

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE CIENCIAS

CREACIÓN DE IMÁGENES NO FOTORREALISTAS UTILIZANDO STIPLING

Tesis profesional que presenta:

AMADO REYES VELÁZQUEZ MONTALVO

Aprobado por:

M.C. José Ignacio Ascencio López
Presidente del Jurado

Dr. José Angel González Fraga
Secretario

M.C. Adán Hiraes Carbajal
Vocal

Dedicatorias

A mis amados padres Zenón y María, a quienes les hubiese dado tanto gusto haber visto que lo logré.

A Beatriz la roca a la que tantas veces me aferré, todo te lo debo a ti mi amor.

A mi querido hijo Iker el motivo de todos los motivos.

Agradecimientos.

Un agradecimiento muy grande a mi amigo y profesor José Ignacio Ascencio López, un hombre íntegro, a quien le debo el aprender a pensar más allá de mis limitantes y gran parte de esta tesis....gracias.

Resumen de la tesis de Amado Reyes Velázquez Montalvo presentada como requisito parcial para la obtención de la Licenciatura en ciencias Computacionales.

Ensenada, Baja California, México, abril de 2009.

CREACIÓN DE IMÁGENES NO FOTORREALISTAS UTILIZANDO STIPPLING

Resumen aprobado:

M.C. José Ignacio Ascencio López

Director de tesis

La recreación de imágenes no fotorrealista por computadora es un área relativamente joven en el procesamiento digital de imágenes, que reviste cierto interés particular de investigación.

La presente tesis documenta una de las técnicas artísticas que es base para el desarrollo de otras más avanzadas: el Stippling.

Se establecen 3 elementos principales que determinan la calidad de una recreación no fotorrealista: la correcta representación de los niveles de intensidad; una adecuada distribución espacial de los elementos pictóricos que conforman la representación; y la no aparición de patrones ajenos al objeto representado (no inducidos con un propósito específico).

Los métodos presentados (métodos basados en la geometría de la imagen, métodos probabilísticos y un método basado en agentes inteligentes) resuelven de diferente forma el proceso de recreación, observando algunos aspectos de calidad mencionados.

En cada caso se presentan los criterios que les rigen, un análisis del orden del tiempo de ejecución y un comparativo de sus resultados por simple inspección visual.

Contenido

Lista de figuras	vi
Capítulo I: Introducción	1
I.1 Introducción	1
I.2 Antecedentes del Strippling	5
Capítulo II: Antecedentes.....	9
II.1 La percepción dentro de la visión humana	9
II.2 Las imágenes fotorrealistas y no fotorrealistas	14
II.3 Cuantización	19
Capítulo III: Uso de métodos Halftoning en el Stippling	24
III.1 Orígenes del Stippling Computacional	24
III.2 Métodos Basados en la Geometría.....	30
III.3 Difusión de error	37
III.4 Métodos Probabilísticos	40
Capítulo IV: Uso de Diagramas de Voronoi en el Stippling.....	49
IV.1 Características artísticas del Stippling	49
IV.2 Método de Lloyd.....	56
IV.3 Diagramas de Voronoi	60

IV.4 Algoritmo basado en el hardware para gráficos.....	63
Capítulo V: Uso de Agentes inteligentes (AI) en el Stippling.....	71
V.1 Un método generalizado.....	71
V.2 Simulación de un Sistema de partículas mediante agentes inteligentes.....	74
V.3 Agentes Inteligentes (AI)	77
V.4 StippleBots	80
V.5 El Medio Ambiente (MA).....	82
V.5 El Estado Simular	85
V.5 El Estado Mover y Pintar	89
V.6 Algoritmo	90
Capítulo VI: Comparativo de resultados.....	95
VI.1 Métodos Halftoning	95
VI.2 Método del Rechazo	102
VI.3 Método del Lloyd.....	106
VI.3 Método de Agentes Inteligentes.....	109
Capítulo VII: Conclusiones	112

Lista de Figuras

<i>Figura</i>	<i>Página</i>
Figura 1. El dithering compara cada elemento de la imagen $f(x,y)$ con el umbral correspondiente de la matriz para colocar o no, un píxel negro en la imagen de salida. ...7	
Figura 2. a) Imagen de Lenna producida por una matriz de oscilación de 2×2 b) imagen Stippling de la misma fuente.....8	
Figura 3. La secuencia en la visión de acuerdo a Marr.....10	
Figura 4. En la esquina inferior derecha, el contorno de la base es imperceptible, pero somos capaces de interpretar la forma11	
Figura 5. Imagen originalmente a color despojada del componente cromático.....13	
Figura 6 Resultado del muestreo (b) de una señal continua original (a).....17	
Figura 7 Dimensionalidad de una imagen. Esta representación corresponde al domino espacial de la imagen muestreada en términos de $f(x,y)$19	
Figura 8. A cada región que va de $d_i - 1$ a d_i se le denomina un bin de cuantización..22	
Figura 9. Pintura Puntillista (tomada de Wikipedia).....24	
Figura 10. Cinco niveles de intensidad aproximada con una unidad de superficie de cuatro puntos29	
Figura 11. (a) Imagen Original (b) Imagen Binarizada con un umbral de 128 (c) acercamiento31	
Figura 12. El dithering compara cada elemento de la imagen $f(x,y)$ con el umbral correspondiente de la matriz para colocar o no, un píxel negro en la imagen de salida ..32	

Figura 13. Patrones que representan las 10 intensidades de una matriz dither 3x3.....	33
Figura 14. Kernel Filtro de Lloyd-Steinberg.....	37
Figura 15. Generación de valores de salida que se calculan considerando la imagen como una FDP.....	43
Figura 16. a) 200,000 posiciones b) con 4,000 stipples, en ambas imágenes los valores aleatorios para la colocación de puntos tienen una distribución Uniforme.....	45
Figura 17. a) Ilustración con 100000 píxeles negros con distribución PD sobre la función de densidad formada por la imagen de entrada b) Imagen stippling con 40000 primitivos de radio 3.....	47
Figura 18. Detalle aumentado 13 veces, donde se observa las diferentes agrupaciones de píxeles que forman regiones oscuras y regiones claras.....	51
Figura 19. Diagrama de Voronoi producido con 20 sitios generadores, y un espacio V de 512 por 512 píxeles.....	61
Figura 20. Delineado de la base circular de un cono utilizando la característica de interpolación del hardware de rasterización.....	64
Figura 21. Funcionamiento de un Triangle Fan. (Tomado de (Shreiner, y otros s.f.))....	64
Figura 22. Formación de dos regiones Voronoi.....	66
Figura 23. Construcción del diagrama de Voronoi mediante el dibujo de conos b) Vista ortogonal desde arriba.....	66
Figura 24. Primera etapa en el proceso de generación de una imagen con Stipples utilizando Voronoi.....	67

Figura 25. (a) Regiones de Voronoi originales, (b) primera iteración, (c) segunda iteración (d) iteración número 50 prácticamente todos los puntos generadores y los centroides han convergido.....	69
Figura 26. Imágenes de la planta de maíz y Lena generadas con Stipples mediante el algoritmo de Lloyd.....	69
Figura 27. Múltiples trazos en una misma imagen NPR. Tomada de (Schlechtweg, Germer and Strothotte Volume 24 (2005), Number 2).	73
Figura 28. Layers de una imagen.	75
Figura 29. Diagrama de Estados	84
Figura 30. Fuerzas que afectan a un StipplingBot	85
Figura 31. Imagen formada con una matriz dither 4x4 que induce patrones triangulares.	95
Figura 32. (a) Imagen generada con un Dither 2x2 y (b) realce de un segmento de la misma donde se pueden observar los patrones generados	96
Figura 33. Imagen generada por binarización.....	97
Figura 34. (a) Imagen generada con un Dither 3x3 (b) figura generada con un Dither 4x4.....	98
Figura 35. Forma en la que se difunde el error	100
Figura 36. Imagen Binarizada de Lena con difusión de error y acercamiento.....	100
Figura 37. (a)Imagen de Lena con Dtiher y Difusión de error.....	101
Figura 38. Imagen de Lenna generada por el método del rechazo con 30000 puntos. .	103

Tablas.

I	Cálculo del orden de ejecución Dither	35
II	Costo de la difusión del error	42
III	Tiempo de ejecución método del rechazo	47
IV	Tiempo de ejecución, método Agentes	93

Introducción

I.1 Introducción

Intuitivamente se puede entender al término fotorrealismo como el conjunto de características que tiene una imagen, generada totalmente por computadora, que intenta imitar los resultados que se tendrían al obtener esa misma imagen mediante un dispositivo fotográfico real.

El fotorrealismo es una tecnología dirigida, en la cual la computadora se utiliza para modelar el trabajo de otra máquina, la cámara (Strothotte 2002).

El trazado de rayos, los modelos de iluminación, y hasta los sistemas de diseño asistido por computadora generan imágenes fotorrealistas. Imágenes frías, calculadas a partir de modelos numéricos, precisas y que consideran una mayor cantidad de variables que caracterizan la escena: objetos geométricos que asemejan a los reales, propiedades ópticas como la reflexión, la refracción, la atenuación en sus diferentes formas, las sombras etc.

Por otro lado, y también de una forma sencilla se puede referir a las imágenes no-fotorrealista como aquellas que no son fotorrealistas. El área no-fotorrealista busca la generación por computadora de imágenes y animaciones que, generalmente hablando,

aparenten haberse creado “a mano”. Intentan captar la escena como un artista o artesano la ilustraría, contrario a las gráficas fotorrealistas que son técnicas algorítmicas que se asemejan a la salida de la cámara fotográfica y que crean esta salida siguiendo las leyes físicas involucradas en el proceso fotográfico.

Los dibujos artesanales tienen ventajas sobre las ilustraciones fotorrealistas. “Dibujando a mano, es posible liberarse de las limitaciones físicas de la realidad para converger en una impresión más allá de la simple representación de los detalles que aparecen en la escena” (Hertzmann 2006).

El propósito de NPR (Non-Photorealistic Image Rendering, que es como se conoce al área), es la creación de imágenes no-fotorrealistas. A lo largo de la historia de la humanidad, muchas de las representaciones gráficas que ha producido el hombre recrean la realidad y no solo la imitan.

Por ejemplo, en algunos grabados egipcios (Strothotte 2002) el artista se tomó la libertad de dibujar los sujetos en posiciones que serían imposibles de existir en la realidad, esto más con el propósito de impactar en la transmisión de una mensaje que de la imagen per se.

Otro ejemplo lo tenemos en algunos mapas turísticos donde se vuelve muy útil insertar ideogramas o distorsionar las proporciones de algún punto de particular interés para el visitante.

En un libro de medicina la ilustración de un esquema, hecho a mano o con técnicas no-fotorrealistas, de las partes de un órgano, es mucho más didáctico que el solo contar con las fotos del mismo.

La calidez que dan las ilustraciones hechas a mano en un libro de cuentos infantiles no tiene nada que ver con las fotografías que aparecen en un libro técnico, por ejemplo.

Por lo tanto al generar una imagen no-fotorrealista se pretende lograr que tenga algunas de las características presentadas.

Esto tiene implicaciones más allá de lo artístico, como el manejo de los aspectos psicológicos de la percepción visual, al lograr la “imperfección” que un dibujo a mano tiene, y que finalmente busca la transmisión del mensaje más que en la mímica de una escena (aunque la matemática involucrada en el proceso a veces resulte muy compleja).

Un ejemplo ilustrativo muy simple de este tipo de complejidades, es el de un dibujo navideño que representa una escena en la cual están cayendo copos de nieve y estos copos se copian repetitivamente y provocan la percepción de un patrón perfectamente identificable.

De inmediato al percibir el patrón en la imagen, para el ojo la imagen pierde cierto encanto. Esto debido a que la imagen se vuelve burda, falsa.

Entonces nos enfrentamos con el problema de que en el contexto no-fotorrealista aunque no se busca una representación exacta de la realidad, sí se tiene que cuidar la predisposición que nuestro cerebro tiene al compararla con las características que el objeto representado tiene en la realidad.

Lo mismo sucede cuando queremos reproducir una técnica artística o artesanal. no solo se trata de que inequívocamente la imagen sea detectada como algo diferente a una fotografía, sino que también debe contener las características de la técnica que estamos emulando, ya que en caso contrario nuevamente sería una representación falsa.

El dibujante tiene la libertad de incluir en su dibujo elementos que no forzosamente se dan en la realidad, pero que están cargados de significados que el propio observador entenderá.

Incluso la falta de información en cierto modo permite al cerebro del observador rellenar los huecos con información que él conoce que ahí va. La complejidad al obtener todas estas características en una imagen a través de algoritmos es alta y las diversas técnicas alcanzan a cubrir varias de ellas, pero no todas. Entre estas técnicas se haya el llamado stippling.

Una técnica artística y de ilustración (conocida como grabado) coloca pequeños puntos de tinta sobre el papel de acuerdo con la densidad dada en la impresión del tono. Un artista que cultiva esta técnica, es muy cuidadoso al colocar cada punto sobre la hoja para representar espacialmente, en lugar de intensidad, lo que intenta comunicar. Esto

explica porqué puede tomar semanas y hasta meses la elaboración de una ilustración que por naturaleza es de tamaño limitado.

Esta técnica en NPR se conoce como 'stippling' y busca colocar 'stipples' (primitivos gráficos que pueden ser puntos, círculos, triángulos u otras figuras básicas incluyendo las irregulares) muy pegados en regiones oscuras y más separados en regiones luminosas para emular la intensidad de una imagen (Ascencio et al. 2006). Los stipples como ya lo definimos deben colocarse de manera un tanto aleatoria para que el ojo humano no observe falsos patrones que no son parte de la impresión pretendida.

Existen otras técnicas NPR que comparten una problemática común con el stippling: hatching, cross hatching, mosaicos, el uso de trazos (strokes), texturizado, etc. Al estudiar cuidadosamente el stippling, se está adentrando definitivamente en un área central para NPR, pues con otros elementos adicionales (gradiente direccional, tamaño del mosaico o segmento de curva, etc.) pueden recrearse las otras técnicas.

I.2 Antecedentes del Strippling

El primer antecedente de Stippling es sin duda el método de aproximación por tonos medios. Esta técnica regenera imágenes en tonos de gris con un conjunto de puntos que, bajo su distribución representan diferentes intensidades, cambiando la resolución de intensidad por resolución espacial. Los métodos de tonos medios o halftoning se usan principalmente en la impresión de libros o periódicos. Si se observa con una lupa una imagen en un libro, se podrá ver que está compuesta de múltiples puntos negros

acomodados de tal manera, que emulan la intensidad de la imagen, pero no se utilizan tintas de diferente intensidad.

Otros antecedentes son el dithering y los esquemas de dispersión de error como el de Floyd y Steingberg, que están difundidos ampliamente en la bibliografía (Foley et al. 1994).

Dithering es una técnica para la aproximación de tonos medios auxiliar en la impresión y despliegue de imágenes con menores intensidades que la imagen original. Su función es minimizar el error producido por la acción de cuantización de intensidades. En el despliegue, dithering es el proceso de yuxtaponer píxeles de dos colores o intensidades para crear la ilusión de que un tercer color o intensidad está presente.

Consideremos una imagen $f(x,y)$ de tamaño N_x por N_y con valores de pixel en el rango $\{0,1,\dots,D-1\}$. Para generar la imagen de tonos medios $b(x,y)$ por este método, se usa el arreglo $d(u,v)$ de tamaño M_u por M_v llamado matriz dither, donde los elementos son umbrales de cada nivel. Cada elemento de la matriz dither toma valores en el rango $\{0,1,\dots,D-1\}$. Para generar la salida se realiza la umbralización elemento por elemento de los píxeles de la imagen con los componentes de la matriz dither como se muestra en la figura 1.

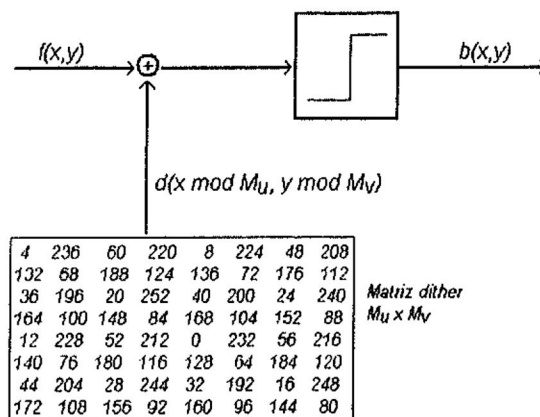


Figura 1. El dithering compara cada elemento de la imagen $f(x,y)$ con el umbral correspondiente de la matriz para colocar o no, un píxel negro en la imagen de salida.

Sin embargo el stippling es diferente pues utiliza un número significativamente menor de puntos, los cuales pueden variar en tamaño y hasta en forma (Deussen et al. 2000). Los puntos orientados en los bordes pueden usarse para resaltar detalles y lo más importante: no añaden patrones visuales no inherentes al dibujo a representar.

Si en el stippling se usa una cantidad exhaustiva de puntos para representar los tonos de gris, la imagen será perfectamente identificable pero se limitará en el tipo de abstracción por parte del cerebro del observador. Si por el contrario se reduce el número de puntos, forzosamente producirá huecos en la representación que deberán ser rellenados por el propio observador dándole una característica diferente a la representación que ya cae en el terreno del no-fotorrealismo.

Este sencillo método (dithering) se diferencia de los métodos que usan stipples, además de usar un número exhaustivo de puntos, por la aparición de artefactos o falsos patrones

en la imagen producida, como muestran las formas regulares apreciables en la figura 2

(a)

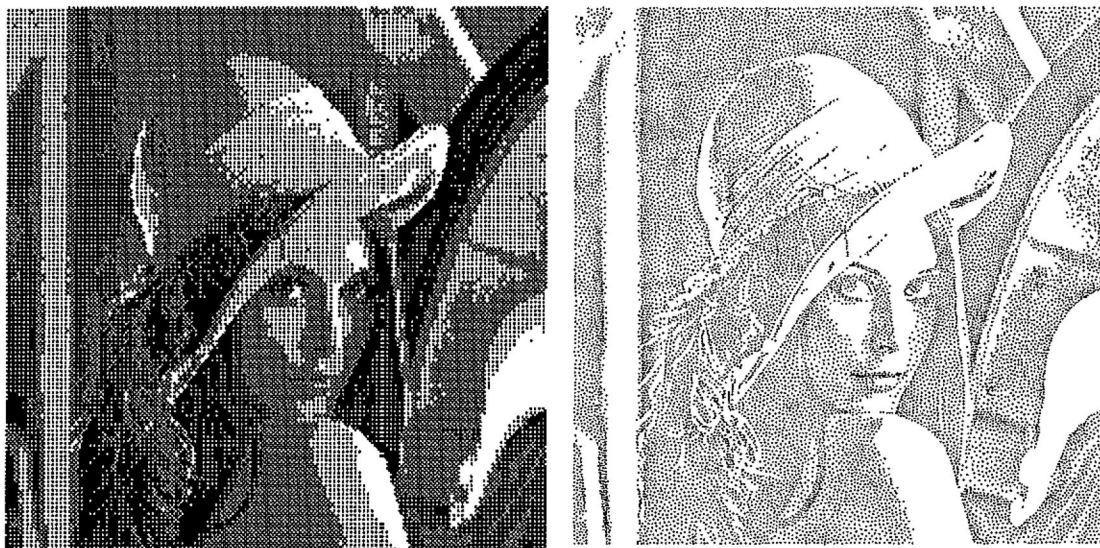


Figura 2. a) Imagen de Lena producida por una matriz de oscilación de 2×2 b) imagen Stippling de la misma fuente

II.1 La percepción dentro de la visión humana

El espectro electromagnético es un continuo que va desde 10 pm (picómetro: una billonésima parte de un metro) hasta los 10 Km, abarcando desde los rayos gamma hasta las frecuencias de radio muy bajas, pasando por la luz visible. De ese espectro la parte que el ojo humano es capaz de captar, va desde los 400 nm (nanómetros: milmillonésima parte de un metro) hasta los 700 nm.

El ojo humano tiene dentro de su glóbulo, al fondo del mismo, alineado con el iris una agrupación de células nerviosas (fóvea) especialmente sensibles a la información cromática (color), dichas células se les denomina conos. En la periferia hay otro tipo de células (los bastones) sensibles a información acromática (luminosidad). Ambos tipos de células transmiten al cerebro (mediante el nervio óptico) información referente a la imagen.

