

# Desarrollo e implementación de una plataforma web para la gestión de imágenes digitales

---

Informe de propuesta

Asignatura	:	Computación Paralela.
Profesor	:	Oscar Magna Veloso.
Integrantes	:	Grupo curso.
Coordinador	:	Felipe Canales Saavedra.
Fecha	:	3 de junio del 2016.

## Tabla de contenidos

1.	Introducción .....	1
2.	Objetivos .....	2
2.1.	Objetivos generales.....	2
2.2.	Específicos .....	2
3.	Marco teórico.....	3
3.1.	Clúster .....	3
3.2.	Nodos .....	4
3.3.	Clúster Beowulf .....	5
3.4.	Open MPI.....	5
3.5.	Ruido .....	6
3.6.	Ubuntu .....	6
3.7.	Python .....	6
3.8.	Web.Py .....	7
3.9.	OpenCV.....	7
3.10.	Ley de Amdahl .....	7
3.11.	Speedup.....	7
3.12.	Eficiencia.....	8
3.13.	Costo.....	9
3.14.	Tratamiento de imágenes .....	10
3.14.1.	Canales RGB.....	10
3.14.2.	Tratado de imágenes.....	10
3.14.3.	Etapas del procesamiento de imágenes.....	11
4.	Alcances y Limitaciones.....	12
5.	Metodología .....	13
5.1.	Grupos de trabajo .....	14
5.2.	Modelo de programación secuencial.....	15
5.3.	Modelo de programación paralelo .....	16
5.4.	Estrategia de paralización .....	18
5.5.	Web socket.....	19
5.6.	Topología de red a emplear para la plataforma web.....	20
6.	Teorización matemática y propuesta de algoritmos.....	21
6.1.	Realzado y filtrado de imágenes .....	21

6.1.1.	Traspuesta de imagen .....	21
6.1.2.	Invertir imagen .....	22
6.1.3.	Redimensionar imagen.....	22
6.1.4.	Convolución con gestión de bordes .....	26
6.1.5.	Enfoque y Desenfoque .....	27
6.1.6.	Efecto espejo .....	28
6.1.7.	Efecto luminosidad.....	29
6.1.8.	Efecto Polar .....	30
6.1.9.	Efecto deformación Bézier .....	32
6.1.10.	Efecto posicionar los bordes .....	33
6.1.11.	Efecto deformación de malla .....	35
6.1.12.	Inversión de colores .....	36
6.1.13.	Escala de grises.....	37
6.1.14.	Efecto sepia .....	37
6.1.15.	Efecto RGB.....	38
6.1.16.	Efecto LoG .....	40
6.1.17.	Efecto 3D de imágenes 2D .....	42
6.1.18.	Efecto “zooming” .....	43
6.2.	Mejora de imagen .....	44
6.2.1.	Mejora de brillo y contraste .....	44
6.2.2.	Mejora de HDR .....	45
6.3.	Movimiento de imagen .....	46
6.3.1.	Suma de fotos.....	46
6.3.2.	Time lapse .....	47
6.3.3.	Bullet time .....	48
6.3.4.	Stop motion.....	49
6.4.	Operaciones matemáticas.....	50
6.4.1.	Dispersiones gaussianas .....	50
6.4.2.	Transformada rápida d Fourier .....	51
6.4.3.	Convolución – Desconvolución .....	55
6.4.4.	Laplaciano de la Gaussiana.....	59
7.	Cronograma.....	63
8.	Cuantificación de esfuerzo.....	64

9.	Planificación .....	69
9.1.	Control de avances .....	69
9.2.	Control de valor ganado .....	70
10.	Marco de evaluación .....	71
10.1.	Criterios de permanencia y amonestaciones .....	71
11.	Desarrollo del proyecto .....	72
11.2.	Herramientas utilizadas .....	72
11.3.	Análisis de desempeño .....	<b>¡Error! Marcador no definido.</b>
11.3.1.	Tablas comparativas .....	<b>¡Error! Marcador no definido.</b>
11.3.2.	Gráficos comparativos .....	<b>¡Error! Marcador no definido.</b>
ANEXOS	.....	87
ANEXO A.	Formato de carta de amonestación .....	88
ANEXO B.	Formato de solicitud de aplazo .....	90
ANEXO C.	Pauta de evaluaciones parciales .....	91
ANEXO D.	Pauta de evaluación final .....	92
ANEXO E.	CPM del proyecto .....	93
ANEXO F.	CPM, ruta crítica .....	95
ANEXO E.	WBS de la aplicación .....	96
ANEXO F.	Plan de hitos .....	97
ANEXO G.	Total de esfuerzos y costos .....	98
ANEXO G.	Evaluaciones .....	101
	Evaluaciones de etapa final .....	101
	Evaluaciones finales .....	103
ANEXO I.	Cronograma final .....	1
ANEXO J.	Costo del personal .....	1
ANEXO K.	Totales de esfuerzo y Costo .....	1

## ÍNDICE DE TABLAS

<i>Tabla 1, Conformación de integrantes del grupo gerencia .....</i>	14
<i>Tabla 2, Conformación de integrantes del grupo uno .....</i>	14
<i>Tabla 3, Conformación de integrantes del grupo dos .....</i>	14
<i>Tabla 4, Conformación de integrantes del grupo tres .....</i>	14
<i>Tabla 5, Cronograma de actividades .....</i>	63
<i>Tabla 6, Tabla de Cuantificación de esfuerzos .....</i>	64
<i>Tabla 7, Tablero Kanban para la distribución de tareas .....</i>	69
<i>Tabla 8, Pauta de evaluaciones parciales .....</i>	91
<i>Tabla 9, Pauta de evaluación final .....</i>	92
<i>Tabla 10, Tabla de esfuerzos y costos .....</i>	98
<i>Tabla 11, Resumen de evaluación por etapas y la evaluación final .....</i>	103
<i>Tabla 12, Cronograma final .....</i>	1
<i>Tabla 13, Costo del personal, asociado al proyecto .....</i>	1
<i>Tabla 14, Totales acumulados de esfuerzo y costo .....</i>	1

## ÍNDICE DE ILUSTRACIONES

<i>Ilustración 1, Modelo de MPI</i>	5
<i>Ilustración 2, Modelo general de programación secuencial</i>	15
<i>Ilustración 3, Modelo lógico de aplicación</i>	16
<i>Ilustración 4, Esquema teórico de servidor DHCP</i>	17
<i>Ilustración 5, Modelo general de programación paralela</i>	18
<i>Ilustración 6, Diagrama lógico de topología estrella</i>	20
<i>Ilustración 7, Destino de imagen modificada</i>	21
<i>Ilustración 8, Matriz de rotación de imagen</i>	21
<i>Ilustración 9, Ejemplo de rotación de imagen</i>	21
<i>Ilustración 10, Ejemplo del efecto invertir imagen</i>	22
<i>Ilustración 11, Imagen de ejemplo</i>	23
<i>Ilustración 12, Ejemplo de clasificación por kNN</i>	23
<i>Ilustración 13, Ejemplo de interpolación</i>	24
<i>Ilustración 14, Resultado</i>	25
<i>Ilustración 15, Función bicubic</i>	26
<i>Ilustración 16, Resultado de algoritmo bicubic</i>	26
<i>Ilustración 17, Efecto espejo en eje vertical</i>	28
<i>Ilustración 18, Simetría por reflexión</i>	28
<i>Ilustración 19, Efecto luminosidad</i>	29
<i>Ilustración 20, Efecto polar</i>	30
<i>Ilustración 21, Coordenadas rectangulares vs. polares</i>	30
<i>Ilustración 22, Ejemplo de conversión</i>	31
<i>Ilustración 23, Efecto de deformación Bézier</i>	32
<i>Ilustración 24, Efecto de deformación de malla</i>	35
<i>Ilustración 25, A la izquierda la imagen original, y a la derecha el filtro aplicado.</i>	36
<i>Ilustración 26, Colores primarios</i>	38
<i>Ilustración 27, Representación de 3D de un pixel</i>	39
<i>Ilustración 28, Efecto LoG</i>	40
<i>Ilustración 29, Ejemplo de renderizado volumétrico</i>	42
<i>Ilustración 30, Ejemplo de efecto zooming</i>	43
<i>Ilustración 31, Extrapolación de fronteras de imagen</i>	43
<i>Ilustración 32, Imagen con distinto grado de exposición fotográfica</i>	45
<i>Ilustración 33, Resultado de filtro HDR</i>	46
<i>Ilustración 34, Adición de colores</i>	47
<i>Ilustración 35, Adición de imágenes</i>	47
<i>Ilustración 36, Ejemplo de matriz de convolución</i>	56
<i>Ilustración 37, Aplicación del PSF</i>	57
<i>Ilustración 38, Desconvolución</i>	58
<i>Ilustración 39, Resultado final</i>	59
<i>Ilustración 40, Imagen de ejemplo para la detección de bordes</i>	59
<i>Ilustración 41, Representación matemática de un borde</i>	60

