\\_steinberg\\_dithering.html](http://www.visgraf.impa.br/Courses/ip00/proj/Dithering1/floyd_steinberg_dithering.html)> [Consultada 2 de Julio 2004]
- Bitmap Vs. Vector Graphics* [en línea]. <<http://fla.esf.edu/research/CAVlab/reference/bmpvec.htm>> [Consultada 6 de Julio 2004]
- Hassanpour, Reza. *Gradient Based Edge Detection* [en línea]. <http://ari.cankaya.edu.tr/~reza/ImLab4.htm>> [Consultada 16 de Julio 2004]
- Jonathan Sachs. *Digital Image Basics* [en línea]. <<http://www.dl-c.com/basics.pdf>> [Consultada 19 de Julio 2004]
- ImageMagick Studio LLC. *ImageMagick.org* [en línea]. <<http://www.imagemagick.org>> [Consultada 23 de Julio 2004]
- Glenn J. Battaglia. *Mean Square Error* [en línea]. AMP Journal of Technology Vol. 5 June, 1996 <[http://www.amp.com/products/technology/5jot\\_5.pdf](http://www.amp.com/products/technology/5jot_5.pdf)> [Consultada 24 de Julio 2004]
- DMC. *DMC Creative World* [en línea] <[http://www.Dmc.com/majic/pageServer/0g01010005/es\\_ES/Home.html](http://www.Dmc.com/majic/pageServer/0g01010005/es_ES/Home.html)> [Consultada 13 de Agosto 2004]
- Muestreo y tamaño de la Muestra* [en línea]. <<http://www.monografias.com/trabajos12/muestam/muestam.shtml>> [Consultada 17 de Agosto 2004]

*pamscale* [en línea]. <<http://netpbm.sourceforge.net/doc/pamscale.html>> [Consultada 21 de Agosto 2004]

Helena Normark. *Graphic Garden* [en línea]. <<http://www.graphicgarden.com>> [Junio 2004]

Ceponus, Alex; Faraz Hoodbhoy. 1999. *Applied XML, A Toolkit for Programmers*. Estados Unidos, Wiley Computer Publishing, 474 págs.

Lara, Gustavo Adolfo. 2003. «*Técnicas de reconocimiento de imágenes para la creación de Fotomosaicos*». Tesis Universidad del Valle de Guatemala. 51 págs.

Dave Mercer. 2002. *Fundamentos de Programación XML*. Colombia , Osborne Mc. Graw Hill. 436 págs.