Nuestra capacidad para captar esta información es limitada, de hecho el proceso de la vista principalmente se desarrolla en el cerebro; y es en sí, un proceso abstracto de recreación de la imagen (Jérome 2002).

En años recientes la Psicología Cognoscitiva ha enfrentado el problema de cómo percibimos el mundo desde un punto de vista diferente, esto es desde el punto de vista del procesamiento de datos. Los modelos actuales provenientes de esta disciplina integran dos modelos viejos en un “framework” coherente.

En la figura 3 se esquematiza a la percepción como un proceso "bottom-up" que inicia con la imagen retinal, de donde se extraen los rasgos simples de la imagen como un bosquejo y organiza las características en objetos más y más complejos.

La segunda parte establece que la percepción está guiada por nuestro conocimiento previo y que consiste principalmente en hacer concordar las representaciones del mundo con la imagen retinal (acercamiento “top_down”). La experimentación señala que estas dos estrategias son trabajadas simultáneamente y que una influencia a la otra.



Figura 3. La secuencia en la visión de acuerdo a Marr

En 1982, Marr presentó un acercamiento computacional al procesamiento visual "bottom-up". Este método divide la percepción del objeto en un cierto número de escenarios que son trabajados secuencialmente. El primer escenario consiste en la extracción de los rasgos desde la imagen retinal.

Estos rasgos incluyen bordes y esquinas los cuales son representados por los cambios de iluminación en la imagen retinal, Aunque la extracción de rasgos se ve como una tarea simple, la figura 4 muestra cómo el delineado de un objeto no es claramente discernible en la retina, debido a las zonas oscuras y a las sombras. En este caso solamente es posible determinar correctamente los rasgos si manejamos lo abstracto y tomamos ventaja del conocimiento conceptual que tenemos del objeto en cuestión (en el caso de la figura 4 sabemos que se trata de un florero porque hacemos uso de lo que conocemos sobre la forma de los floreros).

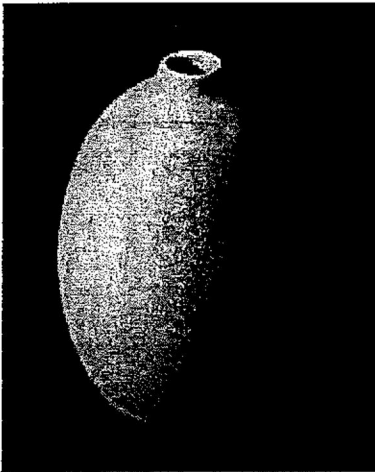


Figura 4. En la esquina inferior derecha, el contorno de la base es imperceptible, pero somos capaces de interpretar la forma

Durante la segunda etapa los rasgos de nivel-bajo son extraídos de la imagen retinal y son integrados en un esquema $2 \frac{1}{2} D^1$ del objeto. En esta etapa se trata de determinar el

¹ El método $2 \frac{1}{2} D$ representa una extensión al acercamiento de “solo basado-en-la-imagen”, trabaja con la base de imágenes aumentadas que incluyen, además del valor de color, información de profundidad o ID's que identifican diferentes áreas de una imagen.

delineado del objeto en la recreación de la imagen y por lo tanto garantizar que los rasgos sean detectados correctamente. Lo que puede ser útil para incluir delineados de rasgos que están ocultos por otros objetos presentes. Claro que estas observaciones no son nuevas y han sido utilizadas por los ilustradores durante mucho tiempo, pero el modelo psicológico puede ser considerado como una justificación para la observación intuitiva.

En 1920 un número de psicólogos empezaron investigaciones sobre la relación entre el estímulo visual y las formas percibidas. Su principal tesis fue que el todo es mayor que la suma de las partes, lo cual significa que nuestro sistema perceptual organiza y clasifica el estímulo visual en construcciones de orden mayor que no necesariamente contiene la imagen retinal. (Jérôme 2002)

La experiencia de percibir el color se asocia a los sentimientos que se despiertan al observar la imagen: amor, tristeza, hambre, etc. (Iterrante 2007)

Podemos observar en la figura 5 que al despojar una imagen de su componente cromático en realidad no hay pérdida sobre la información de forma, textura y línea. En este sentido la información acromática de la imagen es suficiente, y la imagen no sufre gran pérdida, en esos términos, al despojarse del color.

El sombreado afecta la percepción de forma y profundidad, elemento que está dado por como la escena está iluminada, las sombras de la figura 5 hacen que las formas del primer plano se pierdan pero incrementan la sensación de profundidad que hay en la fotografía.

Por último se percibe claramente en el techo que está en la parte central inferior de la imagen, una cierta textura rugosa de un material diferente al de las construcciones alrededor.

Estos mecanismos son importantes en el estudio del procesamiento digital de imágenes (y por ende en NPR), ya que las técnicas que se aplican, hacen uso de sus características.

En particular se hace uso también de la llamada Integración Espacial, la cual consiste en que si vemos un área pequeña desde una distancia lo suficientemente grande, nuestros ojos promedian el detalle fino dentro del área y registran únicamente la intensidad global de la misma.

II.2 Las imágenes fotorrealistas y no fotorrealistas

Para poder ubicar las diferentes formas en que se puede percibir una imagen, (Bovick et al. 2005) nos presenta una clasificación sencilla y funcional, que aún cuando tiene el problema de que una imagen puede llegar a clasificarse en más de una categoría, de acuerdo a la fuente que la origina.

Imágenes de reflexión: Es la radiación sentida que ha sido reflejada desde la superficie de los objetos.

Imágenes de emisión: son incluso más simples, dado que en este caso los que forman la imagen de por sí son objetos luminosos.

Imágenes de absorción. Produciendo información acerca de la estructura interna de los objetos (como en los rayos X).

Ahora bien con el uso de las computadoras, y del procesamiento de imágenes digitales, se establece una clasificación de acuerdo a la fuente de la cual provienen.

Las imágenes puede ser: adquiridas (mediante cualquier dispositivo de captura) o pueden ser no adquiridas (aquellas que son generadas por una computador).

Estas últimas las podemos dividir a su vez en fotorrealistas y no fotorrealistas.

Las imágenes fotorrealistas pretenden representar lo más fielmente posible la realidad, de tal forma que al observarles sea difícil diferenciarles de una imagen fotográfica.

Las imágenes no-fotorrealista pretenden lograr una representación más parecida al arte de una pintura o de una artesanía.

Para la adquisición de una imagen, por medio de una cámara fotográfica digital, la imagen pasa a través de una serie de lentes y mecanismos hasta llegar a un dispositivo CCD (siglas en inglés del *charge-coupled device*: 'dispositivo de cargas [eléctricas] interconectadas'), el cual no es más que un sensor con diminutas células fotoeléctricas que registran la imagen.

El número de celdas fotoeléctricas con que cuenta uno de estos sensores se expresa como “número de píxeles” y se le denomina resolución.

Simulando el proceso descrito para el ojo humano, cada píxel registra 3 colores diferentes (Red-Green-Blue: Rojo-Verde-Azul). Para conseguir esta separación de los colores, la mayoría de cámaras CCD utilizan una máscara de Bayer que proporciona una trama para cada conjunto de cuatro píxeles, de forma que un píxel registra luz roja, otro luz azul y dos píxeles se reservan para la luz verde.

De allí la imagen es procesada por la cámara. Implícito en este procedimiento hay dos procedimientos denominados muestreo (sampling) y cuantización (quantization).

Muestreo

Como se mencionó anteriormente el espectro electromagnético es una señal continua (por ende la luz visible también lo es).

La imagen percibida por el ojo o por la cámara, puede ser expresada como una función continua, a la que podemos denominar $f(n)$, siendo n el sensor muestreado y $f(.)$ el nivel de intensidad percibido por el n -ésimo sensor, sin importar cual sea el tipo de interacción con la señal de la que estemos hablando (esto es: reflexión, emisión o absorción).

Como no contamos con un número infinito de sensores la función continua $f(n)$ debe ser mapeada a una función discreta, lo que se logra mediante el muestreo.

En procesamiento digital de señales, el muestreo consiste en tomar un número discreto de muestras que representen lo mejor posible a la señal continua. En la figura 6 podemos

ver como una señal continua (a) es muestreada en diferentes tiempos, buscando que la señal discreta (muestreada) (b) sea lo más parecida a la señal de entrada, con el requisito de calidad de que “se vea bien”, esto es, que nos dé una idea más o menos exacta de la forma que tenía la señal original.

Es fácil ver que el muestreo describe colectivamente -a grosso modo la forma de la señal, y que las variaciones pequeñas se pierden- y las estructuras son más difíciles de discernir.

En el extremo izquierdo de la figura 6 b) se observa cómo cada punto de muestreo recae exactamente en un valor discreto (sensor 0, 3, 4, 5, etc.). Una forma natural de registrarle es mediante un arreglo bidimensional, normalmente en un formato fila-columna.

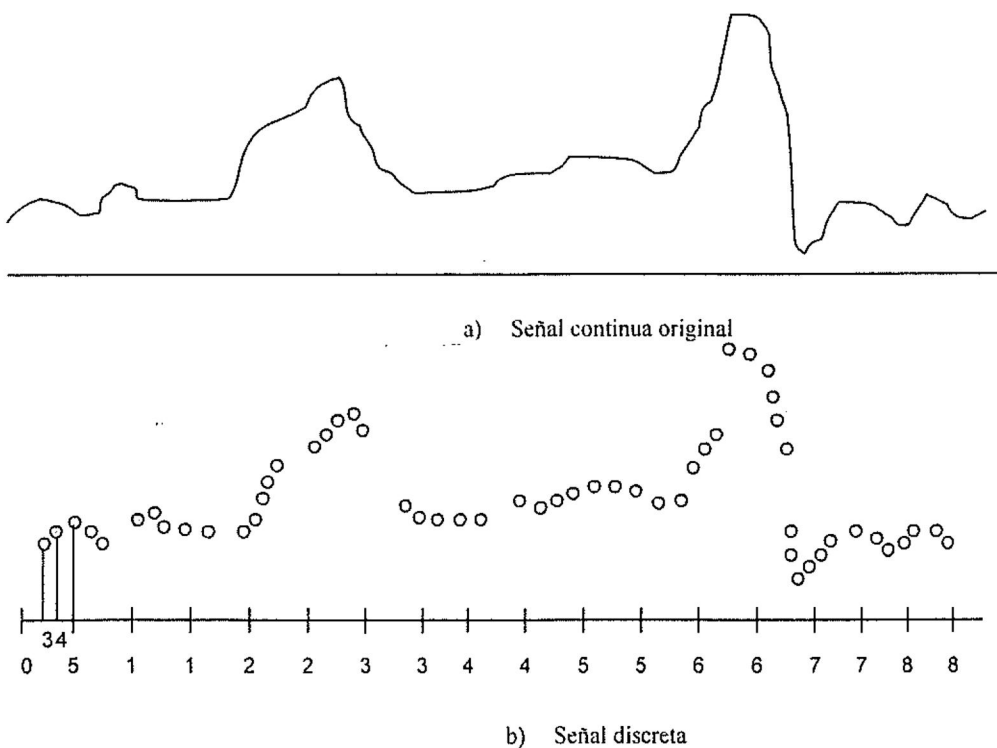


Figura 6 Resultado del muestreo (b) de una señal continua original (a).

De hecho el dispositivo CCD, está formando por una matriz (generalmente rectangular) de sensores electromagnéticos de dimensión $M \times N$.

Al respetar el formato en que se reciben las señales, para su registro, lo que estamos logrando es re expresar la función $f(.)$ en términos del dominio espacial del dispositivo CCD, esto es, a una forma $f(x,y)$.

El número de filas y columnas en una imagen muestreada frecuentemente se selecciona de una potencia de 2, dado que esto simplifica el cálculo de las direcciones computacionales de las muestras.

Entonces lo que se obtiene es una malla cartesiana bidimensional, como la representada por la figura 7.

En esta malla la posición de la intersección de los ejes se encuentra en el extremo superior izquierdo de la matriz.

Utilizándose únicamente valores enteros positivos para referir las posiciones en la malla, en términos matemáticos solo hacemos uso del primer cuadrante del sistema cartesiano, con la salvedad de que dicho cuadrante se expresa con el traslado de ejes mostrado en la figura 7. Así las x corresponden a las columnas y las y a las filas.

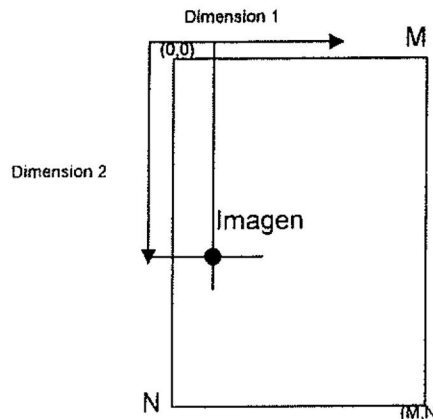


Figura 7 Dimensionalidad de una imagen. Esta representación corresponde al dominio espacial de la imagen muestreada en términos de $f(x,y)$.

II.3 Cuantización

Ahora bien los valores registrados por cada uno de los sensores corresponden generalmente a la intensidad con la cual incidió la señal en el sensor. Dichos valores son positivos y corresponden al conteo de los fotones que inciden en el foto sensor.

La representación de los píxeles puede ser a un valor único, este caso generalmente corresponde a una escala de grises; o puede ser multivaluado, en el caso de las imágenes de color, registrando los sensores en tres cantidades separadas, que son los tres canales RGB.

Como ya se mencionó estos valores generados por la función $f(.)$ son continuos, lo que se necesita hacer es mapear dichos valores a valores discretos, en otras palabras el rango de f será discreto y su límite deberá ser establecido de antemano.

Esto realizado de forma ordinaria se obtiene mediante un proceso de redondeo, truncamiento o algún otro proceso no lineal irreversible de destrucción de información.

La cuantización es necesariamente un precursor del procesamiento digital, dado que las intensidades de la imagen deben ser representadas con una precisión finita (limitadas por el tamaño de 1 byte) en cualquier procesador digital.

A diferencia del muestreo el proceso de cuantización no es lineal, lo que lo vuelve extremadamente difícil al analizar la señal, por ello en muchos tratados teóricos se presentan la información como si ésta no estuviese cuantizada, con el fin de facilitar el análisis.

Igualmente tampoco existe una medida matemática simple de pérdida de información debido a la cuantización. Lo que si es que sus efectos visuales son fáciles de apreciar, por ejemplo bajo ciertas cuantizaciones bajas puede llegar a parecer un efecto que es conocido como "contorno falso" esto es cuando la imagen toma la apariencia de estar formada por rayas, imaginemos a un huevo que al ser representado apareciera como si se tratara de un huevo de pascua. Esto es resultado de una resolución inadecuada en la escala de grises que está siendo utilizada para representar regiones de variación sutil de una imagen. En tales regiones los efectos de un nivel de gris (cuantizado) inadecuado puede ser visualmente exagerado, lo que nos llevará a la aparición de estructuras falsas (no pertenecientes a la imagen de entrada).

Otro efecto bastante conocido es el "aliasing" o aparición de alias, el cual es el resultado de un muestreo bajo y tiene como efecto que las frecuencias de la imagen aparezcan sin un significado físico (en este caso, creando un patrón falso).

Entonces se vuelve necesario un criterio que determine de que calidad es una cuantización.

Criterios de evaluación

Dada una imagen $f(n_1, n_2)$ de tamaño N_1 por N_2 , donde cada píxel toma un valor en el rango $\{0, 1, \dots, D - 1\}$ entonces se puede establecer que un criterio de no distorsión para la cuantización sería cuantificar cada píxel de la imagen a k niveles de salida ($k < D$) tales que el criterio de distorsión sea minimizado.

Como se ha establecido el error es producto de la cuantización $i = Q(S_i)$.

A la forma $y = Q(x)$ se le denomina un cuantificador escalar. Un cuantificador escalar de k -niveles se especifica por $k + 1$ niveles de decisión (ver figura 8)

$$(d_0, d_1, \dots, d_k)$$

y k niveles de salida

$$(y_1, y_2, \dots, y_k)$$

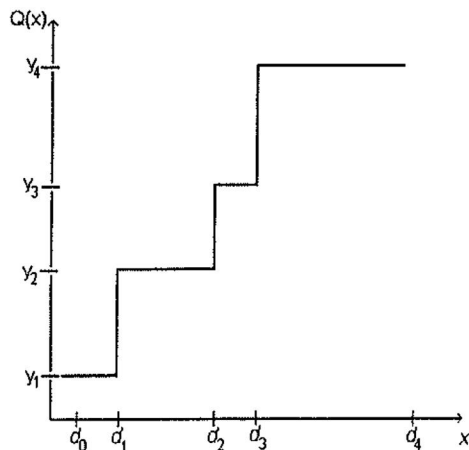


Figura 8. A cada región que va de d_{i-1} a d_i se le denomina un bin de cuantización.

Así la función de cuantización queda

$$Q(u) = y_i, \quad \text{sí } d_{i-1} \leq u \leq d_i, \quad i = 1, 2, \dots, k$$

El error que existe entre el valor cuantizado y el valor original se define como

$$e(u) = Q(u) - u$$

Si evaluamos el momento de enésimo orden (e.g la esperanza matemática) de esta variable aleatoria obtenemos

$$M_n = E(x^n)$$

$$M_n = \int ((e(x)^n)) P_x(x) dx$$

$$= \sum_{i=1}^k \int_{d_{i-1}}^{d_i} (y_i - x)^n P_x(x) dx$$

Un criterio de evaluación de la imagen de salida es el momento de segundo orden de $e(x)$.

$$M_2 = \sum_{i=1}^k \int_{d_{i-1}}^{d_i} (y_i - x)^2 P_X(x) dx - 0$$

Al cual también se le denomina MSQE (siglas en inglés de Mean Squared Quantizer Error).

En secciones posteriores se retomarán estos criterios que son la base para alcanzar un método que minimice el error o métodos óptimos desde el enfoque del MSQE.

Uso de métodos Halftoning² en el Stippling

III.1 Orígenes del Stippling Computacional

La generación de imágenes con Stipples es un método básico, fundamento de otros métodos que generan imágenes No Fotorrealistas.

Un referente indirecto para entender el Stippling es la técnica pictórica denominada Puntillismo (ver figura 9), la cual aparece por primera vez en 1888 y que cuenta entre sus principales representantes a George Seurat, entre otros.

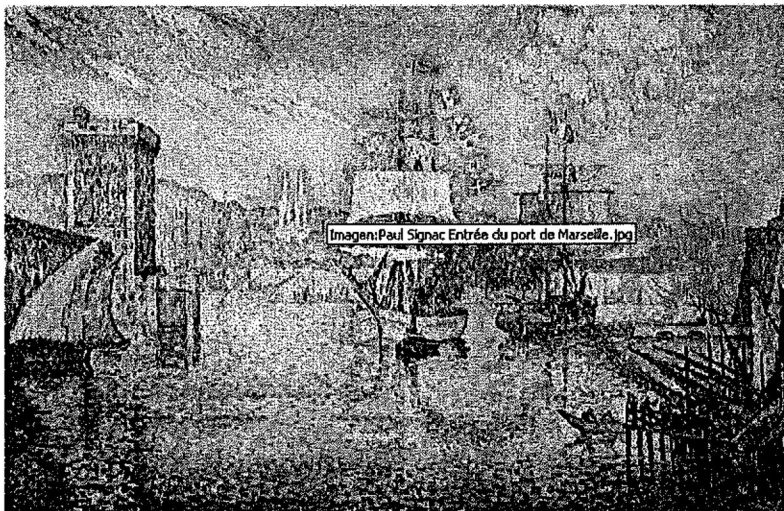


Figura 9. Pintura Puntillista (tomada de Wikipedia)

² Métodos de Tonos medios utilizado generalmente en la impresión

En términos muy simples la técnica consiste en colocar puntos de colores puros sobre el lienzo en vez de pinceladas.

Esta técnica deriva de los estudios cromáticos que llevó a cabo George Seurat, bajo la influencia de la obra Gramática de las Artes del Dibujo (publicado en 1865) de Charles Blanc. Sus experimentos físicos habían probado que la mezcla de colores los ensucia y desemboca finalmente en el negro. Por ello, la única mezcla capaz de producir el efecto deseado es la mezcla óptica.