<i>Ilustración 42, Primera derivada de la curva representada en la ilustración 26 .....</i>	60
<i>Ilustración 43, Segunda derivada de la curva en la ilustración 9 .....</i>	61
<i>Ilustración 44, Efecto del operador Laplaciano .....</i>	61
<i>Ilustración 45, Gráfico de presupuesto al término del proyecto .....</i>	100
<i>Ilustración 46, Tabla evaluaciones finales de grupo 1 .....</i>	101
<i>Ilustración 47, Tabla de evaluaciones finales de grupo 2 .....</i>	101
<i>Ilustración 48, Tabla de evaluaciones finales grupo líder .....</i>	102
<i>Ilustración 49, Tabla de evaluaciones finales de grupo 3 .....</i>	102
<i>Ilustración 50, Gráfico de valor ganado y costo asociado al proyecto.....</i>	3

## 1. Introducción

En el día de hoy, la mayoría del software es sustentado de manera secuencial, esto implica desarrollar una solución que se ejecute paso a paso, lo que implica en una implementación de flujo de instrucciones sucesivas, con el fin de dar solución a una problemática determinada.

En la actualidad, los altos volúmenes de información que se manejan obligan a buscar nuevos métodos y/o paradigmas que faciliten el trabajar con esta información, y que lo hagan en el menor tiempo posible. Es por esta necesidad que surgen herramientas que logran hacer trabajar al unísono distintos procesadores para poder cumplir con un objetivo en común. Para que esta metodología funcione, es necesario realizar estudios previos a la tarea a desarrollar para dar un buen uso a esta herramienta.

En esta fase del laboratorio, la atención se centrará nuevamente en el desarrollo de una solución paralela que sea capaz de procesar imágenes digitales, centrándose en cuatro grandes áreas: Realizado y filtrado, mejora, movimiento y operaciones matemáticas. Cada una de estas con sus propios tipos de efectos u operaciones. Motivo por el cual, dada la alta carga de trabajo que requiere procesar, es posible aprovechar las capacidades del procesamiento paralelo para dividir las funciones y balancear de mejor manera la carga de trabajo, resultando en una operación eficiente de la información a tratar.

Para cumplir con los objetivos de esta fase, se trabajará en un ambiente de procesamiento paralelo, implementando esto en un clúster bajo el sistema operativo Ubuntu, en el lenguaje de programación Python y la librería de paso de mensajes MPI.



## 2. Objetivos

### 2.1. Objetivos generales

Desarrollo e implementación de un entorno web basado en una plataforma clúster homogénea, que permita entregar soluciones a los siguientes problemas:

- Técnicas de realzado y filtrado de imágenes digitales.
- Técnicas de mejora.
- Aplicación de movimiento de imágenes.
- Aplicación de operaciones matemáticas a imágenes.

### 2.2. Específicos

- Implementación de una red interna (intranet) que permita acceder a la plataforma de gestión de imágenes, manteniendo la principal idea de que sea “amigable” e intuitiva para el usuario final.
- Realizar la implementación de una plataforma clúster de tipo homogéneo, dependiendo de la cantidad de computadores disponibles.

## 3. Marco teórico

### 3.1. Clúster

Un clúster es un grupo de sistemas independientes, conocidos como nodos, que trabajan juntos como un sistema único para garantizar la premisa de garantizar al usuario la disponibilidad de recursos. La organización en clústeres permite a los usuarios y administradores tener acceso a los nodos y administrarlos como un sistema único en lugar de como equipos independientes. Un clúster de servidores puede comprender hasta ocho nodos y se puede configurar de tres formas: como clúster de servidores de nodo único, como clúster de servidores de dispositivo de quórum único o como clúster de servidores de conjunto de nodos mayoritario.

Es decir, un clúster funciona como un sistema único o monolítico, es decir, para un usuario cualquiera; al momento de enfrentarse a este; lo verá como un todo funcionando y pensará que se enfrenta a una sola computadora.

El clúster puede ser dividido en tres distintas topologías, las cuales son detalladas a continuación:

- **High Performance:** Son clúster en los cuales se ejecutan tareas que requieren de gran capacidad computacional, grandes cantidades de memoria, o ambos a la vez. El llevar a cabo estas tareas puede comprometer los recursos del clúster por largos periodos de tiempo.
- **High Availability:** Su objetivo es proveer la disponibilidad y la confiabilidad. Estos clústeres tratan de brindar la máxima disponibilidad de los servicios que ofrecen. La confiabilidad se provee mediante software que detecta fallos y permite recuperarse frente a los mismos, mientras que en hardware se evita tener un único punto de fallos.
- **High Throughput:** Son clústeres cuyo objetivo de diseño es el ejecutar la mayor cantidad de tareas en el menor tiempo posible. Existe independencia de datos entre las tareas individuales. El retardo entre los nodos del clúster no es considerado un problema mayor.

Existen distintas técnicas en las que estos clústeres se apoyan para poder controlar la carga de información o datos a los que son sometidos. La técnica más usada es el:

- **Balanceo de carga:** El balance o balanceo de carga es un concepto usado en informática que se refiere a la técnica usada para compartir el trabajo a realizar entre varios procesos, ordenadores, discos u otros recursos. Está íntimamente ligado a los sistemas de multiprocesamiento, o que hacen uso de más de una unidad de procesamiento para realizar labores útiles.

### 3.2. Nodos

Pueden ser simples ordenadores, sistemas multiprocesador o estaciones de trabajo (Workstation). En informática, de forma muy general, un nodo es un punto de intersección o unión de varios elementos que confluyen en el mismo lugar. Ahora bien, dentro de la informática la palabra nodo puede referirse a conceptos diferentes según el ámbito en el que nos movamos:

- En redes de computadoras cada una de las máquinas es un nodo, y si la red es Internet, cada servidor constituye también un nodo.
- En estructuras de datos dinámicas un nodo es un registro que contiene un dato de interés y al menos un puntero para referenciar (apuntar) a otro nodo. Si la estructura tiene solo un puntero, la única estructura que se puede construir con él es una lista, si el nodo tiene más de un puntero ya se pueden construir estructuras más complejas como árboles o grafos.

El clúster puede estar conformado por nodos dedicados o por nodos no dedicados.

En un clúster con nodos dedicados, los nodos no disponen de teclado, ratón ni monitor y su uso está exclusivamente dedicado a realizar tareas relacionadas con el clúster. Mientras que, en un clúster con nodos no dedicados, los nodos disponen de teclado, ratón y monitor y su uso no está exclusivamente dedicado a realizar tareas relacionadas con el clúster, el clúster hace uso de los ciclos de reloj que el usuario del computador no está utilizando para realizar sus tareas.

Cabe aclarar que a la hora de diseñar un clúster, los nodos deben tener características similares, es decir, deben guardar cierta similaridad de arquitectura y sistemas operativos, ya que si se conforma un clúster con nodos totalmente heterogéneos (existe una diferencia grande entre capacidad de procesadores, memoria, disco duro) será ineficiente debido a que el middleware delegará o asignará todos los procesos al nodo de mayor capacidad de cómputo y solo distribuirá cuando este se encuentre saturado de procesos; por eso es recomendable construir un grupo de ordenadores lo más similares posible.

### 3.3. Clúster Beowulf

Para el desarrollo de la plataforma de clúster, que se utilizará durante el laboratorio, se implementará un clúster de tipo Beowulf.

Un clúster Beowulf es una colección de ordenadores conectados en una red privada que pueden ser usados para computación paralela. Consisten en una comodidad de hardware utilizando software de uso libre, como Linux o BSD, usualmente junto con PVM (Parallel Virtual Machine) o MPI (Message Passing Interface). El estándar de conexión consiste en un nodo maestro que controla un cierto número de nodos esclavos. Los nodos esclavos tienen un acceso general a todos los archivos del mismo servidores.

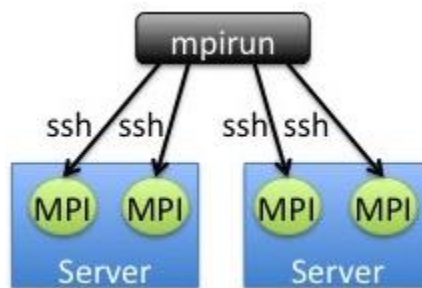
### 3.4. Open MPI

MPI significa “Message Passing Interface” y procede del esfuerzo de la comunidad de la computación paralela (industria y grupos de desarrollo de software libre) que creó el MPI Forum con el objetivo de desarrollar un estándar para programación paralela basada en paso de mensajes. El primer estándar se publicó en 1994 y, posteriormente, se generó una segunda versión “MPI-2” publicada en 1997. A partir de estas definiciones existen implementaciones tanto comerciales como de libre distribución (nosotros utilizaremos MPICH). MPI es, junto con PVM “Parallel Virtual Machine”, uno de los paquetes de software más populares en el entorno de aplicaciones basadas en paso de mensajes.

Una aplicación paralela típica sería aquella en la que un proceso maestro crea varios procesos esclavos para realizar, cada uno de ellos, una parte del trabajo global.

Gráficamente, este modelo es el siguiente:

*Ilustración 1, Modelo de MPI*



Fuente: <http://blogs.cisco.com/performance/tree-based-launch-in-open-mpi>

Si se dispone de más de una máquina física (por ejemplo, PC's o Supercomputadores) que tienen interconectividad entre sí a través de una red de datos, MPI permitirá contemplar dichas máquinas físicas como un conjunto de nodos de computación (clúster) en el que ejecutar alguna aplicación paralela de tal forma que los procesos se distribuyan entre las distintas CPU's disponibles y así poder ejecutarse con mayor paralelismo real.

Open MPI es quién lidera actualmente las implementaciones MPI-2, usadas por muchas supercomputadoras del TOP 500. Es de código abierto y desarrollado y mantenido por el consorcio académico, de investigación y la industria y asociados. El proyecto de Open MPI ha establecido como meta ser la mejor librería de interface para el traspaso de información.

### 3.5. Ruido

En comunicación, se denomina ruido a toda señal no deseada que se mezcla con la señal útil que se quiere transmitir. Es el resultado de diversos tipos de perturbaciones que tiende a enmascarar la información cuando se presenta en la banda de frecuencias del espectro de la señal, es decir, dentro de su ancho de banda.

### 3.6. Ubuntu

Sistema operativo basado en GNU/Linux, este es distribuido como software libre y está disponible en distintas versiones. Para este laboratorio se usará la versión 14.04 con su entorno de escritorio Unity, ya que todos los integrantes se encuentran familiarizados.

### 3.7. Python

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.

Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma.

### 3.8. Web.Py

Se trata del software escrito por Aaron Swartz para la creación de webs dinámicas con Python. Oficialmente fue publicado mientras trabajaba en reddit.com (la cual posteriormente fue rescrita utilizando otro tipo de herramientas). “Simple como potente”, “Es el framework anti-framework. web.py no se interpone en tu camino”; algunas de las características y opiniones por las cuales se da a conocer en el sitio oficial webpy.org. Es de dominio público, por lo que puede aplicarse para el desarrollo de cualquier tipo de proyecto.

### 3.9. OpenCV

OpenCV es una biblioteca libre de visión artificial originalmente desarrollada por Intel. Desde que apareció su primera versión alfa en el mes de enero de 1999, se ha utilizado en infinidad de aplicaciones. Desde sistemas de seguridad con detección de movimiento, hasta aplicaciones de control de procesos donde se requiere reconocimiento de objetos. Esto se debe a que su publicación se da bajo licencia BSD, que permite que sea usada libremente para propósitos comerciales y de investigación con las condiciones en ella expresadas.

OpenCV es multiplataforma, existiendo versiones para GNU/Linux, Mac OS X y Windows. Contiene más de 500 funciones que abarcan una gran gama de áreas en el proceso de visión, como reconocimiento de objetos (reconocimiento facial), calibración de cámaras, visión estéreo y visión robótica.

### 3.10. Ley de Amdahl

El aumento de rendimiento que puede obtenerse al mejorar alguna parte de una computadora puede calcularse utilizando la Ley de Amdahl.

La Ley de Amdahl establece que la mejora obtenida en el rendimiento al utilizar algún modo de ejecución más rápido está limitada por la fracción de tiempo que se pueda utilizar ese modo más rápido. Define la ganancia de rendimiento o aceleración (speedup) que puede lograrse al utilizar una característica particular.

### 3.11. Speedup

En el campo de la arquitectura de computadores, aumento de velocidad es una medida para la mejora relativa de rendimiento al ejecutar una tarea. La noción de aceleración fue establecida por la Ley de Amdahl, que se centra en particular en el contexto de procesamiento en paralelo. Sin embargo, la aceleración puede ser utilizada más generalmente para mostrar el efecto de cualquier mejora del rendimiento.

Para valores de latencia, aceleración se define por la siguiente fórmula:

$$S = \frac{T_{antiguo}}{T_{nuevo}}$$

Dónde:

$S$  = SpeedUp resultante

$T_{antiguo}$  = Tiempo antiguo de ejecución, o sin mejora

$T_{nuevo}$  = Tiempo nuevo de ejecución, o con mejora

Para valores de rendimiento, el rendimiento mejorado estará en el numerador y el rendimiento original estará en el denominador. Esto se debe a que es una cantidad relativa, estamos comparando dos casos concretos de ejecución.

$$S = \frac{P_{nuevo}}{P_{antiguo}}$$

$S$  = SpeedUp resultante

$P_{antiguo}$  = performance antiguo, sin mejora

$P_{nuevo}$  = performance nuevo, con mejora

Cuando se aplica en el caso paralelo, aumento de velocidad se puede predecir a partir de la ley de Amdahl.

### 3.12. Eficiencia

Llamando a  $S_p$  al SpeedUp para  $p$  procesadores, Aceleración lineal o SpeedUp ideal se obtiene cuando  $S_p = p$ . Cuando se ejecuta un algoritmo con la aceleración lineal, duplicando el número de procesadores se duplica la velocidad. Como esto es ideal, se considera muy buena escalabilidad.

La eficiencia es una medida de rendimiento que se define como:

$$E_p = \frac{S_p}{p}$$

Donde:

$E_p$  = Eficiencia en  $p$  procesadores

$S_p$  = SpeedUp para  $p$  procesadores

$p$  = procesadores

### 3.13. Costo

También llamado producto procesador producto procesador-tiempo o trabajo.