Mediante la adopción de minúsculas pinceladas en forma de puntos, logran acumular, incluso sobre superficies reducidas, una gran variedad de colores y tonos cada uno de los cuales se corresponde con uno de los elementos que contribuyen a la apariencia del objeto. A una distancia determinada esas partículas diminutas se mezclan ópticamente y esa combinación produce una gama de colores mucho mayor que cualquier mezcla de pigmentos.

Como ya se mencionó el puntillismo solo se puede considerar un referente del Stippling. Sin embargo un proceso similar, el cual si es un antecedente directo, es el proceso Halftoning, el cual se utiliza en la impresión de libros y revistas.

Los métodos Halftoning nacen inicialmente de la necesidad de representar una gran cantidad de intensidades utilizando únicamente dos: el blanco y el negro.

El problema a resolver entonces se puede plantear de esta forma: ¿cómo representar una gran cantidad de tonos utilizando un dispositivo que solo puede representar dos?

Al parecer se trataría de un problema diferente al puntillismo, y sin embargo, es similar pero planteado con la luz acromática.

Para poder comprender las similitudes y diferencias, primero se debe establecer que el color trabaja dos diferentes tipos de luz: cromática y acromática.

Cuando se habla del componente cromático, se hace referencia a diferentes atributos: matiz o color puro, saturación o pureza, etc.

Por otro lado, la luz acromática tiene un solo atributo: la cantidad de luz percibida.

Este atributo puede ser expresado de dos formas que están muy relacionadas pero que en esencia son diferentes (Foley, et al. 1996):

- Intensidad o Luminancia: analizado en el sentido físico de la intensidad percibida.
- Brillantez: analizado en el sentido psicológico de la intensidad percibida.

De hecho el componente acromático es más importante en la percepción humana que la información de color, pues define, según Margaret Livingston (Livingstone 2002), la forma, textura y línea.

En términos generales lo que tiene en común el Puntillismo y el Halftoning es el principio el cual establece que dado un conglomerado de puntos percibidos a cierta distancia, el ojo les une formando una tonalidad diferente mediante un mecanismo del ojo humano denominado *Integración Espacial*. Consistente en que si vemos un área pequeña desde una distancia suficiente, nuestros ojos promedian el detalle fino dentro del área y registran únicamente la intensidad global de la misma.

El ojo humano tiene la característica importante de ser sensible a razones de niveles de intensidad, no a sus valores absolutos.

Al generar una imagen con un método halftoning, utilizando la escala de grises, se trabaja con el valor de intensidad subyacente en cada posición de la imagen, buscando que el resultado obtenido sea óptimo en términos de lo que registra el cerebro (aun cuando haya cierta pérdida con relación a la imagen original).

Se habla de una pérdida porque lo que se hace es una representación análoga (no exacta) de la intensidad de entrada.

El fenómeno de la Integración espacial es aprovechado en el Halftoning, donde cada unidad pequeña de resolución se imprime con círculos de tinta negra cuya área es proporcional a la negrura $1 - I$.

En términos generales existe una conciliación entre la resolución espacial y la resolución de intensidad.

Al resolver la resolución de intensidad la imagen se va formando espacialmente, sin embargo la representación no es exacta ya que aparece patrones no deseados, denominados *Alias*.

Al final decimos que existe una oscilación entre la resolución de intensidad y la resolución espacial.

Dichas oscilaciones están limitadas por nuestra agudeza visual (aproximadamente 1 minuto de arco con iluminación normal), la distancia a la que se observa la imagen y la resolución de puntos por pulgada del dispositivo gráfico.

Otro fenómeno producido es el aliasing resultado de un muestreo bajo, el cual tiene como efecto que las frecuencias de la imagen aparezcan sin un significado físico (en este caso, creando un patrón falso). Este efecto se verá al presentar imágenes recreadas con matrices de oscilación de orden bajo, siendo el ejemplo más representativo la binarización (ver más adelante).

En Halftoning la intensidad subyacente en una cierta área de la imagen, se representa mediante la aglomeración de puntos, donde el tamaño del punto y el espaciado entre los mismos, logra, mediante el principio de *Integración Espacial*, una representación análoga a la intensidad de entrada y ésta es percibida por el ojo como un nivel en la escala de grises.

Lo expuesto hasta el momento presupone que el arreglo de la imagen que se representa es menor al arreglo de píxeles del dispositivo de representación, de manera que pueden emplearse varios píxeles de representación para un píxel de la imagen.

Cuando se habla de la impresión halftoning realizada en una imprenta, el dispositivo de impresión está concebido a trabajar de forma natural cierto número limitado de representaciones de intensidad y hay un tamaño mínimo y máximo de la mancha de tinta.

El espaciado entre puntos se da por el desplazamiento de la cabeza de inyección.

Los dispositivos de salida gráfica pueden aproximar los círculos de área variable de la reproducción de medios tonos. Por ejemplo, se puede utilizar un área de 2x2 píxeles en una pantalla de dos niveles para producir 5 niveles de intensidad diferentes, a expensas de reducir la resolución espacial sobre cada eje. Los patrones presentados en la figura 10 se pueden usar para rellenar las áreas de 2x2 con el número de píxeles *encendidos* que sea proporcional a la intensidad deseada.

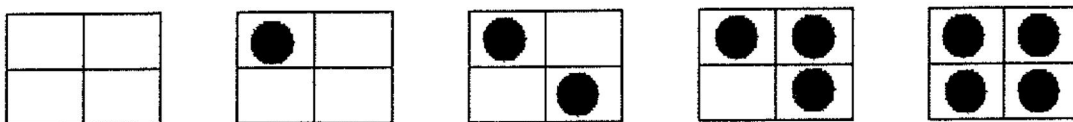


Figura 10. Cinco niveles de intensidad aproximada con una unidad de superficie de cuatro puntos

Los métodos computacionales buscan reducir el número de tonalidades para hacer compatible la imagen con el dispositivo y en el caso del NPR, para darle cierto carácter no fotorrealista.

Un primer ejemplo sencillo, que permite comprender los principios básicos es el procedimiento Dither.

III.2 Métodos Basados en la Geometría.

Características de una recreación no-fotorrealista

Antes de describir a profundidad el procedimiento Dither, hay que recordar que el propósito de la Recreación de Imágenes No Fotorrealistas no es lograr una imagen que sea una representación exacta de la realidad sino más bien similar a la recreación artística que haría un ser humano (artista o artesano).

En estos términos, las características que toda imagen no fotorrealista debe tener son:

- La correcta representación de la imagen en el dominio de las intensidades subyacentes
- La correcta representación espacial de la imagen mediante el uso del primitivo gráfico del método utilizado (e.g. puntos, mosaicos, cruces, etc), como se haría en la técnica artista que se está emulando, sin la aportación de nuevos patrones o alias no inherentes a la imagen de entrada.
- La similitud, lo más cercano posible, a una creación hecha a mano. En términos de ciertas imperfecciones que tienen todas las obras hechas a mano, y que una representación fotorrealista no tiene.

En el proceso del Dither el énfasis se centra en lograr el primer punto.

Cuando hacemos el Dither en un medio computacional el problema principal no radica en las limitaciones físicas del medio, sino más bien en querer emular dichas limitaciones durante el cálculo de la representación de las intensidades.

Reducir el número de puntos en la imagen de salida con relación a la imagen original, le da, de entrada, cierto carácter no fotorrealista.



Figura 11. (a) Imagen Original (b) Imagen Binarizada con un umbral de 128 (c) acercamiento

Binarización

Tal vez la forma más simple de generar una imagen Halftoning es la Binarización,

El método consiste en comparar el nivel de intensidad en cada píxel de la imagen con un valor de umbral .

Sí el valor de intensidad es mayor o igual al umbral entonces el píxel será negro en esa misma posición de salida, sino será blanco.

Este proceso lo podemos expresar como una función del tipo

$$f(I) = \begin{cases} 1, & \text{si } I \geq u \\ 0, & \text{en otro caso} \end{cases}$$

donde u es un valor preestablecido o puede ser la intensidad promedio de la imagen de entrada.

Las imágenes resultantes (figura 11 (b) y (c)) son representaciones limitadas de la imagen de entrada figura 11 (a), las áreas con intensidades similares aparecen como grandes plastas de un solo color, lo que hace que se pierdan parte de los contornos de la imagen y por consiguiente la forma de la misma.

El procedimiento Dither

En el caso del Dither lo que se tiene es cierto número de unidades pequeñas de resolución denominadas matrices dither o matrices de oscilación

La idea principal del procedimiento Dither es la misma de la binarización, y se diferencia en que no se tiene un solo umbral (u) sino una serie de umbrales acomodados en una determinada forma (Matriz Dither o Matriz de Oscilación).

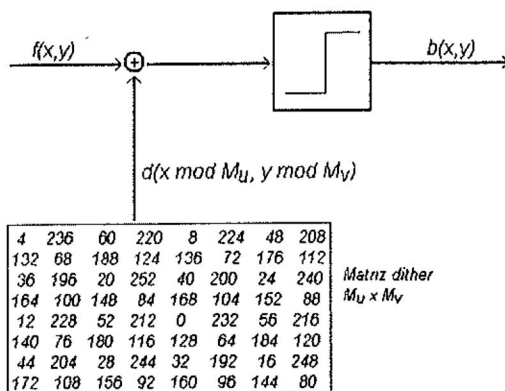


Figura 12. El dithering compara cada elemento de la imagen $f(x,y)$ con el umbral correspondiente de la matriz para colocar o no, un píxel negro en la imagen de salida

En la figura 12 se considera una imagen $f(x, y)$ de tamaño Nx por Ny con valores de píxel en el rango $\{0, 1, \dots, D - 1\}$.

Para generar la imagen de tonos medios $b(x, y)$ por este método, se usa el arreglo $d(u, v)$ de tamaño Mu por Mv llamado matriz dither (ver figura 13), donde los elementos son umbrales de cada nivel. Cada elemento de la matriz dither toma valores en el rango

$\{0, 1, \dots, D - 1\}$.

Para generar la salida se realiza la umbralización elemento por elemento de los píxeles de la imagen con los elementos de la matriz dither como se muestra en la **¡Error! No se encuentra el origen de la referencia.**, donde $u = x \bmod Mu$ y $v = y \bmod Mv$ (mod es el operador módulo). Y el elemento $d(u, v)$ de la matriz de oscilación es el umbral para el píxel (x, y) .

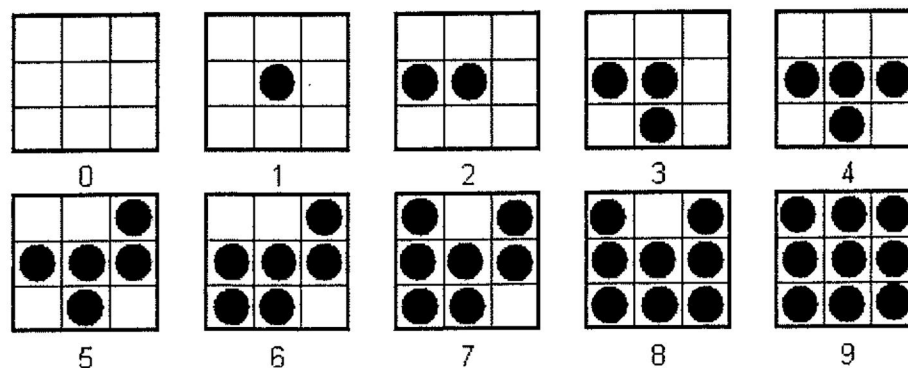


Figura 13. Patrones que representan las 10 intensidades de una matriz dither 3x3.

Dithering es una técnica para la aproximación de tonos medios auxiliar en la impresión y despliegue de imágenes con menores intensidades que la imagen original. Su función es minimizar el error producido por la acción de cuantizar las intensidades de entrada.

Dither es una aplicación intencional de una forma de ruido usando un error de cuantización aleatorio, lo que previene la aparición de patrones a gran escala.

El concepto de Dither viene de la Segunda Guerra Mundial durante la cual los aviones bombarderos llevaban a bordo ordenadores mecánicos, que se había demostrado tenían un mejor desempeño en vuelo que estando en tierra, esto se debía a que en vuelo las vibraciones que producía el mismo avión hacían que las partes mecánicas que estaban pegajosas vibraran y por consiguiente provocaba que se movieran continuamente mejorando el desempeño del aparato. Este "ruido" inducido por la vibración del vuelo, es la idea inicial de la representación Dither, las matrices no están ponderadas iguales para cada elemento $d(u, y)$ lo que induce el efecto oscilatorio de la vibración (Wikipedia 2008).

Ya se mencionó la aparición de patrones al utilizar los métodos halftoning de impresión (ver Ángulo de Pantalla en Capítulo II), dichos patrones son indeseables a menos que se busque estilización mediante un patrón intencional.

Al referir el tipo de Dither aplicado, se hace indicando el orden de la matriz de oscilación utilizada, por ejemplo Dither de orden 2.

En la generación de imágenes no fotorrealistas se utilizan generalmente procesos Dither de dimensiones 2x2 hasta 8x8.

En el caso de representaciones Fotorrealistas se utilizan matrices de orden mayor, ya que a mayor tamaño de la matriz utilizada hay una mayor fidelidad en la representación de salida.

$$D^{(3)} = \begin{bmatrix} 6 & 8 & 4 \\ 1 & 0 & 3 \\ 5 & 2 & 7 \end{bmatrix}$$

Ejemplo de una matriz dither 3x3 que corresponde al patrón presentado en la figura 13.

En general, se buscan matrices con los umbrales distribuidos de forma que disminuyan la aparición de patrones no inherentes a la imagen original. En figuras posteriores se mostrarán los resultados de la aplicación de diferentes matrices dither.

Algoritmo

A continuación se presenta el algoritmo para la generación de imágenes Dither con

matrices de tamaño nxn:

1. for y ← 0 to Ny
2. for x ← 0 to Nx
3. u ← x mod Nu; v ← y mod Nv
5. if (x,y) > D[u ,y]
6. b(x,y) ← 1
7. else
8. b(x,y) ← 0

Siendo $b(x,y)$ una imagen binarizada (imagen de salida), $f(x,y)$ la imagen de entrada,
 $D(u,v)$ la matriz dither de dimensiones $N_u \times N_v$

Si consideramos $N = N_x \times N_y$, el número total de pixeles en las imágenes (tanto de entrada como salida), el siguiente análisis nos dará los costos de ejecución del algoritmo:

Tabla I. Cálculo del orden de ejecución Dither.

INSTRUCCION	COSTO	VECES
for y \leftarrow 0 to Ny for x \leftarrow 0 to Nx	C_1	$N+1$
u \leftarrow x mod N_u ; v \leftarrow y mod N_v	C_2	N
if $f(x,y) > D[u,y]$	C_3	N
$b(x,y) \leftarrow 1$	C_4	N
else	C_5	N
$b(x,y) \leftarrow 0$	C_6	N

Por lo tanto el tiempo de ejecución es

$$T(N) = C_1(N+1) + N(C_2 + C_3 + C_4 + C_5 + C_6)$$

$$= a(N) + b$$

donde $a = C_1 + C_2 + C_3 + C_4 + C_5 + C_6$

$$b = C_1$$

lo que implica que el orden de ejecución del dither sea $O(n)$.

III.3 Difusión de error

El proceso de Difusión de Error

En 1975 Robert W. Floyd y Louis Steinberg desarrollaron uno de los algoritmos considerados clásicos dentro del cómputo gráfico.

De acuerdo a la forma en que un ser humano lee, desplazándose de izquierda a derecha y de arriba hacia abajo, crearon un kernel filtro que distribuye el error en la vecindad del píxel que se está trabajando (de acuerdo a la mecánica de la lectura).

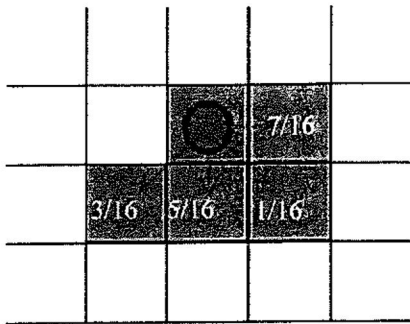


Figura 14. Kernel Filtro de Lloyd-Steinberg

En la figura 14 puede observarse como toman efecto las ponderaciones de error. A la derecha y abajo directamente del píxel que se está trabajando, están las ponderaciones mayores.

El método de Floyd-Steinberg se implementa fácilmente, produciendo generalmente imágenes de cambios más suaves y se le considera una opción recomendable para la generación de imágenes halftoning.

En su forma más simple, el algoritmo de difusión de error, como su nombre lo indica, distribuye el error entre los píxeles vecinos, lo que introduce una especie de suavizado en la imagen dither.

La diferencia entre el valor exacto del píxel de la imagen original y el valor de representación binaria (considerada como el error), se distribuye a través de los píxeles vecinos como cierta razón en cada dirección.

Existe muchas otras opciones a la difusión de error en el dithering, los cuales difieren principalmente en las fracciones que toma el término de error al ser distribuido entre los píxeles vecinos y el conjunto de píxeles que están siendo involucrados.

La suma de todas esas fracciones debe ser uno (en caso que dicha suma no diese uno, el resultado provocaría la aparición de artefactos visuales indeseables).

Algoritmo (Floyd-Steinberg)

1. for j=0 to H (S)
2. for i =0 to W (S)
3. $\Phi (i,j) = S (i,j) + E (i,j)$
4. $K=Q (\Phi (I, j))$
5. $e= \Phi (i, j)-K$
6. $E (i+1,j) += 7/16 e$
7. $E (i-1,j+1) += 3/16 e$
8. $E (i,j+1) += 5/16 e$
9. $E (i+1,j+1) += e /16$

De acuerdo a la implementación realizada, S es la imagen de entrada, $H(S)$ y $W(S)$ son sus dimensiones, $\Phi(\cdot)$ es la imagen halftoning de salida donde se sumaron todos los ajustes de error generados por el kernel píxel a píxel de la imagen de entrada. $Q(\cdot)$ es la función de cuantización que puede ser un función de binarización, dithering u otra. K toma los valores cuantizados que habrán de asignarse a la salida y E es el error acumulado de la cuantización.

Los costos de ejecución son los siguientes:

Tabla II. Costo de error de la difusión del error.

INSTRUCCIÓN	COSTO	VECES
for j:= 0 to W(S) for i:=0 to H(S)	C_1	$N+1$
$\Phi(i,j) = S(i,j) + E(i,j)$	C_2	N
$K=Q(\Phi(i,j))$	C_3	N
$e = \Phi(i,j) - K$	C_4	N
$E(i+1,j) += 7/16 e$	C_5	N
$E(i-1,j+1) += 3/16 e$	C_6	N
$E(i,j+1) += 5/16 e$	C_7	N
$E(i+1,j+1) += e/16$	C_8	N

Por lo tanto el tiempo de ejecución es

$$T(N) = C_1(N+1) + N(C_2 + C_3 + C_4 + C_5 + C_6 + C_7 + C_8)$$

$$= a(N) + b \quad \text{donde } a = C_2 + C_3 + C_4 + C_5 + C_6 + C_7 + C_8 \text{ y } b = C_1$$

lo que implica que la Dispersión de Error tiene un orden de ejecución $T(N) = O(n)$.

Es de notar que el algoritmo procesa la imagen de izquierda a derecha y de arriba hacia abajo de tal forma que los términos de error siempre se adicionan a los píxeles que todavía no han sido involucrados en el proceso del dither. Aunque hay modificaciones que alternan el orden de proceso, leyendo una línea de izquierda a derecha y la siguiente de derecha a izquierda.

III.4 Métodos Probabilísticos

En estos métodos la intensidad de cada posición de píxel de la imagen determina la probabilidad de que en esta posición se coloque un stipple. Entonces los grupos de stipples que conforman la imagen de salida son más densos en las áreas oscuras, y menos en las áreas claras.

Lo que se pretende es crear la imagen de salida como una serie discreta tono-espacio de primitivos gráficos. Esta condición intuitiva es la razón por la cual se puede tratar la imagen de entrada como una función de densidad de probabilidad (FDP) en 2D.

Formalmente una FDP describe la probabilidad diferencial que algún evento ocurra en cualquier punto. Si $q(x)$ es una FDP 2D, entonces la probabilidad de que algún evento ocurra dentro de la región infinitesimal dA en x es $q(x)dA$.