Un sistema es óptimo en coste si el coste de resolver un problema en el sistema paralelo tiene el mismo crecimiento asintótico (en función del tamaño del problema) que el coste en el mejor sistema serie equivalente.



$$\text{Costo} = p * T_p$$

Donde:

*Costo = Costo del proceso*

*T<sub>p</sub> = Tiempo paralelo*

*p = procesadores*

### 3.14. Tratamiento de imágenes

#### 3.14.1. Canales RGB

Son los canales que contienen los colores básicos (Rojo, Verde y Azul de sus siglas en inglés), se entiende que cada pixel de cualquier imagen contendrá un canal RGB. Es por lo que la combinación de estos, es que es posible ver los colores que componen una imagen digital.

#### 3.14.2. Tratado de imágenes

Se trata de diversos procedimientos y técnicas que se aplican a imágenes digitales (dentro de este contexto). Su objetivo principalmente es: filtrar, realce de imagen (el más común es la eliminación de ruido), cambios de color (por medio de los canales RGB), modificación del contraste (por cambios que se realicen en el histograma), suavizados de color, entre otros.

De lo anterior, podemos definir cuáles son las técnicas más pedidas o las más comunes que se realizan. A continuación, se definen las siguientes:

- **Filtrado de imagen:** Consiste en la eliminación de, por lo menos, un color del canal RGB (en toda la imagen. Esto quiere decir, que si se eliminan los canales R (Rojo) y G (Verde), se obtiene un filtrado de color B (Azul). O también realizar mezclas de colores para obtener colores compuestos, por ejemplo, realizar filtrados por color magenta.
- **Mejora de imagen:** Básicamente consiste en la eliminación de ruido provocado por la luz, que se da cuando las tonalidades obtenidas difieren con las vistas por el ojo humano (cambios de matices de los colores o, derechamente, cambio total en la tonalidad del color). La corrección se realiza mediante la modificación de los canales RGB que se vean afectados, teniendo la premisa de aumentar los canales en los colores “fuertes” y no tanto en los colores “débiles”.
- **Realce de imagen:** Consiste en el aumento de los canales RGB de los colores que no pueden ser vistos por el ojo humano, con el fin de obtener un resultado (incluyendo a imágenes en alta definición) más vivo y natural. Es lo que se muestra en algunas cámaras

digitales, en lo que se conoce como procesamiento HDR (High Dynamic Range).

- Es posible encontrar otras técnicas de procesamiento de imágenes, como lo son: el aumento de brillo, manipulación de contraste, cambio de colores a negativo (Invirtiendo los canales RGB), tratamiento de ojos rojos, entre otras.

### 3.14.3. Etapas del procesamiento de imágenes

De manera muy general, las etapas que se ven involucradas en este tipo de procesamiento son persistentes en los softwares que son posibles de encontrar en el mercado, las diferenciaciones se presentan en el resultado que se desea esperar. A grosso modo, es posible resumir estas etapas en las siguientes:

- **Captura:** Proceso por el cual se obtiene una imagen visual. Aquí entran en juego variables como: tipo de cámara, distancia del objeto, condiciones del día, cantidad de píxeles a usar, etc. Es decir, se trata de decidir cómo convertir un objeto en una representación para poder ser procesada (de la mejor manera) digitalmente.
- **Pre-procesamiento:** Se incluyen, en esta fase, técnicas como la reducción de ruido y realce de detalles (Por ejemplo, si se toma una imagen de un cesto de frutas, se decide reducir el fondo de esta, ya que no representa ningún interés). En este momento, las funciones que son aplicadas a una imagen digital desencadenan en otra, totalmente, nueva.
- **Segmentación:** Es el proceso en el cual se divide la imagen en objetos que sean del interés del usuario, por ejemplo: Del cesto de frutas, aplicar algún tipo de operador que permita extraer solo una fruta de la totalidad del cesto.
- **Extracción de características:** En este proceso, se realiza la extracción de características convenientes que resulten de interés para poder diferenciar un objeto de otro (por ejemplo: tamaño y forma). Este procedimiento es realizado de manera obligatoria, por algún procesador de imágenes, ya que las variables que entran en juego son las que utilizará la computadora para realizar todos los posibles ajustes que se necesiten.
- **Reconocimiento:** Proceso mediante el cual se asocia un significado a un conjunto de objetos reconocidos, es decir, entran en juego

patrones de reconocimiento automatizado. En este momento, los pasos anteriores entran a un algoritmo de toma de decisiones o alguna base de datos para poder realizar esta identificación. Siguiendo con el ejemplo de la cesta de frutas, se podría aplicar un algoritmo de reconocimiento para poder determinar a qué tipo de fruta corresponde cada una en ese cesto.

## 4. Alcances y Limitaciones

Se implementará una plataforma web que será capaz de recibir los archivos de imágenes por parte del usuario, estará alojada bajo un servidor DHCP y se configurará mediante la red interna propia de los laboratorios, con lo cual se podrá acceder mediante cualquier computador conectado a esta red. Esto permitirá al usuario obtener los resultados en tiempo real, siendo desplegados dentro de la misma plataforma (facilitando el poder descargar la imagen). La plataforma web, en sí misma, estará construida con el micro framework Web.py, propio de Python y la interfaz del clúster estará realizada con la librería MPI propia de este lenguaje.

El proyecto se encuentra limitado en la forma en que se entregan las soluciones a los problemas, es decir, para los efectos de edición de imágenes se podrán acceder de a uno por vez, mostrando la imagen editada.

También se cuenta con limitaciones logísticas, en específico, el problema de no poder acceder de manera inmediata a los laboratorios, para poder realizar una configuración correcta del clúster y todo lo que compete. Ya que actualmente, todo está siendo en los computadores personales de cada alumno.

En este informe se plantea el uso de un servidor DHCP para poder compartir la información y poder hacer el “levante” de la plataforma web para quede a disposición de la red interna de los laboratorios.

Si bien, esta idea aún sigue vigente. Por el principal problema, de no poder acceder a los laboratorios, se configuró la plataforma para poder funcionar mediante un socket de red, lo cual ahorra bastante tiempo y permite, de igual manera que lo anterior, estar disponible en toda la red.

## 5. Metodología

Para lograr un desarrollo óptimo de este laboratorio y considerando las fechas de entregas (Entrega de propuesta, avance 1, avance 2, y entrega final), además tomando en cuenta el reducido número de grupos de trabajo. La metodología más lógica a usar es la “Metodología XP” (Extreme programming). Para complementar de la mejor manera esta metodología, se decidió agregar un tablero Kanban (para tener un control de los avances) y reuniones.

Los roles necesarios para que esta metodología tenga éxito, son los siguientes:

- Programador: Escribe las pruebas unitarias y produce el código del sistema. Es el elemento más importante del equipo.
- Cliente: Escribe las historias de usuario y las pruebas funcionales para validar su implementación. Asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar el mayor valor de negocio.
- Tester: Ayuda al cliente a escribir las pruebas funcionales. Ejecuta pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- Tracker: Es el encargado de seguimiento. Proporciona realimentación al equipo. Debe verificar el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, comunicando los resultados para mejorar futuras estimaciones.
- Entrenador: Responsable del proceso global. Guía a los miembros del equipo para seguir el proceso correctamente.
- Consultor: Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto. Ayuda al equipo a resolver un problema específico. Además, este tiene que investigar según los requerimientos.
- Gestor: Es el dueño de la tienda y el vínculo entre clientes y programadores. Su labor esencial es la coordinación.

El equipo de trabajo, será dividido en 4 áreas: Jefe de proyecto, jefe del área de desarrollo, jefe del área de diseño y subjefe de proyecto. Como caso especial, el desarrollo de los análisis de Q.A. será realizado por todos los integrantes de cada grupo. Sin embargo, está configuración permite movilidad laboral, pero sin perder los objetivos de cada grupo.

Las reuniones que se añaden, son de carácter obligatorio (permitiendo inasistencias justificadas) con una duración de a lo menos 20 minutos, donde se discutirán los avances realizados, problemas que hayan surgido y las evaluaciones parciales. Si se da la situación de que los avances van lentos o hay problemas entre los integrantes de algún grupo de trabajo, se realizará una reunión pequeña (en duración) para corregir esos problemas.