Es importante hacer notar que la probabilidad en un punto siempre es cero, por carecer de superficie, y por ello no hay sentido establecerla a este nivel. Únicamente este criterio tiene sentido en una región R .

En general la probabilidad de que un evento ocurra en alguna región R es

$$\int_R q(x) dA$$

El evento descrito por la *FDP* será la probabilidad de colocación del primitivo en esa ubicación.

Definiremos a la imagen de entrada como una FDP y entonces generaremos un conjunto de muestras distribuidas acorde a dicha FDP. Es deseable que las muestras se elijan de forma aleatoria distribución uniformemente.

Estas muestras son la posición de los primitivos con los cuales conformaremos el dibujo de salida.

Dichas muestras tienen un área finita. En esta área finita el traslape de los puntos no afecta, sin embargo por el esfuerzo computacional no es deseable. Así que hay que definir la FDP $q(x)$ tomando en cuenta el no traslape.

Dada $I(x_i)$ como la intensidad del i -ésimo píxel x_i , representado por un valor acotado (entre 0 y 1). En el proceso de dibujo de una imagen binaria $p(x_i)$ indicará la probabilidad de colocar un primitivo gráfico en la posición x_i que es igual a la intensidad en la imagen base. Sin embargo, si se utilizan primitivos gráficos de más de

un píxel (strokes, líneas o círculos), después de colocar N muestras de acuerdo a la función de densidad $q(x)$, la probabilidad de que una posición x_i no haya sido puesta es $(1 - q(x))^N$, entonces se requiere que

$$p(x_i) = 1 - (1 - q(x_i))^N \text{ o}$$

$$q(x_i) = 1 - \sqrt[N]{1 - p(x_i)}$$

$$= 1 - \sqrt[N]{1 - I(x_i)}$$

que satisface la definición de una función de densidad.

$$\sum_{i=1}^n q(x_i) = 1$$

$$\Rightarrow \sum_{i=1}^n 1 - \sqrt[N]{1 - I(x_i)} = 1$$

$$\Rightarrow \sum_{i=1}^n \sqrt[N]{1 - I(x_i)} = n - 1$$

Dando ambos N (el número total de primitivos a colocar) y $q(x_i)$ el valor modificado FDP para cada nivel de intensidad, podemos generar la posición de los primitivos y dibujar la salida.

De hecho no existe ningún conjunto particular de muestras finitas que puedan representar la FDP continua de una imagen de entrada.

Un conjunto de muestras S está distribuido de acuerdo a la FDP q si para cualquier región R , la fracción de muestras en R es igual a

Hay varios métodos estándar de generar muestras de acuerdo a una determinada FDP.

La distribución uniforme tiene la característica de ser directamente generada. además que su escalamiento y traslación es trivial.

Utilizando un método de transformación es posible generar transformaciones analíticas que mapeen los puntos desde a alguna otra distribución fácil de resolver.

El método del Rechazo

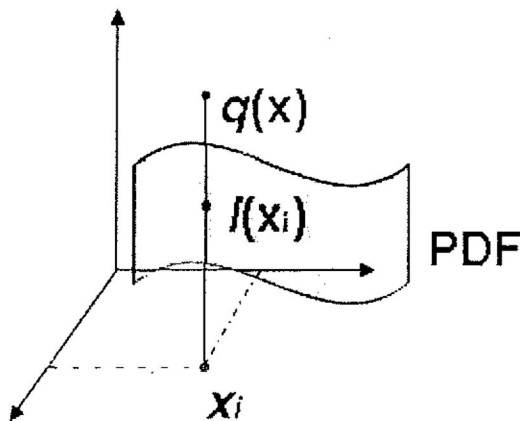


Figura 15. Generación de valores de salida que se calculan considerando la imagen como una FDP

La figura 15 muestra en forma gráfica la conceptualización del proceso. la imagen de entrada es la FDP representada por la superficie bidimensional de oscuridad. Se generan

muestras aleatorias con una distribución r donde generalmente se usa la distribución uniforme $U_{0,1}$ dentro del espacio de la imagen (e.g \mathbf{x}_i).

Después se genera un valor aleatorio de intensidad $U_{0,1}$ de la posición x , l , para compararle con el valor de intensidad de la imagen en esa misma posición $q(x)$, el cual funciona como un umbral.

Si el valor l es mayor o igual al umbral $q(x)$ entonces se pinta el punto.

Algoritmo.

1. mientras $|S_q| < N$ haz
2. muestra $x \sim r$
3. muestra $l \sim U_{0,1}$
4. si $q(x) / Mr(x) \leq l$ entonces
5. $S_q \leftarrow S_q \cup x$
6. fin si
7. fin mientras

La figura 16 a) se generó utilizando la distribución uniforme (como distribución r) de 200000 posiciones de las cuales solo 102200 fueron conservadas al considerar valores aleatorios superiores al umbral. La figura 16 b) tiene solo 4000 stipples de radio 3.

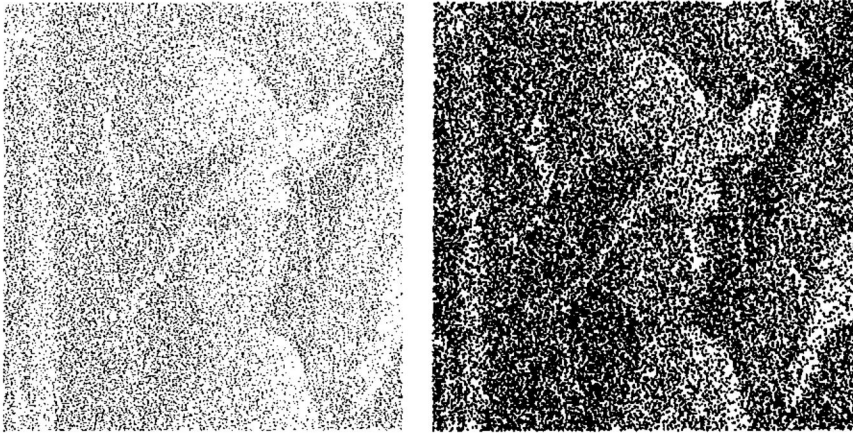


Figura 16. a) 200,000 posiciones b) con 4,000 stipples, en ambas imágenes los valores aleatorios para la colocación de puntos tienen una distribución Uniforme.

La distribución Poisson-Disk

Existen otras alternativas de distribución para muestrear las posiciones a umbralizarse, y que en el algoritmo definimos como muestras r -distribuidas. Entre ellas, la distribución Poisson-Disk ha tenido enorme difusión por las características de ruido azul que le hacen uno de los mejores patrones de muestreo para un rango amplio de aplicaciones (Lagae y Dutre 2006).

Particularmente Yellot (Yellot 1983) encontró en sus estudios que las células fotorreceptoras en la retina del ojo están esparcidas de acuerdo a la distribución Poisson disk, indicando con ello que este patrón de muestreo es efectivo para la percepción de las imágenes.

La distribución Poisson disk es una distribución uniforme de puntos 2D en la cual todos los puntos satisfacen la restricción de una distancia mínima. La mitad de esa distancia se conoce como radio de la distribución. Un conjunto de puntos tendrá una distribución

Poisson disk si se coloca un disco con el radio correspondiente en cada posición sin que se den traslapes entre los discos.

Para generar un conjunto de puntos con distribución Poisson disk, comúnmente se usa el algoritmo “lanzamiento de dardos” (Cook 1986). Este algoritmo genera un punto en posición aleatoria con distribución uniforme, y se rechaza si no satisface la condición de separación mínima con los puntos previamente generados. Este proceso continúa hasta que no se puedan agregar más puntos. Como puede advertirse, es un algoritmo complejo y difícil de controlar; pues en lugar de especificar el número de puntos necesarios, se proporciona el radio que la distribución habrá de observar, y por ello resulta difícil predecir cuál será el número total de puntos en la distribución. Y si el proceso se detiene de forma temprana, la densidad de los puntos no es uniforme.

McCool y Fime (McCool y Fiume 1992) propusieron una versión mejorada del algoritmo lanzamiento de dardos el cual se conoce como lanzamiento de dardos relajado. Los puntos se colocan inicialmente con un radio grande, y cuando no hay mas espacio para un mayor número de puntos, el radio se reduce en alguna razón.

Sin querer entrar a detalle en el tema, pues su tratado merece un apartado especial, fuera del alcance de este trabajo, y considerando una secuencia de posiciones pseudoaleatorias (que pueden calcularse con “lanzamiento de dardos relajado”, diagramas de voronoi centralizados sobre una superficie de densidad uniforme, u otros métodos), se utilizó el rechazo para crear la figura 16. Se advierten mejoras al considerar Poisson-disk sobre la distribución uniforme, y en la sección VI.2 se realiza con mayor detalle su análisis.

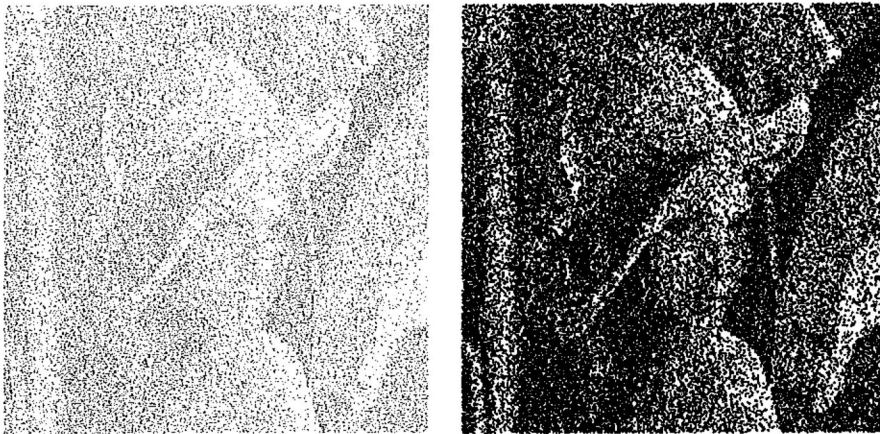


Figura 17. a) Ilustración con 100000 píxeles negros con distribución PD sobre la función de densidad formada por la imagen de entrada b) Imagen stippling con 40000 primitivos de radio 3.

En el capítulo Resultados (Capítulo VI Sección VI.1) la figura 38 presentan la misma imagen de la figura 17(b) pero aplicando un preprocesamiento para aumentar el contraste de la imagen de entrada, lo que provoca una mejora significativa en la preservación línea y figura de la imagen.

Orden de ejecución

Siendo $Q(\cdot)$ un función para el cambio en el rango de intensidades de acuerdo al orden de la matriz dither D .

Si tomamos a la imagen de entrada $S(N)$ como un arreglo unidimensional podemos facilitar nuestro cálculo

definiendo $N = W(S) \cdot H(S)$ siendo $W(\cdot)$ y $H(\cdot)$ el ancho y alto de la imagen de entrada respectivamente.

Obtenemos los siguientes costos de ejecución:

Tabla III. Tiempo de ejecución método del rechazo

INSTRUCCIÓN	COSTO	VECES
mientras $ S_q < N$ haz	C_1	$N+1$
muestra $x \sim r$	C_2	N
muestra $l \sim U_{0,1}$	C_3	N
Si $q(x)/Mr(x) \leq l$ entonces	C_4	N
$S_q \leftarrow S_q \cup x$	C_5	N

Por lo tanto el tiempo de ejecución es

$$T(N) = C_1(N+1) + N(C_2 + C_3 + C_4 + C_5)$$

$$= a(N) + b$$

donde $a = C_1 + C_2 + C_3 + C_4 + C_5$ y $b = C_1$

lo que implica $O(n)$.

Uso de Diagramas de Voronoi en el Stippling

IV.1 Características artísticas del Stippling

La técnica del Stippling realizada a mano por un artista es un proceso minucioso, que requiere de suma atención y de un trabajo paciente por parte de la persona que lo realiza. Lograr que un algoritmo obtenga resultados de calidad similar sin la intervención humana, implica conceptualizaciones complejas.

Sobre todo porque en el trabajo de la recreación de imágenes no fotorrealistas un elemento fundamental es lograr que, tanto la distribución espacial como el agrupamiento que se hace de los stipples, sean los óptimos emulando lo que el artista haría a mano, de tal forma que la imagen resultante tenga ese mismo carácter artístico (entre otros requisitos).

El Stippling, como técnica, originalmente se da por la necesidad de tener control sobre los procesos Halftoning utilizados al imprimir imágenes en libros; lo que en su momento fue una tarea nueva y difícil de realizar.

La técnica consiste de la colocación cuidadosa de muchos puntos (entiéndase como pequeños círculos) de tinta sobre el papel para aproximar diferentes tonalidades.

Los stipples se colocan lo suficientemente juntos para formar regiones oscuras y más separadas en las regiones luminosas.

Deben ser colocados en forma dispareja produciendo cierta aleatoriedad tal que el ojo humano no detecte patrones falsos que no sean parte de la imagen de entrada.

El stippling tiene cierto significado y mérito artístico independiente de su utilidad. Los stipples pueden representar detalles finos y texturas con un pequeño costo en la complejidad.

Una de las características de un buen dibujo de stipples es que no se formen plastas (no se amontonan unos encima de otros) de puntos de tinta, no dejar espacios desiguales, ni patrones no deseados. Los artistas logran esto mediante la cuidadosa colocación de cada stipple dentro de la imagen, utilizando lentes de aumento para poder observar cada punto en áreas muy aglomeradas, lo que explica porque elaborar un dibujo de estos se lleva varias semanas.

El método presentado utiliza una imagen de tonos de gris como entrada.

Para entender el papel que juega la imagen de entrada hay que establecer ciertas consideraciones relacionadas con la imagen.

Características de la imagen de entrada

La luz acromática tiene un solo atributo: la cantidad de luz percibida. Denominado intensidad.

Esta información subyace en cada punto de la imagen y es suficiente para la representación correcta de la imagen.



Figura 18. Detalle aumentado 13 veces, donde se observa las diferentes agrupaciones de píxeles que forman regiones oscuras y regiones claras.

Una imagen se define como un espacio discreto donde reside un conjunto (e.g. también discreto) de puntos denominados píxeles. En cada una de una de estas posiciones, subyace un valor de intensidad correspondiente a la escala de grises. Dicha escala va del 0 para el negro (ausencia total de color) hasta el 255 blanco (la máxima luminosidad de todos los colores).

Sin embargo, es de uso común tomar esta escala como sinónimo de densidad u oscuridad, en forma normalizada (escalada de 0 a 1), esto es:

$$i = \frac{I-255}{255}$$

Al examinar un área determinada de la imagen, figura 18, y ampliarle lo suficiente aparecen regiones claras, donde los píxeles tienen valores de densidad bajos; y oscuras donde los píxeles tienen valores de densidad altos.

Considérese a cada punto de la imagen un par ordenado x e y , que forma parte del dominio espacial de la imagen y de un sistema coordenado que lo representa.

Este sistema por convención considera su origen en el extremo superior izquierdo de la imagen. El eje de las x se desplaza, partiendo de este punto de origen, de izquierda a derecha de la imagen. Y el eje de las y se desplaza de arriba hacia abajo partiendo del mismo punto de origen.

Para ambos casos, las escalas de los ejes siempre son referidas con valores enteros positivos.

La información que proporciona un píxel está relacionada con su vecindario, por ello se puede razonablemente inferir la información de un píxel que reside dentro de una región dada: *el valor esperado de intensidad de ese píxel, debe ser cercano al valor promedio de la región.*

Lo cual permite definir al valor promedio de intensidad de una región dada, como la densidad representante de dicha región.

En el método presentado se pretende crear una imagen de salida como una serie de conjuntos discretos tonal-espaciales de primitivos gráficos. Los cuales responderán en su

distribución y en valor de intensidad a la imagen de entrada que funge como una función de densidad de probabilidad (FDP) dada a priori para ayudar en la recreación.

En la FDP $f(x, y)$ cada píxel $p(x, y)$ tendrá un valor de intensidad representado por la probabilidad $Px(x)$ de que dicho píxel (x, y) forme parte de la región A_i .

Así por definición la imagen de entrada está formada de k regiones A_i

$$A_1, A_2, \dots, A_k \quad i = 1, 2, \dots, k \quad | \quad A_i \neq A_j \text{ para toda } i \neq j$$

$$S = \cup A_i \quad | \quad A_i \cap A_j = \phi \text{ para toda } i \neq j$$

Lo que implica que cada región A_i de la imagen de entrada es mutuamente excluyente con relación a todas las demás regiones existentes.

En otros términos: un píxel dado solo puede pertenecer a una única región y nada más.

A continuación se presenta una forma de evaluar del muestreo o representación de espacios continuos con valores discretos mediante un proceso llamado cuantización.

Criterios de calidad de una cuantización

Un primer criterio es considerar el error (ver Capítulo III), en otras palabras que tanto difiere el valor de representación del valor original subyacente en ese mismo punto de la imagen.

El error es producto de la cuantización $i = Q(S_i)$.

Así la función de cuantización queda

$$Q(u) = y_i, \quad \text{sí } d_i - 1 \leq u \leq d_i, \quad i = 1, 2, \dots, k$$

El error que existe entre el valor cuantizado y el valor original se define como

$$e(u) = Q(u) - u$$

Si dada una imagen $f(n_1, n_2)$ de tamaño N_1 por N_2 , cada píxel toma un valor en el rango $\{0, 1, \dots, D - 1\}$ entonces se puede considerar que un criterio de no distorsión sería hacer la cuantización para cada píxel de la imagen a k niveles de salida ($k < D$) tales que el criterio de distorsión sea minimizado.

En el caso del MSQE la función es

$$M_2 = \sum_{i=1}^k \int_{d_{i-1}}^{d_i} (y_i - x)^2 P_X(x) dx \quad (1)$$

Entonces ¿Qué características debe tener un cuantizador óptimo?

Características de un cuantizador óptimo

El problema de encontrar un cuantizador óptimo fue resuelto por Lloyd a mediados de los 50s (Thomas y Stefan 2002)

Si x tiene un rango dinámico (α, β) , donde establecemos $d_0 = \alpha$ y $d_k = \beta$ con el propósito de generar una salida de todos los posibles valores que X puede tomar. Para encontrar el cuantizador óptimo primero debemos diferenciar la ecuación 1 con respecto a d_i con lo cual obtenemos:

$$(y_i - x_i)^2 - (y_{i+1} - x_i)^2 = 0$$

lo que es equivalente a

$$x_i = \frac{y_i + y_{i+1}}{2}, \quad i = 1, 2, \dots, k - 1$$

Esta ecuación es una condición necesaria para un cuantizador escalar óptimo.

A esta condición le denominamos de *punto medio* y considera que cualquier nivel de decisión entre dos bins de cuantización consecutivos es el punto medio de los dos niveles de salida adyacentes.

Segundo criterio: si diferenciamos la ecuación 1 con respecto a y_i ($i=1,2,\dots,k$) obtenemos

$$\int_{d_{i-1}}^{d_i} (y_i - x) P_X(x) dx = 0, \quad i = 1, 2, \dots, k$$

lo cual da

$$y_i = \frac{\int_{d_{i-1}}^{d_i} x P_X(x) dx}{\int_{d_{i-1}}^{d_i} P_X(x) dx}$$

$$= E [x \mid d_{i-1} \leq x < d_i], \quad i = 1, 2, \dots, k$$

Esta otra condición indica que el nivel de salida de cualquier bin de cuantización debe ser el valor esperado condicional (centroide) de X dentro del bin. A esta condición se le denomina la *condición centroidal*.

Ambas condiciones implican el análisis de las condiciones espaciales de cada pixel dentro del área A_i .

Ninguna de las dos condiciones son aseguradas con el método del rechazo. Ya que como se vio, en dicho método solo se toma la intensidad subyacente en la imagen de entrada, y de forma aleatoria se coloca o no el punto, únicamente basado en la información de intensidad.

Lo que implica que el método del rechazo no es un método óptimo para generar la imagen de salida.

IV.2 Método de Lloyd

El algoritmo de Lloyd utiliza ambos criterios de optimización y siempre converge en la práctica.

En el método de Lloyd equivale a los llamados Diagramas de Voronoi Centralizados.

Antes de describir a fondo el método de Lloyd, y el uso de los llamados diagramas de Voronoi dentro de él, es necesario ubicar cuales son los principios que nos permitirán identificar dentro de la imagen las diferentes regiones que la componen.