## 5.1. Grupos de trabajo

Tabla 1, Conformación de integrantes del grupo gerencia

Grupo LIDER		
Integrantes	Rol a cumplir	Valor Hora (UF)
Felipe Canales	Jefe de proyecto	1,40
Nelson Cariqueo	Jefe del área de diseño	1,20
Camila Vera	Subjefe de proyecto	1,30
Diego García	Jefe del área de desarrollo	1,20

Fuente: Elaboración propia

Tabla 2, Conformación de integrantes del grupo uno

Grupo UNO		
Integrantes	Rol a cumplir	Valor Hora (UF)
Elías Cáceres	Programador/Jefe de grupo	1,00
Gabriel Sanhueza	Programador	0,50
Katherine Sáez	Programador/diseñador	0,70
Mario Urbina	Programador	0,50

Fuente: Elaboración propia

Tabla 3, Conformación de integrantes del grupo dos

Grupo DOS		
Integrantes	Rol a cumplir	Valor Hora (UF)
Nicolás Soto	Programador/Jefe de grupo	1,00
Alvaro Cabedo	Programador/diseñador	0,70
Dominique Huenteleo	Programador	0,50
Boris Vásquez	Programador	0,50

Fuente: Elaboración propia

Tabla 4, Conformación de integrantes del grupo tres

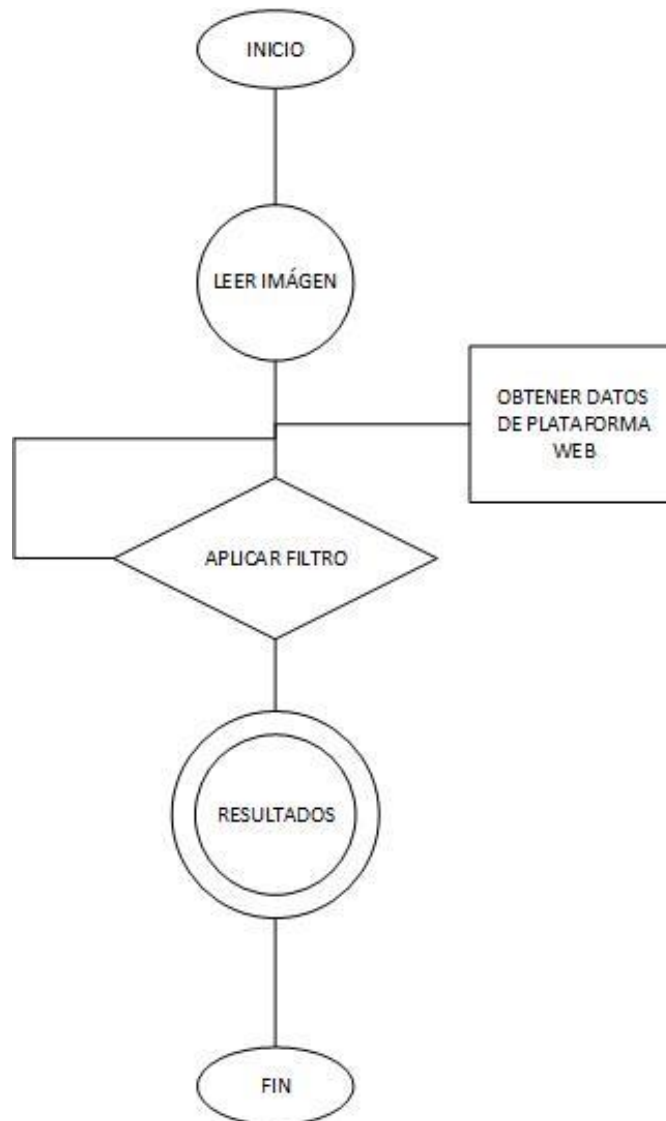
Grupo TRES		
Integrantes	Rol a cumplir	Valor Hora (UF)
Gladys Cerro	Programador/Jefe de grupo	1,00
Felipe Abarca	Programador	0,50
David Martínez	Programador	0,50
Jonathan Acuña	Programador/diseñador	0,70

Fuente: Elaboración propia

## 5.2. Modelo de programación secuencial

La estrategia de programación secuencial para poder dar solución a los 4 problemas principales, será la siguiente (Modelo aplicable a cada diseño de la solución):

*Ilustración 2, Modelo general de programación secuencial*

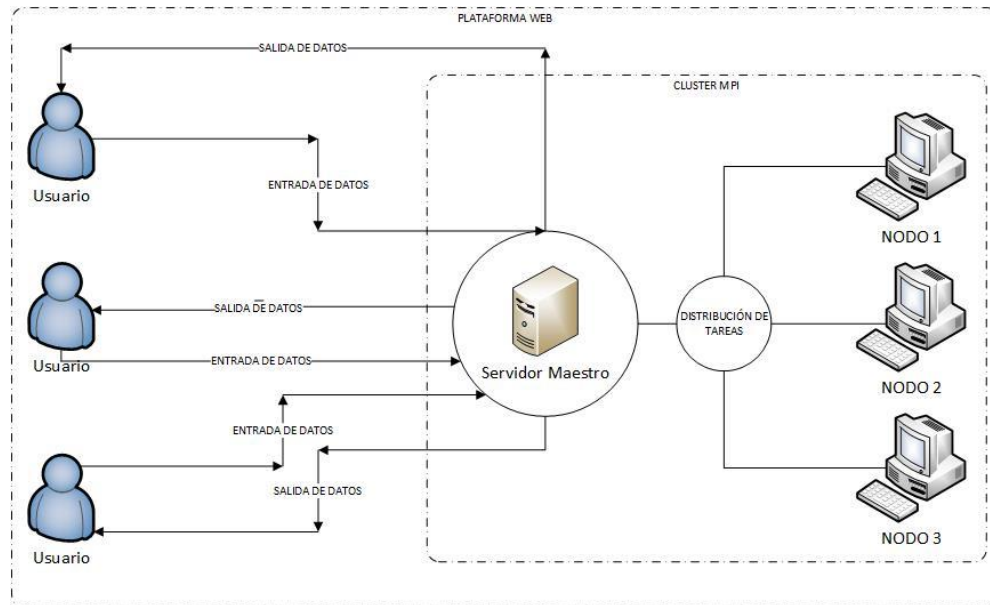


Fuente: Elaboración propia

### 5.3. Modelo de programación paralelo

Para acceder a este modelo, el usuario del sistema se conecta a la dirección que provee el nodo maestro (con la cual se ingresa a la plataforma desde intranet), permitiendo a este poder distribuir el proceso hacia los nodos esclavos. Y en una última instancia, este mismo nodo es el encargado de recopilar la información y entregarla. A continuación, se presenta un modelo para la construcción de la intranet:

Ilustración 3, Modelo lógico de aplicación



Fuente: Elaboración propia

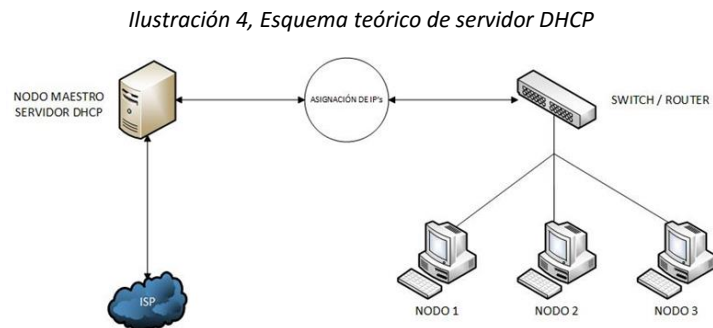
De esta forma, el servidor maestro será configurado dinámicamente con un servidor DHCP (Dynamic Host Protocol), el cual consiste en ser un estándar TCP/IP diseñado para simplificar la administración de la configuración de IP's de la red y lograr realizar esta configuración de forma automática.

Este tipo de servidor consiste en recibir peticiones de clientes (usuarios) solicitando una configuración de red IP. El servidor responderá dichas peticiones otorgando los parámetros para que estos clientes puedan auto configurarse.