Generación de conglomerados

Instintivamente hemos definido las diferentes regiones de la imagen como conglomerados de puntos donde la densidad de los mismos está determinada por lo clara u oscura que es la región. De lo cual es representativo el valor promedio de intensidad dentro de dicha región A_i .

El proceso debe generar los conglomerados identificando grupos homogéneos de casos dentro de la población (e.g. los píxeles de la imagen).

Así el propósito del proceso será determinar el agrupamiento intrínseco en cada región en la imagen.

Al proceso de identificar grupos intrínsecos en un universo de datos se le denomina clusterización.

Entre las características que debe reunir un algoritmo de clusterización están:

- Escalabilidad
- Lidar con diferentes tipos de atributos
- Descubrir conglomerados de formas diferentes
- Minimizar los conocimientos del dominio para determinar los parámetros de entrada.
- Habilidad para lidiar con el ruido
- Intensidad al ordenar los datos de entrada
- Alta dimensionalidad

- Interpretabilidad y usabilidad

La problemática que presentan estos algoritmos es:

- Al tratar con un gran número de datos se vuelven problemáticos por el tiempo que requieren para procesar la información
- La efectividad del método depende de la métrica de similitud o “distancia” (la clusterización puede ser realizada como en este caso por distancia euclídiana)
- La definición obvia de distancia no existe, le debemos definir, lo cual no siempre es fácil

Un acercamiento llamado K-medias ayudará a comprender el planteamiento.

El método de K-medias

El método de k-medias es un método de clusterización no supervisado.

En este método se dan de entrada n prototipos que corresponderán en forma biunívoca con cada cluster a generar.

Algoritmo de K medias

1. Colocar los K puntos en el espacio a clusterizar. Estos puntos son el grupo inicial de prototipos
2. Se asigna cada objeto al grupo que tiene el prototipo más cercano

3. Cuando todos los objetos han sido asignados se recalculan las posiciones de los K prototipos cambiándose a la posición del punto medio ponderado.

Los pasos 2 y 3 se repiten hasta que los prototipos ya no se mueven. Esto produce una separación de los objetos en grupos para los cuales la métrica determinada se ha minimizado.

Cuando el método utiliza la distancia euclidiana como métrica de distancia, la posición nueva del prototipo corresponde al centroide o centro de masas.

$$\text{dist}(p, A_i) = \text{distancia} = \sqrt{(x - x_c)^2 + (y - y_c)^2}$$

Los centroides se definen como el promedio ponderado de las m_i (e.g. masa = intensidad en este caso) de las coordenadas de los puntos en A_i , entonces para un espacio discreto el cálculo sería

$$x_g = \frac{\sum m_i x_i}{\sum m_i} = \frac{m_1 x_1 + m_2 x_2 + \dots + m_n x_n}{m_1 + m_2 + \dots + m_n}$$

En el caso de una imagen consideramos a la FDP como un espacio continuo, lo que hace que use una ecuación equivalente

$$C_i = \frac{\int_A xp(x)dA}{\int_A p(x)dA} \quad (2)$$

Un centroide tiene las siguientes propiedades algebraicas:

- Homogeneidad: no cambia el centroide si se multiplica todas las masas por un factor $k \neq 0$
- Asociatividad: el centroide se puede calcular reagrupado puntos, es decir introduciendo centroides parciales

Por ejemplo, si $D = \text{centroide} \{(A, a), (B, b)\}$ (con $a + b \neq 0$) entonces $\text{centroide} \{(A, a), (B, b), (C, c)\} = \text{centroide} \{(D, a + b), (C, c)\}$ ($a + b + c \neq 0$)

El método de K medias procesado por fuerza bruta, tal y como se ha planteado, tiene el problema de que con un gran número de datos vuelve el tiempo de proceso prohibitivo.

Además de que los clusters se han establecido de antemano (se han inducido a priori).

El método de Lloyd equivale a K medias bajo la métrica de la distancia euclidiana o también a los llamados Diagramas de Voronoi Centralizados (CVT)

IV.3 Diagramas de Voronoi

Dado un espacio V y un conjunto de prototipos o sitios generadores, un diagrama Voronoi parte el espacio en regiones, donde cada región consiste de todos los puntos que están más cercanos a un sitio generador que a cualquier otro.

Al conjunto de sitios de entrada se denotan como x_1, x_2, \dots, x_k . Para cualquier punto p en el espacio V , $\text{dist}(p, x_i)$ denota la distancia desde el punto p hasta el sitio x_i .

El dominio de la región de x_i sobre x_j se define por

$$\text{Dom}(x_i, x_j) = \{p \mid \text{dist}(p, x_i) \leq \text{dist}(p, x_j)\}$$

Para un sitio x_i , la región Voronoi A_{ij} está definida por

$$A(x_i) = \bigcap_{j \neq i} \text{Dom}(x_i, x_j)$$

La partición del espacio V en $A(x_1), A(x_2), \dots, A(x_k)$ es un diagrama de Voronoi generalizado

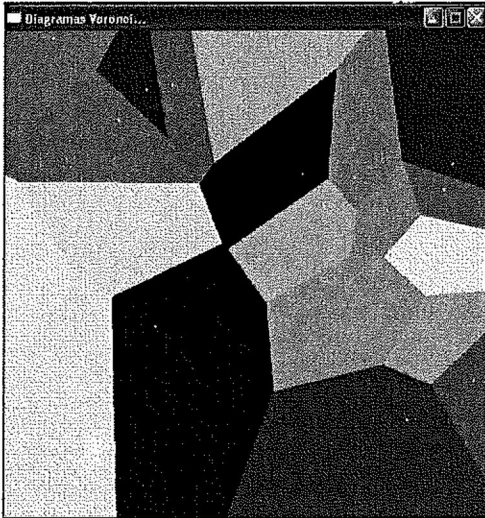


Figura 19. Diagrama de Voronoi producido con 20 sitios generadores, y un espacio V de 512 por 512 píxeles

El conjunto de puntos en el plano identificados con un i -ésimo sitio generador en particular se le denomina región de Voronoi $A(x_i)$ o A_i , en la figura 19 dichas regiones aparecen con diferentes colores y, podemos observar como cada región delimita con otra mediante líneas (fronteras) que les separan. Los puntos blancos que aparecen en la figura 19 corresponden a los sitios generadores de cada región, se observa como el conjunto de regiones Voronoi cubren el plano V por completo

Diagramas de Voronoi Centralizados

Los Diagramas de Voronoi Centralizados (CVT) son Diagramas de Voronoi cuyo conjunto de sitios generadores son al mismo tiempo sus centros de masa.

Un CVT es un caso especial de un Diagrama de Voronoi cuando $C_i = x_i \forall i$ donde c_i son los centros de masa y x_i son los sitios generadores.

Como ya se mencionó en el algoritmo de Lloyd se hace uso de las dos condiciones para un cuantizador óptimo: la condición de punto medio y la condición centroidal.

Algoritmo.

En 1982 Lloyd presentó su algoritmo determinístico:

1. Elección aleatoria de los k sitios generadores $\{x_i\}_{i=1}^k$
2. Determinar las regiones Voronoi $\{A_i\}_{i=1}^k$ correspondientes a $\{x_i\}_{i=1}^k$
3. Determinar los centros de masa $\{c_i\}_{i=1}^k$ correspondientes a cada region $\{A_i\}_{i=1}^k$
4. Hacer $x_i = c_i$ para $i=1, \dots, k$ y regresar al paso 2 hasta que converja

El método de fuerza bruta aplicado a la construcción de Diagramas de Voronoi tiene un orden de ejecución de $O(kn^2)$ donde n es la cantidad de píxeles en la imagen o función de densidad de entrada, y k el número de iteraciones lo cual es alto (de orden cúbico si consideramos $k \approx n$).

Hay algoritmos que calculan los Diagramas de Voronoi en $O(n \log n)$ sin embargo el método que a continuación se presenta tiene mejores tiempos de ejecución aunque es difícil describir formalmente su complejidad.

IV.4 Algoritmo basado en el hardware para gráficos

Otro enfoque interesante para construir diagramas de Voronoi (y diagramas de Voronoi Centralizados), considera a los puntos generadores como ápices de conos de altura uniforme que proyectan sus bases hasta el plano 2D xy .

Los conos tienen un mismo radio lo suficientemente grandes para provocar traslapes entre los conos. Las fronteras Voronoi aparecerán como los límites de intersección entre cada una de las bases circulares.

Cabe recordar que un cono, no es más que la proyección de círculos concéntricos, partiendo de un ápice a través del eje ortogonal z hasta llegar al plano xy colocado a una cierta distancia h . Dicha consideración asegura que cualquier punto ubicado dentro en la base del cono está a una distancia $d \leq r$ tal que no hay otro punto generador más cercano a él, que aquel que se le ha asignó. Esto se da siempre excepto en aquellos puntos ubicados en las fronteras, en cuyo caso se le puede asignar arbitrariamente una de las regiones que hacen frontera cuidando que esta asignación sea única.

Kenneth (Kenneth, et al. 1999) menciona al presentar su método, que fue inspirado en una anotación de la guía de programación de OpenGL 1.1. en la sección “Ahora que sabe cómo se hace” los autores discutían someramente un método simple para construir

diagramas de Voronoi 2D discretos para puntos, utilizando el hardware gráfico con OpenGL.

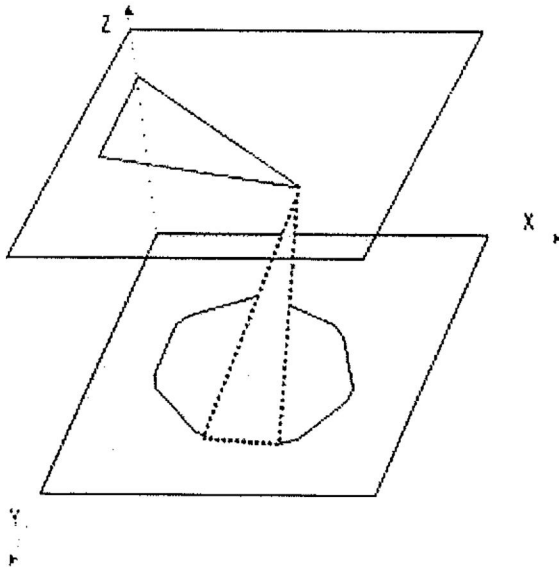
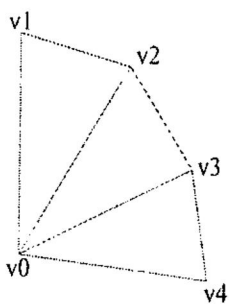


Figura 20. Delineado de la base circular de un cono utilizando la característica de interpolación del hardware de rasterización.

En la nota indica que para agilizar el proceso en lugar de conos, se utilizan pirámides que lo aproximen (por la gran cantidad de polígonos que necesita un cono bien dibujado). Ver figura 20.



GL_TRIANGLE_FAN

Figura 21. Funcionamiento de un Triangle Fan. (Tomado de (Shreiner, y otros s.f.))

En el caso de la implementación realizada, los cuerpos utilizados son pirámides y se hace uso de un tipo de primitivo gráfico de OpenGL denominado `Triangle_Fan` (figura 21).

Cuando se dibuja un prisma en OpenGL se proporcionan los vértices que le componen.

OpenGL permite diferentes tipos de especificaciones de dichos vértices, en este caso se utilizó una declaración en 3D (`glVertex3f`).

Cualquier tipo de prisma puede ser dibujado haciendo uso de éstas herramientas, si como en este caso lo que queremos es un cono, en vez de una pirámide regular, se aprovecha la facilidad de poder generar una circunferencia mediante segmentos de línea, si éstos son lo suficientemente pequeños como para que a los ojos del observador aparezcan como un una línea continua.

Dados los vértices de un triángulo, como podemos observar en la figura 21, el hardware de rasterización utiliza la interpolación lineal para calcular los valores de profundidad a lo largo de la superficie del triángulo.

A cada sitio se le asigna un ID único de color y el polígono regular proyectado (que aproxima un círculo), se recrea en ese color, en el caso de la figura 22 el ID de color asignado es el amarillo.

Cuando se regenera la escena con la proyección ortogonal al plano XY, el rasterizado OpenGL proyecta ortogonalmente los conos.

De esta forma cada píxel en el frame buffer tendrá el color correspondiente al sitio al cual está más cercano sin calcular de forma explícita la distancia para ese sitio.

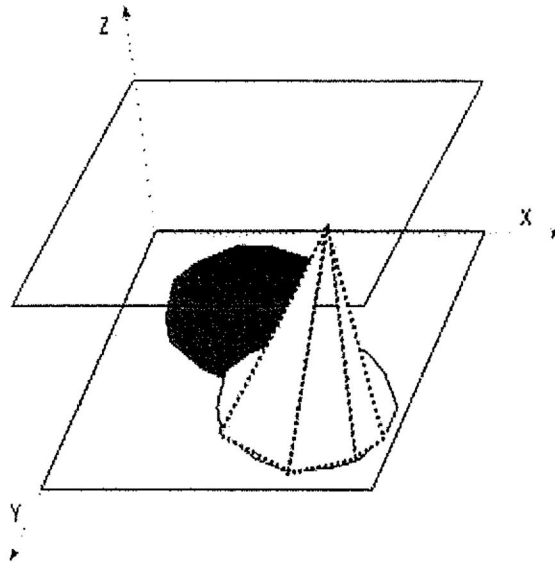


Figura 22. Formación de dos regiones Voronoi

La representación de los diagramas de Voronoi son regiones de colores índice que corresponden al conjunto de regiones de Voronoi.

En este acercamiento simplemente los límites se encuentran en el espacio entre muestras de píxeles de diferentes colores (figura 23).

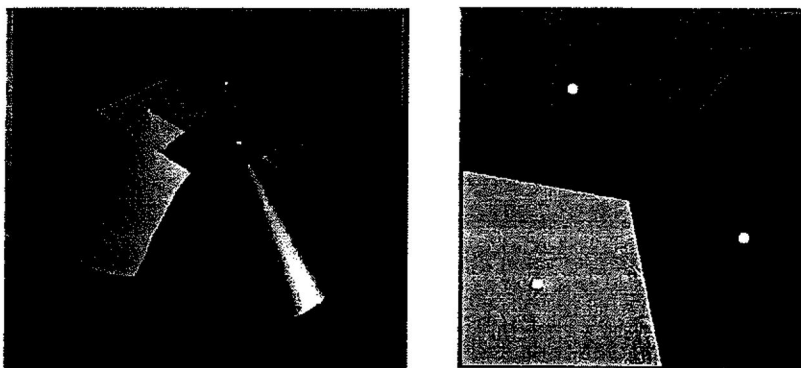


Figura 23. Construcción del diagrama de Voronoi mediante el dibujo de conos b) Vista ortogonal desde arriba.

En este algoritmo se dibuja un cono de diferente color con cada punto generador como ápice. Todos los conos tienen la misma altura y solo se diferencian por el color que los identifica y su posición $k(p_i, color_i)$, cuyo ápice corresponde a la posición del punto generador y su eje es ortogonal al plano xy .

Al generar una proyección ortogonal vista desde arriba, la intersección $k(p_i, color_i)$ y $k(p_j, color_j)$ forman un plano que corresponde a las regiones p_i y p_j (ver la figura 24).

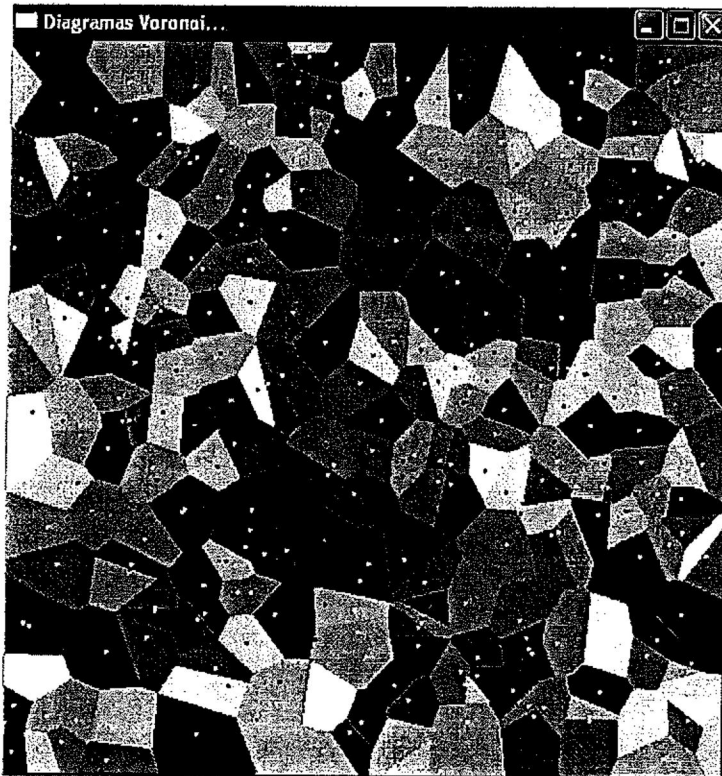


Figura 24. Primera etapa en el proceso de generación de una imagen con Stipples utilizando Voronoi.

Como ya se dijo, los CVT's son Diagramas de Voronoi cuyo conjunto de sitios generadores son al mismo tiempo sus centros de masa.

Podemos observar en la figura 24 la formación de un Diagrama de Voronoi con 300 sitios generadores.

Las áreas de colores corresponden a las regiones de Voronoi generadas en el plano XY calculados mediante la intersección de los conos sobre la GPU.

Cada región A_i se encuentra perfectamente delimitada por las superficies compuestas de píxeles de color i . A diferencia de otros métodos, el cálculo y la representación de dichas regiones no implican el uso de cálculos extremadamente complejos. Al contrario las fronteras aparecen simplemente por la proyección que se hace de los conos y sus intersecciones sobre el plano xy .

En la figura 24, el punto blanco y el negro que aparecen en cada región, corresponden al sitio generador y al centroide respectivamente.

En el siguiente paso, el algoritmo de Lloyd itera desplazando cada punto generador sobre el centroide correspondiente, y entonces procede a la regeneración de los conos y por ende de las nuevas regiones Voronoi correspondientes.

Se puede observar en la figura 25 el comportamiento de las regiones Voronoi a través de diferentes iteraciones del proceso, en las primeras se puede percibir como los puntos generadores y los centroides están separados.

En la iteración 50 (figura 25 d) prácticamente ya no hay movimiento de los puntos generadores.

Es en este momento que se ha logrado obtener un Diagrama de Voronoi Centralizado, en el cual los puntos proporcionan la posición de los Stipples con las características óptimas en la colocación de los mismos. Ver figura 26 con las imágenes stippling resultantes.

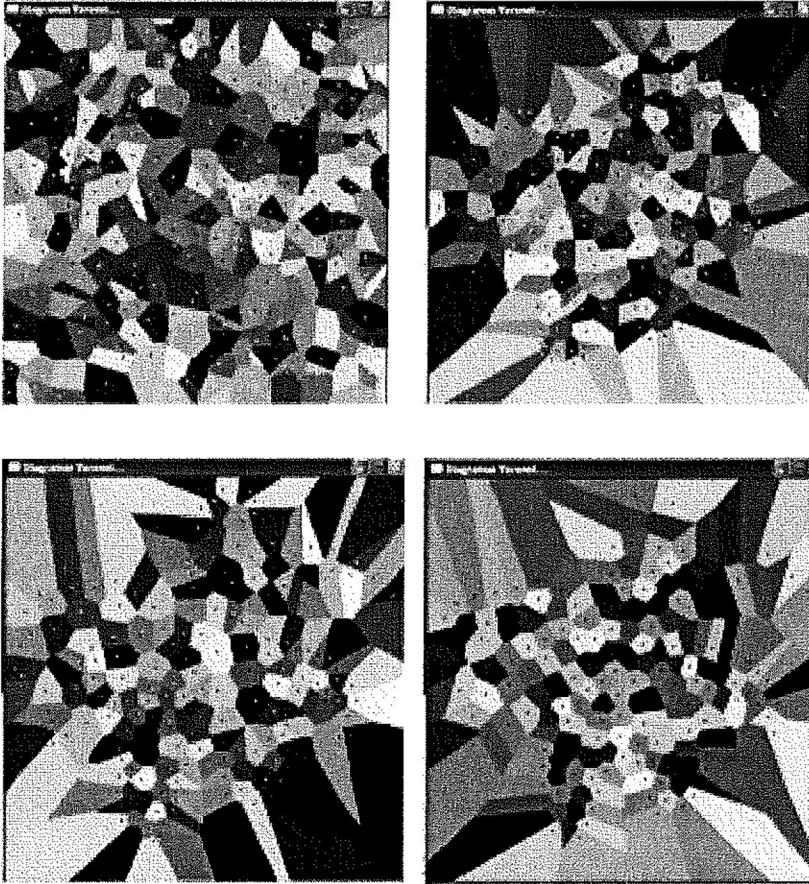


Figura 25. (a) Regiones de Voronoi originales, (b) primera iteración, (c) segunda iteración (d) iteración número 50 prácticamente todos los puntos generadores y los centroides han convergido.