El servidor tendrá la misión de proporcionar al menos los siguientes datos:

- Dirección IP.
- Máscara de subred.

El esquema para el desarrollo de este servidor, se ve a continuación:



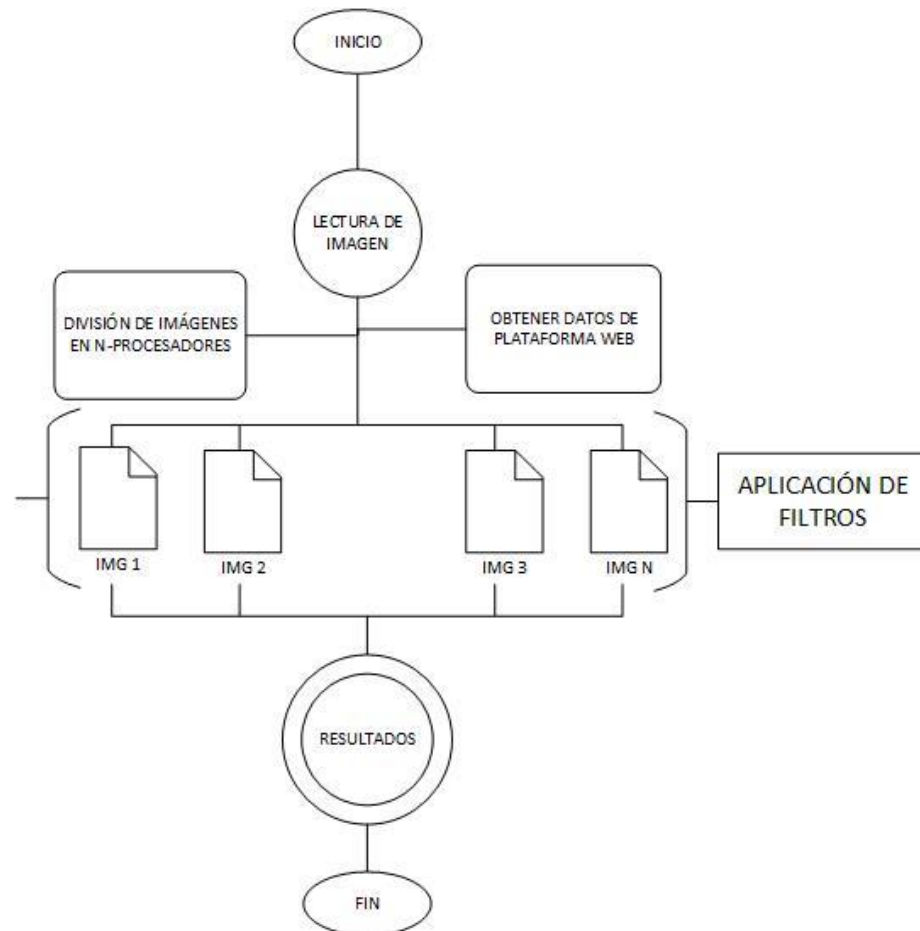
Fuente: Elaboración propia



#### 5.4. Estrategia de paralización

La estrategia para poder conseguir la paralización de los problemas a desarrollar e implementar, en el entorno, queda demostrada en el siguiente modelo:

Ilustración 5, Modelo general de programación paralela



Fuente: Elaboración propia

## 5.5. Web socket

Socket designa un concepto abstracto por el cual dos programas (posiblemente situados en computadoras distintas) pueden intercambiar cualquier flujo de datos, generalmente de manera fiable y ordenada.

El término socket es también usado como el nombre de una interfaz de programación de aplicaciones (API) para la familia de protocolos de Internet TCP/IP, provista usualmente por el sistema operativo.

Los sockets de Internet constituyen el mecanismo para la entrega de paquetes de datos provenientes de la tarjeta de red a los procesos o hilos apropiados. Un socket queda definido por un par de direcciones IP local y remota, un protocolo de transporte y un par de números de puerto local y remoto.

Para que dos programas puedan comunicarse entre sí es necesario que se cumplan ciertos requisitos:

- Que un programa sea capaz de localizar al otro.
- Que ambos programas sean capaces de intercambiarse cualquier secuencia de octetos, es decir, datos relevantes a su finalidad.

Para ello son necesarios los dos recursos que originan el concepto de *socket*:

- Un par de direcciones del protocolo de red (dirección IP, si se utiliza el protocolo TCP/IP), que identifican la computadora de origen y la remota.
- Un par de números de puerto, que identifican a un programa dentro de cada computadora.

Los *sockets* permiten implementar una arquitectura cliente-servidor. La comunicación debe ser iniciada por uno de los programas que se denomina programa "cliente". El segundo programa espera a que otro inicie la comunicación, por este motivo se denomina programa "servidor".

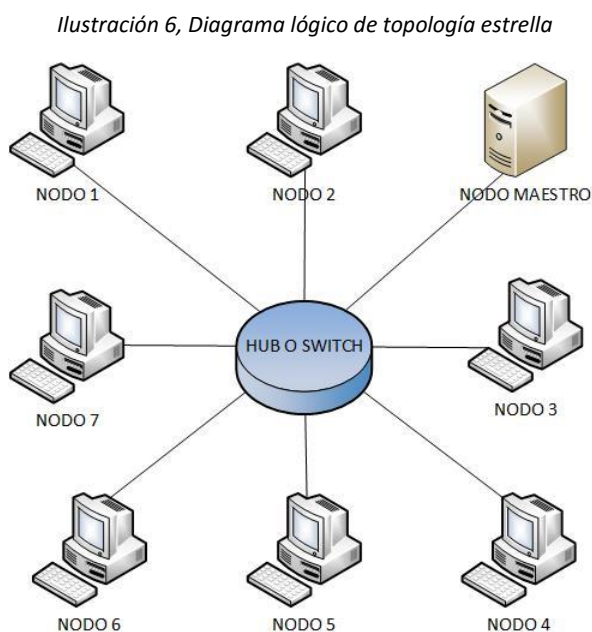
Un *socket* es un proceso o hilo existente en la máquina cliente y en la máquina servidora, que sirve en última instancia para que el programa servidor y el cliente lean y escriban la información. Esta información será la transmitida por las diferentes capas de red.

## 5.6. Topología de red a emplear para la plataforma web

Para poder lograr un funcionamiento correcto del clúster, lo que incluye una distribución de carga coherente y lograr que los equipos se comuniquen con el menor ruido posible. Se hace necesario implementar un estándar de conexión de redes, el cual es llamado topología de red.

Tomando en consideración los implementos con los que se puede contar en el laboratorio de trabajo (llámese equipos, arquitectura, etc.) la topología elegida en este caso es la de tipo estrella, esto es porque el switch/router está conectado a un sistema mayor. Entonces, físicamente y lógicamente es la más correcta.

En la siguiente figura se muestra el esquema topológico que representa la situación actual:



Fuente: Elaboración propia

## 6. Teorización matemática y propuesta de algoritmos

### 6.1. Realzado y filtrado de imágenes

#### 6.1.1. Traspuesta de imagen

Este tipo de modificación de imágenes, forma parte de varias transformaciones geométricas. Por objetivo no cambian el contenido, en sí de la imagen, pero son capaces de deformar la grilla de pixeles para obtener el resultado deseado copiando la modificación en un nuevo destino.

Ilustración 7, Destino de imagen modificada

$$\text{dst}(x, y) = \text{src}(f_x(x, y), f_y(x, y))$$

Fuente: Elaboración propia

Para lograr concebir este efecto, es necesario aplicar la siguiente función:

- `getRotationMatrix2D`: la cual calcula una matriz afín para la rotación 2D. Utilizando la siguiente fórmula matemática, donde: `Center`, corresponde al centro de rotación de la fuente. `Angle`, es la rotación en grados de la imagen. `Scale`, es el factor de escala isotrópico.