Figura 26. Imágenes de la planta de maíz y Lena generadas con Stipples mediante el algoritmo de Lloyd.

En el método de Hoff Kenneth es difícil de determinar el costo computacional de la rasterización automática de los prismas, lo que si se puede determinar es que es mucho más rápido que los conocidos hasta el momento.

Uso de Agentes inteligentes (AI) en el Stippling

V.1 Un método generalizado

El Stippling forma parte de las técnicas artísticas que utilizan trazos para recrear una imagen.

Un trazo se define como un elemento gráfico mayor a un píxel, en el Stippling se trata de círculos los que correctamente colocados, y espaciados, van formando la imagen a recrear.

Sin embargo, este no es el único tipo de trazo existente. De los más comunes están: el Hatching, los Mosaicos, y las Pinceladas (entre otros). Ver figura 27.

El hatching son líneas cortas y largas, las cuales de acuerdo a su espaciamiento dan áreas luminosas u oscuras. La orientación de las líneas va siguiendo la geometría de la imagen de entrada.

Pueden ser de diferente tamaño, en pequeñas áreas llevan la misma orientación, pero se busca que no se formen patrones repetitivos que no pertenecen a la imagen de entrada.

Los mosaicos recrean la técnica artesanal de colocar pequeñas piezas de vidrio o lozetas de color, tales que el color del mosaico represente al color subyacente en la imagen de

entrada. La densidad o luminancia de la imagen permite establecer las zonas claras y oscuras de la misma.

La orientación de los mosaicos (tiles en inglés) conlleva dos propósitos principales: contribuir a la línea que define al objeto representado (contornos y bordes); y aportar la textura en una zona de la imagen.

Para lograr este propósito, la orientación “correcta” tiene implícita cierta complejidad que va desde el evitar la generación de patrones repetitivos, hasta conseguir efectos visualmente agradables mediante la aleatoriedad en la colocación de cada mosaico.

Las pinceladas, análogamente a las de un pincel, consisten en segmentos de curvas que inician de cierto grosor y al final del trazo se diluyen.

En las pinturas las pinceladas casi siempre cubren áreas de colores semejantes, de tal forma que una pincelada cubre un área que va desde un punto en la imagen de entrada con cierto color, hasta otro en el cual el color varía más allá de un umbral.

Los métodos analizados en los capítulos anteriores usando stipples pueden ser adaptados para manejar otro tipo de trazo. Sin embargo los métodos presentados hasta ahora no pueden, en una misma imagen, aplicar más de un tipo de trazo.

El poder combinar más de uno de ellos, permite enriquecer la recreación dando una variedad visual al dibujo obtenido, por ejemplo: realzando texturas en un área determinada del objeto representado; enfatizando detalles en la imagen de salida; o logrando un collage de técnicas que incrementan la calidad artística de la recreación obtenida.

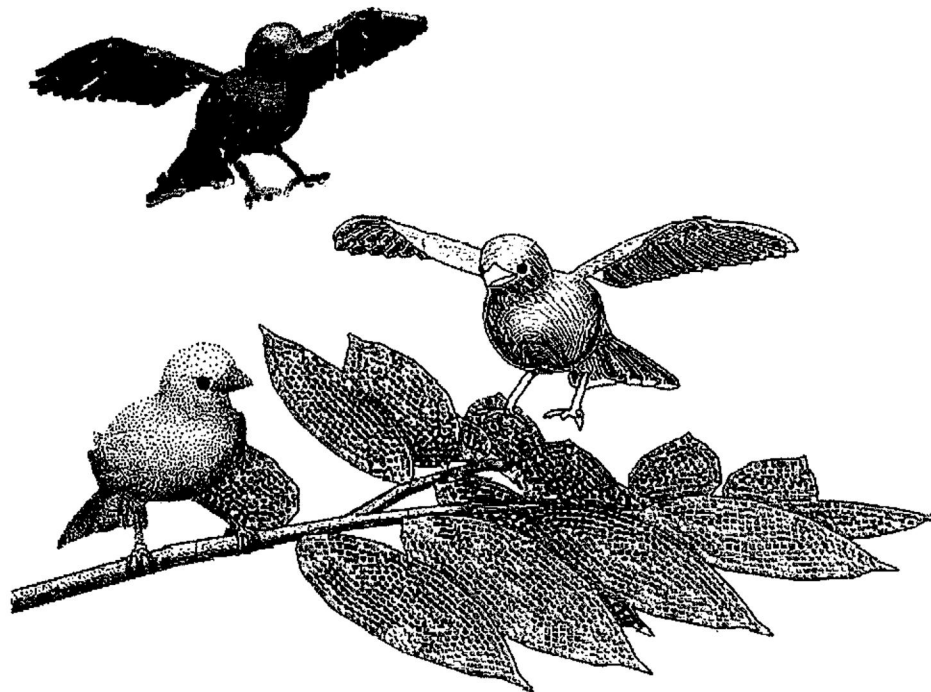


Figura 27. Múltiples trazos en una misma imagen NPR. Tomada de (Schlechtweg, Germer and Strothotte Volume 24 (2005), Number 2).

Schlechtweg et al (Schlechtweg, Germer and Strothotte Volume 24 (2005), Number 2) presentaron un método nuevo con un enfoque diferente.

A diferencia de los métodos hasta ahora vistos, el método de Schlechtweg no se basa en algoritmos que implementan una geometría específica ligada al tipo de trazo, ni tampoco se centra en la forma en que se van a distribuir los trazos para lograr cierta aleatoriedad en su posicionamiento o el muestreo de los mismos.

Aún cuando a fin de cuentas estos objetivos se cumplen, el método tiene su base en considerar a cada trazo como un ente independiente que responde a ciertas leyes propias dictadas por la técnica a emular.

Planteado de esta forma el problema, cada stipple se comporta como una partícula la cual interactúa con los demás stipples mediante leyes preestablecidas que rigen su comportamiento de un sistema de partículas.

V.2 Simulación de un Sistema de partículas mediante agentes inteligentes

Como antecedente directo del enfoque presentado, están la simulación de sistemas difusos, tales como aquellos que representan fuego, gases, líquidos, etc. Y los estudios realizados por Reynolds (Reynolds July 1987) sobre el comportamiento que mantienen las parvadas de pájaros al volar para evitar colisiones.

Aún cuando en este caso la mecánica y las leyes que controlan el comportamiento de los trazos son simples, la lógica y la forma en que se construyen dichos sistemas difusos funcionan perfectamente para el caso de la recreación de una imagen mediante stipples.

Considérese a una imagen de entrada S como un universo virtual compuesto por los píxeles que la conforman; estos píxeles contienen información tal como su posición, el valor de intensidad, los contornos, textura, etc.

En el enfoque completo presentado por Schlechtweg la imagen (el universo virtual) se puede descomponer en capas (layers en inglés) que contiene cierto tipo de información. Así podemos tener la capa que contiene los bordes, la profundidad, las distancias a los bordes, coordenadas de las texturas o los vectores normales.

Sin embargo en el stippling se pueden omitir la mayoría de estos layers y trabajar únicamente con tres: la imagen de entrada, el buffer temporal, y buffer de salida.

Los bots o agentes inteligentes nunca toman “conciencia” de la existencia de otros bots mediante una comunicación directa entre ellos, ni tampoco existe una entidad que controle su comportamiento.

En vez de la entidad de control, lo que existe es un buffer temporal cuya función es servir como un recipiente de una serie de mensajes que se van dejando por las partículas al interactuar con su media ambiente. En términos de un sistema de partículas, cada partícula tiene una masa y una fuerza que afecta a las demás partículas en su cercanía.



Figura 28. Layers de una imagen.

Cada stipple (trazo) considerado como una partícula autónoma de las demás, vive en ese medio ambiente con el cual interactúa.

A mayor información proporcionada mediante los layers (figura 28), habrá un enriquecimiento de la recreación lograda. El layer principal es la propia imagen de entrada.

El enfoque replantea la forma de trabajar los stipples con relación a las metodologías hasta ahora vistas.

Hay que definir las leyes que rigen el comportamiento de las partículas y encontrar un paradigma de programación que permita que cada trazo tenga un comportamiento autónomo que responda a las leyes establecidas.

Como en la naturaleza las leyes afectan a todas las partículas por igual, y las interacciones que hay entre ellas se deben al efecto de las fuerzas que los dominan.

El paradigma de programación que permite éste tipo de funcionamiento es llamado Orientado a Agentes Inteligentes.

Un Agente Inteligente es cualquier cosa que pueda ser vista como un ente que percibe su medioambiente a través de sensores y actúa en dicho ambiente mediante efectores. (Russell and Norving 1995) En este paradigma los AI son considerados SoftBots³.

En común con las demás metodologías se toman dos aspectos básicos que se deben conseguir mediante la aplicación de las leyes planteadas:

1. La correcta representación espacial de los elementos de línea de la imagen de entrada

³ Agentes autónomos o Software Bots

2. Y la correcta representación de las regiones oscuras y claras que componen la imagen.

V.3 Agentes Inteligentes (AI)

Para lograr el primero de los propósitos Schlechtweg propone el uso de un tipo especial de SoftBot encargado de detectar los bordes (contenido en el layer de bordes) y de dibujarlos en la imagen de salida.

El efecto que tienen los bordes en la recreación es de suma importancia, ya que delinear el objeto a representar, e influyen a los trazos cercanos estableciendo una distancia máxima a la que un trazo se puede colocar cercano a un borde.

En la consecución del segundo propósito, al igual que en los demás métodos, la aglomeración de stipples debe ser mayor en zonas oscuras y menor en zonas claras.

Los trazos también son representados por softbots, a los cuales en forma genérica se les denominan RenderBots.

En el caso particular los stipples son llamados StippleBots y su única función es pintar un único stipple y dejar en el buffer temporal rastros de su acción.

Las leyes que controlen su comportamiento deberán reflejar que la distancia que separa a los StippleBots es directamente proporcional a la densidad promedio del área representada.

Para lograr esto, cada StippleBot, al ser colocado en cierta ubicación, genera una fuerza de repulsión (con relación a los demás bots en su cercanía) de tal forma que esta fuerza

establece una distancia mínima entre bots de acuerdo a la densidad del área que se está recreando.

Cada StippleBot influye y es influenciado por los demás bots dentro de su vecindario. La dinámica que se sigue es similar a la de partículas cargadas dentro de un campo electromagnético, existe una distancia mínima de influencia de un bot alrededor de su posición actual. Este radio a su alrededor depende de su masa, en este caso es la razón de el área densidad que cubre el stipple (el área del círculo) entre la intensidad subyacente para dicha posición dentro de la imagen de entrada.

Cada bot deja en el buffer temporal un rastro que corresponde a la magnitud de su propia fuerza de repulsión. El punto donde está registrada la fuerza es la cola de un vector cuya cabeza es la posición del vector actual. Este vector determina la dirección y la magnitud es la cantidad escalar registrada por el bot dentro del vecindario.

La suma de estas fuerzas hace que el bot actual se desplace, este comportamiento continúa durante toda la simulación.

El concepto de agentes viene del campo de la Inteligencia Artificial. Un agente es una entidad que actúa autónomamente, “viviendo” en un mundo artificial y teniendo su propio comportamiento. Tessier (Tessier, Chaudron and Miller 2002) establece las siguientes propiedades para un agente, el cual es capaz de:

- Actuar de forma autónoma
- interactuar y comunicarse con otros agentes
- sentir su medioambiente y reaccionar a esas sensaciones
- actuar sin ningún sensor de entrada

- adaptarse a su medio ambiente
- cambiar su posición en el mundo artificial

De hecho se pueden clasificar a los AI como:

- Reactivos o Reflexivos
- Deliberativos
- Colaborativos

Un StippleBot corresponde a los llamados Agentes reactivos o reflexivos. Este tipo de agente responde a un modelo evento-condición-acción.

Entre sus características está el no contar de forma interna con un modelo del mundo que le rodea, responden de manera separada a los estímulos externos de la información disponible a sus sentidos en el medio ambiente.

Exhiben un comportamiento emergente, el cual es el resultado de la interacción de dichos agentes individuales, y solo comparten datos de nivel bajo.

En un Sistema de Múltiples Agentes Inteligentes (MAS), como el presentado por Schlechtweg, el funcionamiento de los IA se potencializa al ser la acción del conjunto de agentes, mayor a la de cada una de sus partes por separado.

Los agentes se agrupan en “enjambres” que trabajan juntos una cierta área de la imagen.

V.4 StippleBots

En el método presentado por Schlechtweg hay la posibilidad de generar diferentes tipos de Bots, a los cuales de forma genérica se les denomina RenderBots. Uno para cada técnica a representar, además de aquellos que trabajan con las orillas.

Siguiendo la lógica de la nomenclatura utilizado existen los EdgeBots, HatchingBots, MosaicBots y los PaintBots, además de los StippleBots.

Todos ellos pueden convivir en el mismo mundo virtual y trabajar de forma totalmente independiente los unos de los otros.

En este trabajo nos ocupamos de los StippleBots.

Todos los RenderBots conceptualmente son máquinas de estado finito. Mientras la recreación de la imagen no está terminada, los estados por los cuales pasa un StippleBot son:

1. Simular
2. Mover
3. Pintar

En el estado SIMULAR cada bot, primero, accede a su medio ambiente para obtener la información subyacente en cada píxel de la imagen de entrada.

Para establecer una distancia mínima entre cada stipple se utiliza un versión simplificada de los principios que Reynolds (Reynolds July 1987) los cuales establecen que “evitar colisiones es el impulso de evitar un camino que lleva a un impacto inminente”. Más específicamente se lleva a cabo lo que él denomina como evasión estática de colisiones,

la cual se basa únicamente en la posición relativa e ignora la velocidad de movimiento de la parvada.

En su caso el vecindario es una zona esférica definida por un radio y una exponencial, ya que trabaja con 3D, en el caso de las imágenes se trabaja en 2D y el vecindario solo se define por el radio.

En el siguiente paso el bot calculará el radio de búsqueda R que delimita el vecindario alrededor de él.

Cualquier otro bot que se haya colocado en el vecindario cercano al bot actual dejará en el buffer temporal una señal consistente en la magnitud de la fuerza de repulsión que repelerá el bot actual.

Es a través del buffer temporal que el bot actual podrá percibir si hay fuerzas que lo afectan dentro de su radio de búsqueda.

Una vez detectadas las fuerzas se hace el cálculo del vector resultante mediante la suma de las fuerzas dentro del vecindario.

La fuerza resultante impulsará al bot actual a moverse, pasando a su segundo estado **MOVER**.

Todo este proceso se repite, por lo cual se puede decir que un bot se mantiene oscilando hasta que las fuerzas a su alrededor ya no lo muevan o se de por terminada la simulación por parte del usuario.

Un **StippleBot** contiene los métodos para generar un stipple, los cuales difieren en algunos aspectos de los de los demás **RenderBot**.

El modelo completo además de los **RenderBots** cuenta con un Medioambiente (MA).

V.5 El Medio Ambiente (MA)

El Medio se encarga de presentar una interfaz de usuario donde éste puede dar la imagen de entrada. Además (si existe) permite cargar la información adicional de los demás layers (contornos, profundidad, etc.). Los layers son accesibles a través de un índice único que les identifica.

La imagen de entrada es el layer 0, el buffer temporal el layer 1 y el buffer de salida el 2. De tal forma que para acceder a un punto específico de un layer se hace a través de una tripleta $(x,y,IdLayer)$.

Cada píxel de la imagen contiene un valor de intensidad, si es $f(x_i)$ la función que proporciona el valor de intensidad en un píxel determinado, entonces, la suma de todos los valores de $f(x_i)$ es la llamada Densidad Acumulada.

Sea $S_{m \times n}$ la imagen de entrada, entonces

$$\text{densidad acumulada} = \sum_{i=1}^{m \times n} f(x_i)$$

Siendo cada stipple un área circular de color negro, por si sola dicha área representa cierta densidad de acuerdo al r (radio del stipple)

$$a_j = \text{número de píxeles por cada stipple}$$

ejemplo con un lado $r=2$

$$a = \pi * r^2 = 3.1416 * 2^2 = 12.57$$

Para facilitar la implementación se puede tomar (aún cuando el cálculo no es exacto) que el área cubierta por el stipple es cuadrada. Con la única consideración de que el valor de que r sea entero impar, esto es

$$a_j = r^2 \text{ donde } r \in \text{enteros impares}$$

Entonces el número idóneo que representen a la densidad acumulada, es

$$N = \text{número de stipples}$$

$$\text{densidad acumulada} \cong \sum_{j=1}^N a_j$$

visto de otra forma

$$a_j = \frac{\sum f(x_i)}{N}$$

lo que implica que

$$r = \sqrt{\frac{\sum f(x_i)}{N}} \text{ o } N = \sqrt{\frac{\sum f(x_i)}{r}}$$

Lo ideal es calcular N de acuerdo a un r fijo, y entonces dejar a r y N como constantes.

Una variante es la presentada por Schlechtweg quien para obtener una imagen de mayor calidad utilizó stipples de diferentes tamaños, con r de acuerdo a la intensidad promedio del área representada (mayores para áreas oscuras).

N debe asegurar una representación suficiente de la densidad acumulada de la imagen de entrada.

Para el cálculo de los valores de intensidad se toma el valor inverso de la escala de entrada, normalizando dicho valor a 1, esto es

$$f(x_j) = \frac{255 - I}{255} \therefore 0 \leq f(x_j) \leq 1$$

Ya con esta información el medio ambiente instancia los bots y los coloca en posiciones aleatorias uniformemente distribuidas a lo largo de la imagen de entrada.

Es importante asegurar la equiprobabilidad de todas las posiciones aleatorias por ello se utiliza una distribución $U_{0:mxn}$ para determinar donde nacen los stipples.

A partir de ese momento el medio ambiente cede el control a los bots quienes inician a trabajar de forma independiente, y el medioambiente queda latente debajo de los procesos de los bots.

Este proceso continúa hasta que el usuario lo interrumpe (situación que es detectada por el Medio Ambiente).

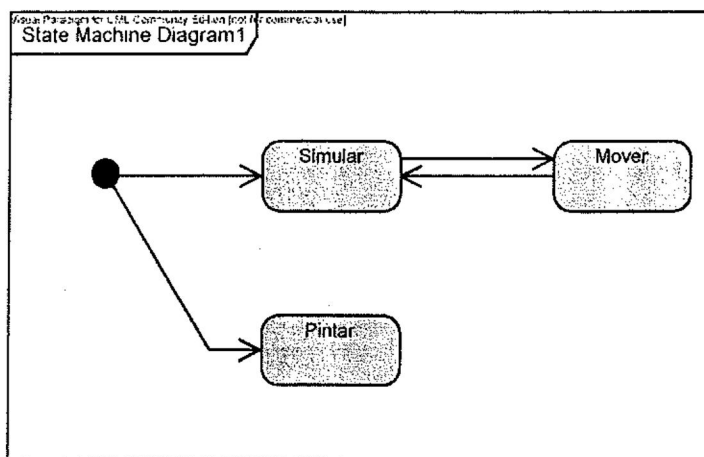


Figura 29. Diagrama de Estados

V.5 El Estado Simular

Como se observa en la figura 29, los bots al ser instanciados inician en el estado SIMULAR.

El bot nace en una posición aleatoria dentro de la imagen de entrada, tomando dicha posición (x, y) el siguiente paso es calcular el radio de búsqueda R .

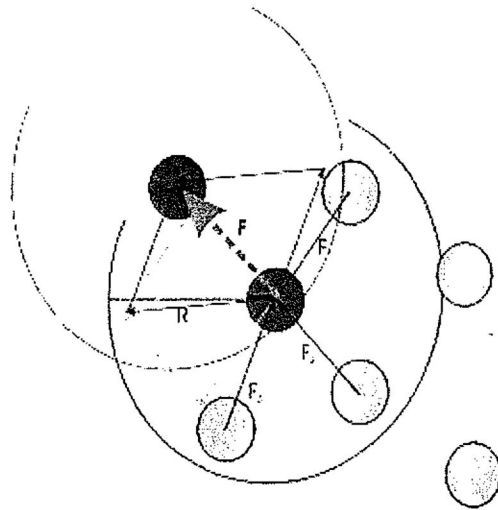


Figura 30. Fuerzas que afectan a un StipplingBot

En la figura 30 se observa como el bot actual (círculo azul) está colocado al centro de un área circular cuyo radio es R . A esta área se le denomina área de influencia y dentro de ella no debe existir ningún otro bot.

El propósito es el mismo que hemos visto en los demás métodos, representar cierta área de la imagen mediante el correcto espaciamiento de los stipples, de tal forma que la aglomeración de los mismos sea mayor en áreas oscuras y menor en claras

De tal forma que se cumple la siguiente razón:

$$i_j = \frac{\text{área del stipple}}{\text{área de búsqueda}} = \frac{\pi r^2}{\pi R^2} = \left(\frac{r}{R}\right)^2$$

Lo que nos dice que hay una equivalencia entre la intensidad en el j -ésimo píxel y la razón del área cubierta por el stipple entre el área de influencia del bot.

$$i_j = \left(\frac{255-I}{255}\right) \text{ donde } I \text{ es el valor de la intensidad del píxel sin normalizar}$$

$$\therefore R = \frac{r}{\sqrt{i_j}} \quad (3)$$

Hacer el cálculo con relación a un solo píxel no es conveniente, ya que si aleatoriamente se tratase de un píxel negro rodeado de un área blanca el valor calculado de R sería pequeño. Lo que haría que se afectara en forma mínima a los demás bots produciendo al final un área oscura en vez de una clara.

Otra forma de calcular R es tomando como vecindario un área que equivalga a a_i , esto es:

$$0 \leq f(x, y) \leq 1$$

$$d = \sum_{k=0}^{L^2} f(x', y') \mid (x', y') \in A$$

$$\text{donde } A_j = \cup(x', y') \mid x - W \leq x' \leq x + W, \quad y - H \leq y' \leq y + H$$

W es el ancho y H es el alto de una ventana cuadrada que enmarca el área A y $d \cong a$.

Entonces la intensidad en A es

$$i_j = \frac{d_j}{L^2 * 255}$$

En éste otro método el tomar la densidad promedio en A_j como la intensidad, evita el problema planteado.

El cálculo de R se hace aplicando la misma ecuación 2 pero con la intensidad promedio de A_j .

En ambos casos (por píxel o por área) si $i_j = 0$ la ecuación 2 se indefine. lo que implicaría una R infinito de búsqueda.

Por conveniencia

$i = 0 \rightarrow i = 0.00196$ (equivalente a tomar un área de un solo píxel con una intensidad de 0.5 en vez de cero en su forma normalizada $\frac{0.5}{255} = 0.00196$)

Entonces el valor máximo para el radio de búsqueda R en la región A_j se topa a

$$\max_{A_j}(R) = \frac{\text{radio stipple}}{\sqrt[2]{0.00196}}$$

Lo que impide que en un área totalmente blanca el radio de búsqueda se vuelva infinito.

El valor con que el bot afecta al medio ambiente a su alrededor es un escalar que representa la magnitud de la fuerza, y está dado por:

$$c = \|F_j\| = i * \max_{A_j}(i)$$

la razón de la intensidad promedio en A_j aplicada al máximo de influencia posible de un bot. En otras palabras la magnitud de la fuerza se re expresa escalándola al máximo valor producido por un área totalmente blanca.

Como se puede observar en la figura 30 todos los demás bots dentro del vecindario tienen su propia fuerza las cuales repelen al bot actual.

Para determinar cuales bot están en el vecindario del área de influencia del bot actual debemos establecer un criterio de distancia, que en este caso será la distancia euclidiana.

En el buffer temporal los bots han dejado asentada la magnitud de su fuerza, el bot actual tiene acceso a dicho bufer mediante la tripleta $(x_i, y_i, 2)$.

Sean B_1, B_2, \dots, B_n bots en el vecindario de B_i (el bot actual) tal que el conjunto de bots en la vecindad de B_i es:

$$V = \left\{ B_j \mid \forall (B_i, B_j) \text{ donde } \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \leq R, j = 0..n, j \neq i \right\}$$

Sí $V \neq \phi$ entonces cada elemento de $\{V\}$ es un vector con cola en B_j y cabeza es B_i con magnitud

$$\|V_j\| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}.$$

El vector V_j se puede re expresar como una combinación lineal del tipo

$$U = x_1 i + y_1 j$$

El j-ésimo vector unitario es

$$U_j = \left(\frac{|x_i - x_j|}{\|V_j\|}, \frac{|y_i - y_j|}{\|V_j\|} \right) = (x_{u_j}, y_{u_j})$$

A los componentes de este vector les podemos multiplicar por el escalar c , y así obtener un vector con la misma dirección que V_j pero con magnitud $\|F_j\|$. Componentes que sirven para obtener la fuerza resultante y por consiguiente la nueva posición del bot.

$$B'_i = (x_i - [c_1xu_1 + c_2xu_2 + \dots + c_nxu_n], y_i - [c_1yu_1 + c_2yu_2 + \dots + c_nyu_n])$$

En el juego de fuerzas que afectan al StippleBot actual no se toma en cuenta la fuerza de él mismo (esta debería ser una fuerza de atracción) lo cual contribuye a la estabilidad del movimiento de los bots.

V.5 El Estado Mover y Pintar

El estado MOVER simplemente implica la actualización de las coordenadas x,y dentro del bot actual. Con cada movimiento el StippleBot actual asienta la información de su fuerza en el buffer temporal.

Por fuerza bruta se continúa el proceso comparando cada StippleBot con los demás.

Como se puede observar en la figura 30 es el usuario puede indicar en cualquier momento el termino de la simulación con lo cual todos los datos asentados en el buffer temporal se pintan.

Dicho mensaje de entrada es detectado por el MA y entonces usando la información existente en el buffer temporal genera círculos

círculo(x_i, y_i, r) .

V.6 Algoritmo

El algoritmo original de Schlechtweg para el Sistema de Agentes Inteligentes implica la ejecución en paralelo de cada stippling bot, lo cual está más allá del alcance de la presente tesis, se homologó el proceso mediante un algoritmo lineal donde el cálculo es controlado por un ciclo iterativo que por fuerza bruta realiza los cálculos de la simulación.

generaStipple (imgIn, imgOut, N, rep)

1. Cargar imagen de entrada
2. **for** i=1 to N
3. **do** $Pos(bot_i) = U_{0,1}$
4. intensidad = IntensidadPromedio($boot_i$)
5. agente[i].R \leftarrow (1-intensidad) \times MxF
6. **for** k \leftarrow 1 to rep
7. **for** i \leftarrow 1 to N
8. **do** Simula (i, N)
9. Dibuja Stipple (agente[i].x, agente[i].y)
10. Graba ImgOut

En las líneas 2 a 3 se posicionan los stipples en N posiciones iniciales aleatorias. En la línea 4 se hace un cálculo de la intensidad promedio (normalizada) en base a un área predefinida (véase capítulo V), donde el centro de dicha área es el píxel actual.

La línea 5 hace el cálculo de del inverso del promedio obtenido, para que este quede expresado como 1 para el blanco y 0 para el negro. En la línea 6 se topa dicha intensidad a un máximo que puede alcanzar la fuerza de repulsión entre bots.

El *for* de la línea 7 controla el proceso a terminar después de cierto número de iteraciones (*rep*) definidas por el usuario.

Simula (agente, id, imgIn, N)

1. $c \leftarrow \text{ceil} (3 / \text{raíz} (1 - \text{agente}[\text{id}].\text{in}))$
2. **for** $i \leftarrow 1$ to N
3. **do** $\text{magnitud} \leftarrow \sqrt{(\text{agente}[\text{id}].x - \text{agente}[i].x)^2 + (\text{agente}[\text{id}].y - \text{agente}[i].y)^2}$
4. **if** $\text{magnitud} > 0$ then
5. **do** $xu \leftarrow (\text{agente}[\text{id}].x - \text{agente}[i].x) / \text{magnitud}$
6. $yu \leftarrow (\text{agente}[\text{id}].y - \text{agente}[i].y) / \text{magnitud}$
7. **if** $(\text{magnitud} \leq \text{agente}[\text{id}].R)$ then
8. $\text{sumaX} \leftarrow \text{sumaX} + (c * xu)$
9. $\text{sumaY} \leftarrow \text{sumaY} + (c * yu)$
10. $\text{agente}[\text{id}].x = \sum \text{sumaX}$ y $\text{agente}[\text{id}].y = \sum \text{sumaY}$
11. $\text{intensidad} = \text{IntensidadPromedio}(\text{boot}_i)$
12. $\text{agente}[i].R \leftarrow (1 - \text{intensidad}) \times MxF$
13. return

Cálculo del tiempo de ejecución:

Tabla IV. Tiempo de ejecución método Agentes

generaStipple

INSTRUCCIÓN	COSTO	VECES
1. Cargar imagen de entrada	C_1	1
2. for $i \leftarrow 1$ to N	C_2	N
3. do determinar la posición aleatoria del i 'ésimo StipplingBot	C_3	$N-1$
4. intensidad = Determinar IntensidadPromedio(imgIn, agente[i].x, agente[i].y, m, n)	C_4	Se considera un costo fijo, por lo tanto se toma como $N-1$
5. agente[i].in $\leftarrow 1 -$ intensidad	C_5	$N-1$
6. agente[i].R \leftarrow intensidad $\times M \times F$	C_6	$N-1$
7. for $i=1$ to N	C_7	N
8. do Simula (i, N)	$O(N)$	$N-1$
9. Dibuja Stipple (agente[i].x, agente[i].y)	C_9	$N-1$
10. Graba ImgOut	C_{10}	Se considera un costo fijo, por lo tanto se toma como 1

Simula

INSTRUCCIÓN	COSTO	VECES
1. $c \leftarrow \text{ceil} (3 / \text{raíz} (1 - \text{agente}[id].in))$	C_1	1
2. for $i \leftarrow 1$ to N	C_2	N
3. do magnitud \leftarrow determina	C_3	$N-1$

la distancia euclidiana entre agente[id] y agente[i]		
4. if magnitud > 0 then	C ₄	N-1
5. do xu ← (agente[id].x – agente[i].x) / magnitud	C ₅	N-1
7. yu ← (agente[id].y – agente[i].y) / magnitud	C ₆	N-1
8. if (magnitud ≤ agente[id].R) then	C ₇	N-1
9. sumaX ← sumaX + (c * xu)	C ₈	N-1
10. sumaY ← sumaY + (c * yu)	C ₉	N-1
14. actualiza las coordenadas x y y del agente[id] con las sumas calculadas	C ₁₀	N-1
15. Determinar IntensidadPromedio(imgIn, agente[id].x, agente[id].y, m, n)	C ₁₁	Se considera un costo fijo, por lo tanto se toma como 1
16. agente[id].in ← 1 – intensidad	C ₁₂	1
17. agente[id].R ← intensidad × MxF	C ₁₃	1
18. return	C ₁₄	1

Entonces tenemos que el costo total se calcula:

Para Simula:

$$\begin{aligned}
 T(n) &= C_1 + N(C_2) + (N-1)C_3 + (N-1)C_4 + (N-1)C_5 + (N-1)C_6 \\
 &\quad + (N-1)C_7 + (N-1)C_8 + (N-1)C_9 + C_{10} + C_{12} + C_{13} + C_{14} \\
 &= N(C_2 + C_3 + C_4 + C_5 + C_6 + C_7 + C_8 + C_9) + [C_1 - C_3 - C_4 - C_5 - C_6 - C_7 - C_8 \\
 &\quad - C_9 + C_{10} + C_{11} + C_{12} + C_{13} + C_{14}] \\
 &= an + b
 \end{aligned}$$

Para el tiempo total de generaStipple:

$$\begin{aligned}
 T(n) &= C_1 + N(C_2) + (N-1)C_3 + (N-1)C_4 + (N-1)C_5 + (N-1)C_6 + N(C_7) \\
 &\quad + (N-1)(O(N)) + (N-1)C_9 + C_{10} \\
 &= N(C_2 + C_3 + C_4 + C_5 + C_6 + C_7 + C_9) + [C_1 - C_3 - C_4 - C_5 - C_6 - C_9 + C_{10}] + \\
 &\quad k(N(O(N))) - kO(N) \\
 &= kn^2 + an + b
 \end{aligned}$$

$$T = O(kn^2) + an + C$$

De lo que se establece que el costo de ejecución total de éste algoritmo es de $O(kn^2)$

Comparativo de resultados

VI.1 Métodos Halftoning

Dithering



Figura 31. Imagen formada con una matriz dither 4x4 que induce patrones triangulares.

Aún cuando una imagen dither no reúne todas las características enlistadas para una imagen del alta calidad no fotorrealista (ver II.1 Introducción), sí se puede considerar una representación no fotorrealista. Sobre todo cuando la matriz dither induce patrones que pudieran considerarse “artísticos”. La figura 31 muestra un ejemplo de este tipo de patrones, utilizando la matriz dither que sigue:

$$\begin{pmatrix} 9 & 5 & 2 & 0 \\ 12 & 8 & 4 & 1 \\ 14 & 11 & 7 & 3 \\ 15 & 13 & 10 & 6 \end{pmatrix}$$

El método dithering cubre dos aspectos importantes de las recreaciones no fotorrealistas:

1. El método asegura la correcta representación espacial y tonal de la imagen subyacente, al grado que aun cuando se sabe diferente la imagen resultante, se reconoce perfectamente como la imagen original.

Esto lo podemos observar al comparar la imagen de entrada y la imagen dither figura 32. Definitivamente la figura 32 no es una representación exacta y recuerda un poco ciertos trabajos de arte de los años 70's.

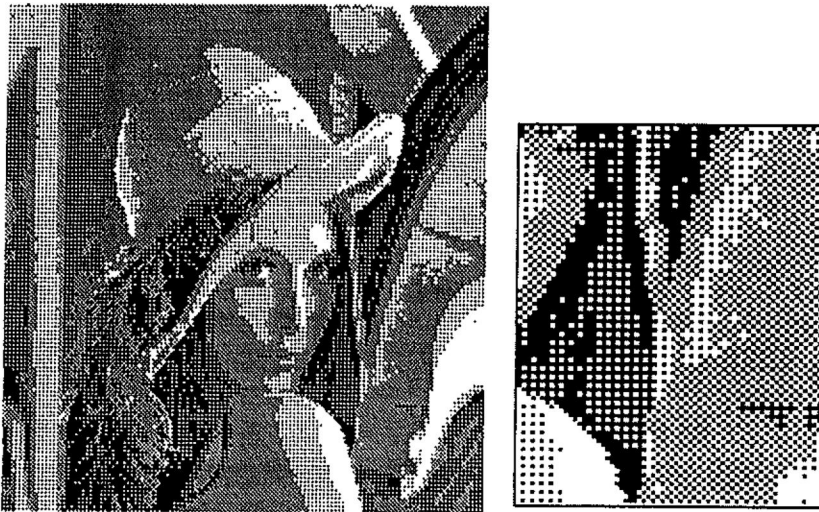


Figura 32. (a) Imagen generada con un Dither 2x2 y (b) recorte de un segmento de la misma donde se pueden observar los patrones generados



Figura 33. Imagen generada por binarización.

A simple vista podemos reconocer que la imagen resultante de una binarización (figura 33) es una representación mucho más pobre en cuanto a los detalles que se pueden percibir a simple vista y al número de intensidades que componen a la imagen original. Esto se debe a la gran pérdida de información que hay al representar en solo dos colores, blanco y negro, la gama de intensidades que hay en 255 niveles de gris.

Esta pérdida implícitamente conlleva otra más: la de las líneas que definen a la imagen. Al cruzar de un área clara a una oscura (y viceversa), hay un cambio importante y repentino en el valor de intensidad lo que es percibido como una línea que divide a ambas áreas.

Al binarizar una imagen se reduce el número de áreas generadas con intensidades similares a solo dos tipos de regiones (blanco y negro), todas esas áreas que se perdieron implicaban bordes que también desaparecieron. Esto provoca una pérdida en el delineado de las figuras que componen a la imagen.

Este mismo fenómeno se da con el dither pero en menor grado, y en general se observa en todas las representaciones que solo se basan en la intensidad subyacente de la imagen de entrada al hacer la cuantización.



Figura 34. (a) Imagen generada con un Dither 3x3 (b) figura generada con un Dither 4x4.

Las matrices de oscilación disminuyen el problema, por el mayor número de niveles de intensidad emulados, los cuales nos dan una mejor representatividad.

Sin embargo este efecto tiene un límite, ya que al utilizarse matrices mayores al orden 4x4 no se obtuvo una mejoría significativa, e incluso en un momento determinado la imagen resultante perdió calidad a simple vista al utilizar una impresora lasser normal.

Hay que recordar que este tipo de matrices se utilizan cuando se trata de impresiones de libros y periódicos (impresoras de gran formato). Otro efecto que se puede observar es la aparición de artefactos ajenos a la imagen de entrada.

La premisa es que la cuantización de los datos digitales, producen errores, si esos errores se repiten y se correlacionan con la señal, el error que resulta es repetitivo, cíclico y

matemáticamente determinable, en algunos casos. Las Matrices Dither de orden mayor pueden generar artefactos menos detectables.

Se puede observar en la figura 32 la aparición patrones cuadrados producidos por una matriz Dither $D_{2 \times 2}$.

En las figura 34 (a) y (b) podemos observar cómo, aun cuando siguen apareciendo dichos artefactos, se van volviendo un poco menos detectables con matrices de oscilación de orden mayor.

Un post proceso para resolver este tipo de problemas es la aplicación del método de difusión de error a la imagen cuantizada.

Difusión de error

En la figura 36 se presenta un ejemplo de binarización con difusión de error, compárese con la figura 33 (binarización sin difusión de error) y a simple vista hay una mejora dramática entre ambos resultados, sobre todo en la preservación de línea y figura. La dispersión del error hace que, aún cuando hay solo dos niveles de intensidad (blanco y negro), la umbralización produzca en la región una mezcla de píxeles blancos y negros (no todos iguales), que con la integración espacial se perciben como tonalidades de gris.

Se desea mantener la representación lograda, pero mejorando visualmente el resultado, esto es: mejorar la representación con relación a la forma en que el ojo humano le percibe. En la figura 35 podemos observar cómo cada píxel que va a ser procesado, difunde el error entre la vecindad de acuerdo al kernel de Floyd y Steinberg en una

forma acumulativa, de tal manera que $U(n1, n2)$ es igual a la imagen de entrada mas la suma de todos las afectaciones que ha sufrido $P(n1, n2)$, en términos de la implementación.



Figura 35. Forma en la que se difunde el error



Figura 36. Imagen Binarizada de Lena con difusión de error y acercamiento.

La figura 37 (a) es una imagen Dither $D3 \times 3$ habiendo aplicado Difusión de Error.

En el detalle figura 34(b) podemos observar artefactos en forma de pequeñas esquinas triangulares que aparecen repetidamente, las cuales son típicas en el Dither de orden $D3 \times 3$.

Al realizar la difusión de error los artefactos desaparecen y aún cuando hay cierto tramado este aparece más aleatoriamente, lo que contribuye a que la imagen sea de mayor calidad.



Figura 37. (a) Imagen de Lena con Dither y Difusión de error.

El error se distribuye alrededor del píxel actual, diluyendo con ello el patrón inducido por la matriz de oscilación.

Podemos apreciar en el detalle figura 34(b) como aparecen otros artefactos en forma de gusano. Pero éstos son menos visibles que los producidos por el Dither debido a que parecen ser más aleatorios que los que se perciben en grandes áreas de la imagen dither.