Ilustración 8, Matriz de rotación de imagen

$$\begin{bmatrix} \alpha & \beta & (1 - \alpha) \cdot \text{center.x} - \beta \cdot \text{center.y} \\ -\beta & \alpha & \beta \cdot \text{center.x} + (1 - \alpha) \cdot \text{center.y} \end{bmatrix}$$

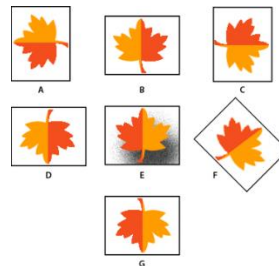
$$\alpha = \text{scale} \cdot \cos \text{angle},$$

$$\beta = \text{scale} \cdot \sin \text{angle}$$

Fuente: Elaboración propia

Fácilmente esta operación es sencilla de implementar en Python, ya que se encuentra de manera inmediata en la librería OpenCV, bajo la siguiente instrucción: `"cv2.getRotationMatrix2D(center, angle, scale)"`, donde sus parámetros, fueron definidos anteriormente.

Ilustración 9, Ejemplo de rotación de imagen



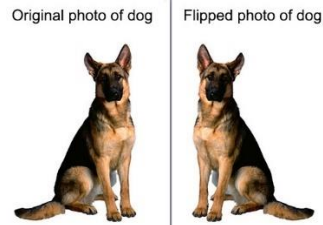
Fuente: Elaboración propia

### 6.1.2. Invertir imagen

Este efecto, perteneciente a las transformaciones geométricas, consiste en voltear una imagen (completa). Con respecto a un eje determinado.

Usada en la vida cotidiana para poder corregir una foto conseguida de manera horizontal o vertical

*Ilustración 10, Ejemplo del efecto invertir imagen*



Fuente: [http://www.lauramfoley.com/wordpress/wp-content/uploads/2013/03/flip-flop\\_dog.jpg](http://www.lauramfoley.com/wordpress/wp-content/uploads/2013/03/flip-flop_dog.jpg)

Matemática está definido de la siguiente manera:

$$dst_{ij} = \begin{cases} SRC_{src.rows-i-1,j} & \text{if flipCode} = 0 \\ SRC_{i,src.cols-j-1} & \text{if flipCode} > 0 \\ SRC_{src.rows-i-1,src.cols-j-1} & \text{if flipCode} < 0 \end{cases}$$

Se trata de copiar los datos de la imagen original, hacia la imagen de destino, siguiendo los operadores algebraicos propuestos.

Afortunadamente, es posible implementar esto de manera inmediata con OpenCV, ya que cuenta con funciones internas para este fin.

### 6.1.3. Redimensionar imagen

La necesidad de incrementar el tamaño de una imagen nos lleva a la "interpolación". Lo que significa añadir píxeles para poder magnificar la imagen.

Cuanto se perderá en esa imagen depende de la calidad de imagen, el método de interpolación y cuánto va a ser ampliada.

En general, la interpolación es el proceso matemático de calcular nuevos datos a partir de los disponibles, procurando que compartan la misma "forma". En imágenes digitales, supone crear nuevos píxeles a partir de los existentes, de modo que la imagen resultante venga a representar lo mismo que la original, sólo que con más píxeles.

En la realidad, la interpolación se usa ampliamente para el trabajo de imágenes digitales, ya sea en tareas de rotación, deformación, corrección, ampliación, reducción e incluso montaje de imágenes.

A modo de ejemplo, se usará la ilustración 11, a la cual se le aplicarán los 3 algoritmos de redimensión, y así apreciar cual ofrece los mejores resultados.

Ilustración 11, Imagen de ejemplo



Fuente: Elaboración propia

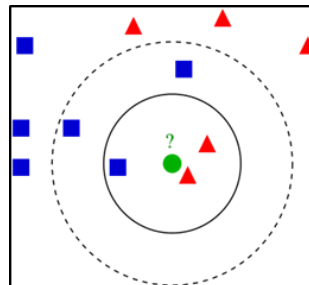
#### 6.1.3.1. Interpolación Nearest Neighbor (vecino más cercano)

Este algoritmo de interpolación utiliza la función matemática Nearest Neighbor o K-vecinos más cercanos, esta función utiliza un patrón para la clasificación de objetos de ciertas características con otros objetos más cercanos (vecinos).

La elección de  $k$  depende de las características de los datos. En general, el aumento de  $k$  disminuye el ruido que afecta a la clasificación, la elección puede ser tomada por medio de técnicas heurísticas tales como la validación cruzada.

Para el cálculo de distancia, por lo general se utiliza la distancia euclidiana, pero también otros tipos de distancia son igualmente utilizables, por ejemplo, la distancia Manhattan.

Ilustración 12, Ejemplo de clasificación por kNN



Fuente: Elaboración propia

En la ilustración 12, se muestra un ejemplo de clasificación por medio de kNN (K-nearest neighbors). El punto en observación es el punto verde. Las clases son dos: Triángulos rojos y cuadrados azules.

Si  $k = 3$  (es decir, son considerados los 3 más cercanos), entonces el punto verde se coloca en los mismos objetos de clase triángulos rojo porque hay 2 triángulos y 1 cuadrado. Si  $k = 5$ , entonces es colocado en la misma clase de cuadrados azules debido a que hay 3 cuadrados y 2 triángulos.

Es así como se puede utilizar Nearest Neighbor en imágenes digitales, en donde en vez de trabajar con cuadrados y triángulos, se utilizan los píxeles para determinar el valor del píxel más cercano.

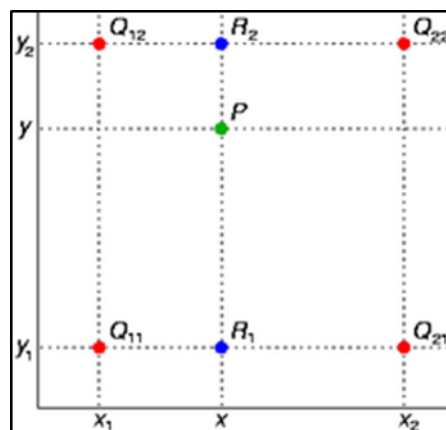
#### 6.1.3.2. Interpolación Bilineal

Este algoritmo toma los valores de cuatro píxeles de alrededor para determinar de qué color serán los nuevos píxeles. Puede hacer buen trabajo para incrementar el tamaño de una imagen, pero en general se utiliza para reducirlo.

La interpolación bilineal es una extensión de la interpolación lineal para interpolar funciones de dos variables (por ejemplo,  $x$  e  $y$ ) en una malla regular de dos dimensiones.

La idea principal es realizar una interpolación lineal en una dirección, y después en la otra. Aunque cada uno de estos pasos es lineal, la interpolación en su conjunto no es lineal sino cuadrática.

Ilustración 13, Ejemplo de interpolación



Fuente: Elaboración propia

Los cuatro puntos rojos muestran los datos conocidos y el punto verde representa el punto que queremos interpolar.

Supóngase que se quiere encontrar el valor para la función  $f$  desconocida en el punto  $P = (x, y)$ . Conocemos el valor de  $f$  en los

cuatro puntos  $Q_{11} = (x_1, y_1)$ ,  $Q_{12} = (x_1, y_2)$ ,  $Q_{21} = (x_2, y_1)$  y  $Q_{22} = (x_2, y_2)$ .

Primero se hace una interpolación lineal en la dirección x. Esto genera:

$$f(R_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21})$$

Donde  $R_1 = (x, y_1)$ ,

$$f(R_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22})$$

Donde  $R_2 = (x, y_2)$ .