La aleatoriedad en la distribución de los puntos, es una premisa importante ya que mientras más patrones repetitivos haya, la calidad de representación será menor, aún cuando se trate de una representación no fotorrealista.

Este punto se comprende mejor con un ejemplo: imagínese un dibujo de una escena nevando, si los puntos que representan a la nieve aparecen como un tramado repetitivo, la escena pierde inmediatamente credibilidad aún cuando se trata de una representación no fotorrealista.

VI.2 Método del Rechazo

Los métodos probabilísticos toman este criterio y lo explotan, en el caso del método de rechazo la imagen de entrada funciona como una función de densidad de probabilidad FDP. De forma aleatoria se genera un valor que es comparado contra el valor de la FDP (umbral) si el valor rebaza al umbral se coloca el punto.

La figura 38 es una imagen generada por el método de Rechazo utilizando 300000 puntos aleatorios, en el detalle se observa la ausencia de artefactos y la aleatoriedad de los puntos que conforman la imagen.

De acuerdo a los requisitos planteados para una representación no fotorrealista de calidad:

- La correcta representación de la imagen en el dominio de las intensidades subyacentes
- La correcta representación espacial de la imagen mediante el uso de cierto artefacto representativo del método utilizado (e.g. puntos, mosaicos, cruces, etc), como se haría en la técnica artista que se está homologando.

- La similitud, lo más cercano posible, a una creación hecha a mano. En otros términos cierta imperfección que tienen todas las obras hechas a mano, y que una representación fotorrealista no tiene.

La imagen resultante es una representación bastante aceptable de las intensidades de la imagen y la aleatoriedad en el posicionamiento de los puntos realza el carácter no fotorrealista de la imagen obtenida.

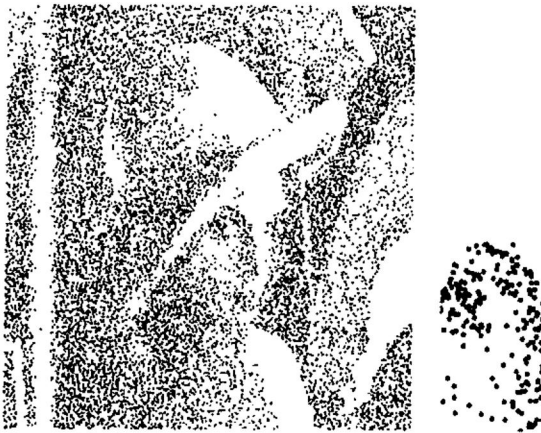


Figura 38. Imagen de Lenna generada por el método del rechazo con 30.000 puntos.

En comparación de las imágenes generadas hasta ahora por el Dither y la Difusión de Error, las imágenes generadas por el Método de Rechazo representan un enfoque totalmente diferente al planteado por los primeros dos métodos.

La aleatoriedad, además de eliminar los artefactos no inherentes a la imagen original, rompe también con los patrones repetitivos que se observan claramente en los detalles de las figuras 31, 32, 34, 36 y 37, mientras que en el detalle de la figura 38 se puede observar la ausencia de dichos artefactos.

Además se observa en el detalle de la figura 38 que los puntos que le forman mantienen la línea que define a la figura, además de parecer haber sido “puestos” en una forma más artesanal, menos mecánica que en el caso de las imágenes generadas por el dither. Los primitivos gráficos que se utilizan ya no son píxeles (como en dithering) sino círculos que emulan las manchas de tinta o puntos usados por un artesano.

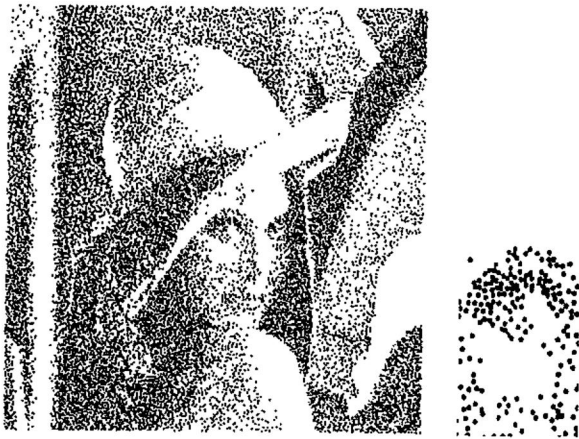


Figura 39. Imagen generada con el método del rechazo con distribución Poisson Disk.

La figura 39 se generó mediante el método de Rechazo pero utilizando una distribución Poisson Disk para determinar los puntos muestra.

A simple vista parece no haber una mejora significativa entre la figura 38 y la figura 39, sin embargo al utilizar posiciones pseudoaleatorias (distribución Poisson-disk), la imagen resultante representa mejor la línea y figura, y que los puntos ya no se vean caóticos con la distribución uniforme.

Hay que recordar que en el stippling hecho a mano (figura 40) un principio básico es el que cada punto sea cuidadosamente colocado de forma aleatoria y espaciados de acuerdo a la intensidad promedio del área representada. Sin perder la información espacial que proporcionan las líneas de los bordes y contornos



Figura 40. Stippling hecho a mano (Tomado de Randy Glass Studio).

El carácter no fotorrelista de la imagen obtenida por el método del rechazo es todavía somero, el proceso sigue sin tomar en cuenta la relación geométrica entre los puntos, en el sentido de que los artefactos generados solo responden a la intensidad subyacente (aunque en este caso de forma aleatoria).

Existen métodos que generan los puntos con otro criterio distinto de aleatoriedad en donde hay una relajación en la distribución de los puntos, además de tomar en cuenta la orientación espacial de la imagen y no solo la intensidad subyacente.

Los artistas de pluma y tinta han desarrollado técnicas satisfactorias para representar el valor tonal de la superficie por el espaciamiento disparejo de los puntos. Se observa en

la figura 40 que la colocación de los puntos inducen líneas que realzan la forma de la boca y las sombras que se forman a su alrededor.

El Stippling utiliza círculos de para representar las manchas de tinta, y procesos como la generación de conglomerados utilizando los diagramas de Voronoi para lograr una mejor representatividad de la figura e intensidad de los píxeles de la imagen de entrada. El uso de los Diagramas de Voronoi induce:

- la relajación de los puntos
- la correspondencia entre la densidad de puntos y el valor de intensidad
- iniciar una correcta relación entre la colocación del punto y la distribución espacial de la imagen.

VI.3 Método del Lloyd

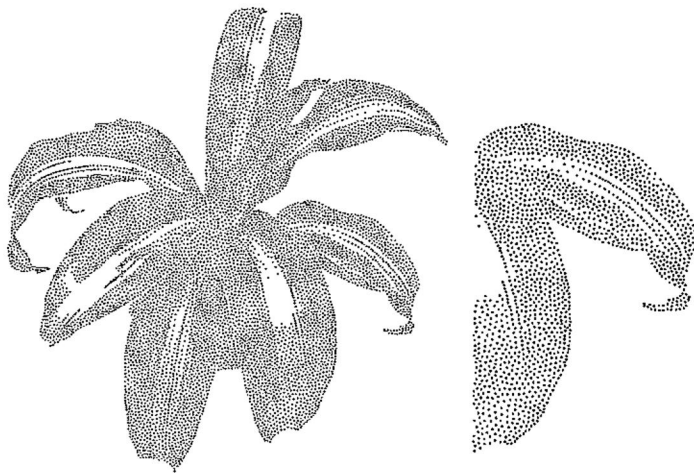


Figura 41. Imagen de salida generada con Stipples mediante el método de Lloyd con 30,000 stipples.

Al comparar visualmente las figuras 38 y 41, se nota en la figura 38 una distribución mucho más caótica de los stipples, lo que provoca que detalles pequeños se pierdan, principalmente las líneas de contornos. Esto se debe al carácter anárquico en la colocación de los puntos. De hecho la representación de las diferentes áreas de intensidad se pierde al observar la imagen en detalle, aún cuando la imagen completa si permite observarlas.

En cambio en la figura 41 al hacer el zoom se alcanzan a percibir detalles finos, a pesar de la limitada cantidad de stipples presentados.

Otro punto importante es que aún cuando se percibe la aleatoriedad en la colocación de los stipples, no se pierde la orientación espacial que guarda las líneas en la imagen de entrada.

Esto se parece mucho más a lo que encontramos en la figura 40 donde el artista marca trayectorias curvas que son seguidas por los stipples para representar de forma no fotorrealista los rasgos de la mejillas y otras áreas cercanas a la boca.



Figura 42. Imagen original y mismo detalle.

Al comparar visualmente la figura 41 y figura 42 se observa como los stipples en la imagen de salida aún cuando tienen una distribución aleatoria, siguen perfectamente las líneas de la flor de maíz a la que representan. La calidad de la recreación obtenida cubre los tres criterios de calidad del stippling antes citados.



Figura 43. Imagen Stippling formada por el método de Lloyd.

En la figura 43, al igual que en el caso de la figura 40 hay patrones generados en la imagen de salida de acuerdo a la información de la imagen de entrada dan realce a detalles de luminosidad y textura del objeto representado. Esto se logra únicamente por el uso de los criterios optimizados que el método utiliza en la ubicación de cada stipple, en las diferentes áreas que compone a la imagen de entrada (sin ninguna intervención humana).

Para realzar el carácter no fotorrealista de la figura 43 y figura 38, basta decir que el número de stipples utilizados para la representación fue de 30,000. Esto es 2 veces menos que el caso de la figura 38.

VI.3 Método de Agentes Inteligentes

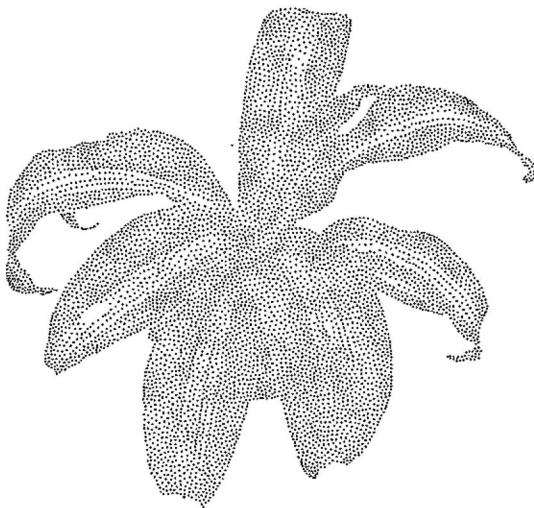


Figura 44. Imagen generada por el emulador del Sistema de Agentes Inteligentes.

El método de Agentes inteligentes tiene características similares al método de Lloyd.

En el algoritmo de Lloyd los sitios generadores agrupan a los puntos que están más cercanos a ese sitio generador que a cualquier otro, en el caso de agentes inteligentes una serie de partículas distribuidas aleatoriamente en el layer de trabajo se repelen unas a otras de acuerdo a fuerzas de repulsión relacionadas con la intensidad subyacente en cierta posición del layer de entrada. El efecto final es dividir a la imagen en regiones con intensidades similares.

En el método de Lloyd hay convergencia de los puntos generadores hacia los centros de masa de las diferentes regiones Voronoi. En el caso de Agentes Inteligentes, la propia dirección de los movimientos generados por las fuerzas de rechazo van provocando que los stipples se aglomeren y preserven la figura de la imagen de entrada.

Las partículas tienden a converger a posiciones donde prácticamente están estables (sin cambio o con cambios mínimos en su posicionamiento). Ver figura 44.

Una diferencia significativa entre ambos métodos es que en el caso de Lloyd el stipple se coloca donde se encuentra el punto generador una vez que el mismo, lo que asegura el cumplimiento de la condición de punto medio y la condición centroidal (ver características de un cuantizador óptimo página 55), entonces el stipple es colocado en una posición que nos da la mejor representación de dicha región, y por lo tanto la distribución de los mismos es la óptima, reduciendo al máximo la aparición de objetos que no correspondan a la imagen de entrada. En el caso de agentes inteligentes aún cuando tenemos una distribución similar, el posicionamiento no asegura las dos

condiciones de un cuantizador óptimo. Entonces al final los stipples en estas zonas se agrupan de acuerdo a la intensidad promedio, pero no forzosamente su posicionamiento final es el óptimo, como el caso de Lloyd.

Conclusiones

Aún cuando el propósito del halftoning es la representación de imágenes fotorrealistas de calidad usando medios de impresión con capacidad limitada (eg. imprentas de periódicos y revistas), su análisis aporta dos temas de especial interés para la generación de imágenes no-fotorrealistas: la cuantización y la aparición de patrones no inherentes a la imagen de entrada debido a esa misma cuantización.

Al observar los resultados en el proceso llamado binarización, se pudo ver como a la pérdida de información al cuantizar a solo 2 niveles de gris, corresponde una pérdida de información de línea y figura (pérdida de contornos).

El recrear imágenes no fotorrealistas implica una cuantización y una pérdida de información con respecto a la imagen de entrada, ya que una de las características del no fotorrealismo es utilizar un número limitado de primitivos gráficos para representar la imagen de entrada.

Dithering es un proceso halftoning que aporta mejores resultados visuales que la binarización. Pero a pesar de los esfuerzos por dispersar los umbrales que forman la matriz de oscilación, en mayor o menor proporción, inducen la aparición de artefactos en la imagen de salida. Estos artefactos son totalmente indeseables en NPR. En el stippling hecho mano, el artista tiene mucho cuidado en la colocación aleatoria de los stipples y constantemente agrega y quita stipples hasta lograr que a cierta distancia los patrones no aparezcan o respondan a la correcta representación de la forma, textura o información a resaltar, y no a patrones repetitivos indeseables.

El método de Difusión de error diseñado por Floyd y Steinberg, mejora notablemente el proceso de cuantización realizado (por dithering, binarización u otro método), mediante el uso de un kernel que distribuye el error en la vecindad y al hacerlo, toma en cuenta cómo percibe un ser humano una imagen, al menos en el proceso de lectura. En las imágenes resultantes se observó cómo se redujeron significativamente la aparición de artefactos indeseables. Este método sigue utilizando un número superior de primitivos.

En el stippling hecho a mano el artista cuida, al colocar los stipples siguiendo los contornos y con una cierta distancia mínima entre stipples. Induce patrones controlados para representar texturas, por ejemplo, lo que refuerza la calidad no fotorrealista del trabajo.

Los métodos probabilísticos pueden reducir a control del usuario, la cantidad de primitivos gráficos de la imagen de salida. Al aplicar el método del rechazo se observó como los patrones repetitivos prácticamente desaparecen, pero la colocación aleatoria con distribución uniforme de los puntos provoca una pérdida significativa de líneas y forma contenidas en la imagen de entrada. El uso de muestreo con distribución Poisson-disk mejora notablemente la información asociada a la textura y evita traslapes o que dos primitivos gráficos estén demasiado cercanos. Los resultados son también atractivos porque implican cálculos de complejidad $O(n)$.

Se presentó un método usando cuantizadores óptimos (el método de Lloyd), que asegura que cada stipple se colocará en el lugar donde se obtiene la mejor representación posible; en otras palabras las regiones se representan con stipples colocados de forma que minimizan el error cuadrático medio.

Las imágenes resultantes, aún cuando cuentan con un número significativamente menor de elementos para representar la imagen de entrada, tienen una calidad mayor (a simple inspección visual) en cuanto a las intensidades de las distintas regiones de la imagen, y en cuanto a colocación adecuada del stipple para representar la línea y la forma. Con este método también es posible usar primitivos gráficos mayores a un píxel (círculos rellenos), lo que es más acorde con la técnica artística del stippling a mano.

Por último se ha presentado una variación interesante que utiliza la lógica de los simuladores de sistemas de partículas, consiguiendo mediante la interacción de agentes inteligentes un efecto muy similar al de las regiones de Voronoi.

El método tiene la bondad de no estar restringido solo para el sitipling, aún cuando en esta tesis nos hemos orientado solo al manejo de stipples. Cada bot genera un stipple. No existe una interacción directa entre ellos. En el medio Ambiente en el cual viven los bots dejan rastros que representan fuerzas de repulsión. Un bot colocado en cierta posición repele en alrededor a todos aquellos que pretenden colocarse cerca, a una distancia mínima de acuerdo a la intensidad subyacente en la posición actual. Logrando con esto el mismo efecto de que en las zonas oscuras haya una mayor densidad de stipples y menor en las zonas claras.

Quizá la única desventaja de estos dos últimos métodos (CVT y stiplingBot) es su complejidad pues son métodos que están lejos de ser lineales (lo cual resultaría extraordinario).

Como última conclusión, NPR es un área de investigación floreciente con muchas oportunidades de aportación. El uso de primitivos gráficos de tamaño variable, establecer métricas de calidad automatizadas para el stippling y el descubrir nuevos métodos de complejidad menor, son temas que son una continuación lógica del presente trabajo.

Referencias

- Alejo, Hausner. *Graph Order Dither*. . Department of Computer Science, University of New Hampshire, Durham, NH, USA.
- Ascencio-Lopez, José Ignacio, Hugo Hidalgo-Silva, and Angel González-Fraga. *Imágenes no-fotorrealistas con primitivos gráficos puntuales*. UABC, 2006.
- Bigus, Joseph P. *Constructing intelligent agents using Java*. John Wiley & Sons. 2001.
- Bovick, Al. *Handbook of Image & Video Processing*. Elsevier Academic Press. 2005.
- Ezparsa-Nuñez, Eric, Ignacio Ascencio-López, and Amado Velázquez-Montalvo. *Conglomeración Jerárquica utilizando Diagramas de Voronoi Centrados y Arbol Genérico Mínimo*. UABC, 2006.
- Foley, Van Dam, Feiner, Hughes, and Phillips. *Introducción a la Graficación por Computadora*. Addison Wesley. 1996.
- Fuller, John. *OpenGL Programming guide*. Addison-Wesley Publishing Company.
- Iterrante, Victoria. "Perceptual and Artistic Principles for Effective Computer Design." *SIGGRAPH course applications to art, 2007*: 1-26.
- Jérôme, Thoma. *Non-Photorealistics Rendering Thecniques for Real Time Character Animation*. Aachen, Germany, 2002.
- Kenneth, E. Hoff III, Tim Culver, John Keyser, Ming Lin, y Dinesh Manocha. *Fast Computation of Generalized Voronoi Diagrams Using Graphics Hardware*. ACM press. 1999.
- Livingstone, Margaret. *Vision and Art: The Biology of Seeing*. Harry N. Abrams, 2002.

- Lloyd, S. P. "Least squares quantization in PCM." *IEEE Transactions on Information Theory*, March 1982: 28, 129-137.
- Oliver, Deussen, Hiller Stefan, van Overveld Cornelius, and Strothotte Thomas. *Floating Points: A Method for Computing Stipple Drawings*. EUROGRAPHICS 2000 / M. Gross and F.R.A. Hopgood, 2000.
- Reynolds, Craig W. "Flock, Herds and Schools: A Distributed Behavioral Model." *Computer Graphics*, 21(4), July 1987: 25-34.
- Russell, Stuart J., and Peter Norving. *Artificial Intelligence*. Englewood Cliffs, New Jersey: Prentice-Hall, 1995.
- Schlechtweg, Stefan, Tobias Germer, and Thomas Strothotte. "RenderBots-Multi Agent Systems for Direct Image Generation." *The Eurographics Association and Blackwell Publishing 2006*, Volume 24 (2005), Number 2: 137-148.
- Secord, Adrian. *Weighted Voronoi stippling*. 2nd. International Symposium on Non-Photorealistic Animation and Rendering, NPAR, 2002.
- Secord., Adrian Joseph. *Random Mask on Paper. Non Photorealistic rendering with Small Primitives*. The University of British Columbia, 2002.
- Tessier, C., L. Chaudron, and H.J. Miller. *Conflicting Agents: Conflict Management in Multi-Agent Systems*. Boston: Kluwer Academic Publisher, 2002.
- Thomas, Strothotte, and Schlechtweg Stefan. *Non-Photorealistic Computer Graphics*. Morgan Kaufmann, 2002.
- Tran-Quan, Luang, Seth Ankush, Klein Allison, and Lawrence Jason. *Isoluminant Color Picking for Non-Photorealistic Rendering*. 2005.

V., Ostromoukhov, and Hersch R. *Multi-color and artistic dithering*. In Proc of SIGGRAPH, 1999.

Weiss, G. *Multiagent System: A Modern Approach to Distributed Artificial Intelligence*. Cambridge, MA, USA: MIT Press/AAAI press, 1999.