Después se hace una interpolación en la dirección y:

$$f(P) \approx \frac{y_2 - y}{y_2 - y_1} f(R_1) + \frac{y - y_1}{y_2 - y_1} f(R_2).$$

Esto proporciona una estimación de  $f(x, y)$ .

$$\begin{aligned} f(x, y) &\approx \frac{f(Q_{11})}{(x_2 - x_1)(y_2 - y_1)} (x_2 - x)(y_2 - y) + \\ &\quad \frac{f(Q_{21})}{(x_2 - x_1)(y_2 - y_1)} (x - x_1)(y_2 - y) + \\ &\quad \frac{f(Q_{12})}{(x_2 - x_1)(y_2 - y_1)} (x_2 - x)(y - y_1) + \\ &\quad \frac{f(Q_{22})}{(x_2 - x_1)(y_2 - y_1)} (x - x_1)(y - y_1) \\ &= \frac{1}{(x_2 - x_1)(y_2 - y_1)} \left( f(Q_{11})(x_2 - x)(y_2 - y) + \right. \\ &\quad \left. f(Q_{21})(x - x_1)(y_2 - y) + \right. \\ &\quad \left. f(Q_{12})(x_2 - x)(y - y_1) + \right. \\ &\quad \left. f(Q_{22})(x - x_1)(y - y_1) \right). \end{aligned}$$

Aplicando interpolación bilineal, se obtiene el siguiente resultado:

Ilustración 14, Resultado



Fuente: Elaboración propia



#### 6.1.3.3. Interpolación Bicubic

La interpolación bicubic es un método más lento, pero más preciso al realizar un promedio “ponderado” de los pixeles adyacentes. El resultado es una gradación tonal mucho más suave que con los métodos anteriores. Es el método más usado en fotografía y artes gráficas.

El núcleo de interpolación para la interpolación bicubic está dado por:

Ilustración 15, Función bicubic

$$h(x) = \begin{cases} 1 - 2|x|^2 + |x|^3 & \text{si } 0 \leq |x| < 1 \\ 4 - 8|x| + 5|x|^2 - |x|^3 & \text{si } 1 \leq |x| < 2 \\ 0 & \text{cc} \end{cases}$$

Fuente: Elaboración propia

Al aplicar este algoritmo, se obtendrá el siguiente resultado:

Ilustración 16, Resultado de algoritmo bicubic



Fuente: Elaboración propia

#### 6.1.4. Convolución con gestión de bordes

La convolución está relacionada con la matemática de convolución, la cual, básicamente dadas una o dos funciones, se produce como resultado una tercera.

Este tipo de efecto es aplicado como una operación de matrices. Sin embargo, al operar como una simple multiplicación, tiene una manera distinta de operar, como se logra apreciar en lo siguiente:

$$\left( \begin{bmatrix} a & v & i \\ q & e & l \\ \sigma & p & c \end{bmatrix} * \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \right) [s' s] = (i * 1) + (v * 2) + (a * 3) + (l * 4) + (e * 5) + (q * 6) + (c * 7) + (p * 8) + (\sigma * 9)$$

Entonces a convolución, básicamente es una operación de multiplicación sobre la matriz (imagen) principal y otra (cualquiera) llamada kernel.

En OpenCV la convolución con gestión de bordes se realiza implementando la función Canny, la cual, implementa el operador convolución dado un kernel y realiza la multiplicación especial de los índices

#### 6.1.5. Enfoque y Desenfoque

Este tipo de herramientas, son usadas comúnmente al momento de la edición de imágenes digitales. Estas herramientas son utilizadas es para los casos en que un objeto de la imagen resalta demasiado, en comparación con el resto o, en caso contrario, centrar la atención en un punto que no destaca mucho. Sin embargo, en el caso del enfoque, utilizar demasiado el filtro de enfoque puede generar ruido en la imagen por lo que, si se abusa de este, puede arruinar la imagen con la que se esté trabajando.

Al igual que los anteriores, ambos filtros trabajan con un operador kernel, l cuál trabaja con toda la imagen.

En el caso del desenfoque, el caso más utilizado es del tipo linear, el cual trabaja con cada uno de los pixeles de la imagen. Está basado en la siguiente teoría:

$$g(i, j) = \sum_{k, l} f(i + k, j + l)h(k, l)$$

Donde:

- $i, j$ : corresponden a los índices de la matriz de la imagen.
- $h(k, l)$ : corresponde al kernel a aplicar, que no es más que un coeficiente de intensidad del filtro.

OpenCV cuenta con operadores propios para poder realizar todo este trabajo.

Para el caso del efecto de enfoque, también es apoyado por un operador de kernel, sin embargo, en lugar de ser una matriz de ceros o unos, es de un valor mayor positivo, para poder obtener el realce esperado. Apoyado por la función propia “filter2D” la cual permite aplicar el kernel a la imagen objetivo

#### 6.1.6. Efecto espejo

Este efecto duplica la imagen en un eje deseado. Se debe seleccionar el tipo de simetría que se desea aplicar, la cual puede ser vertical, horizontal o diagonal. Como lo dice su nombre, este efecto simula un espejo.

*Ilustración 17, Efecto espejo en eje vertical*

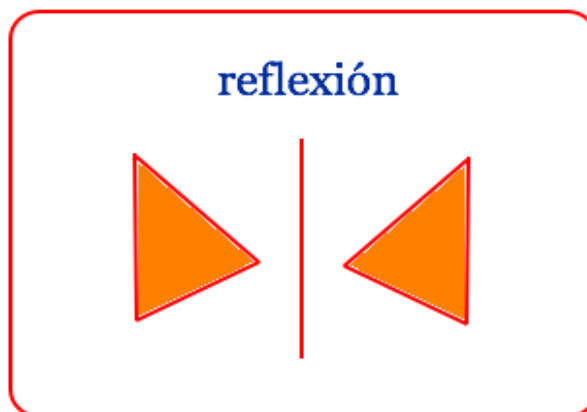


Fuente: <https://pantinesdistinto.wordpress.com/2008/11/07/efecto-espello-efecto-espejo-effect-mirror-effet-miroir/>

La forma de poder desarrollar este tipo de algoritmo es la siguiente: primero se debe indicar un eje de simetría. Luego, copia a partir del eje indicado los pixeles de un lado del eje en el otro lado.

La teoría matemática asociada, corresponde a utilizar la simetría por reflexión (o simetría especular) para realizar la copia de pixeles de la imagen respecto del eje indicado.

*Ilustración 18, Simetría por reflexión*



Fuente: <http://www.disfrutalasmaticas.com/geometria/simetria.html>

#### 6.1.7. Efecto luminosidad

Este tipo de efecto, permite agregar luminosidad (o brillo) a ciertos objetos de una imagen. Se seleccionan los objetos deseados y se le agrega luminosidad.

*Ilustración 19, Efecto luminosidad*



Fuente:

como primera parte se recorre la matriz (una imagen corresponde a una matriz de píxeles) utilizando una herramienta para el reconocimiento de objetos, que nos permite visualizar una máscara (formada de los objetos seleccionados). Una esquina o córner se caracteriza por ser una región de la imagen con cambios de intensidad en diferentes direcciones.

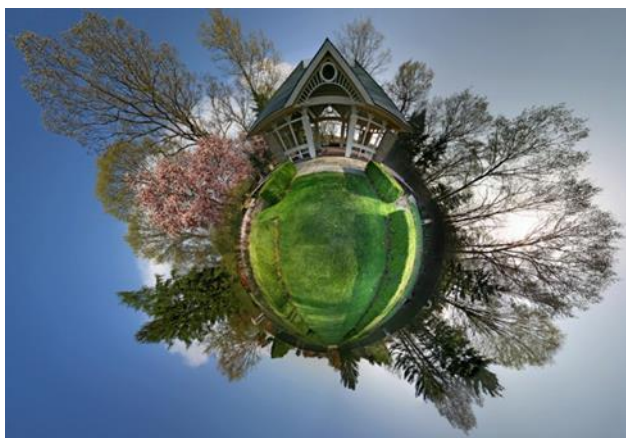
El método consiste en mirar los cambios de intensidad que se producen en cada píxel dentro de una imagen para luego aplicar brillo a estos objetos, esto de la siguiente manera: se utiliza en cada pixel la función Brightness (perteneciente a la librería PIL de Python), la cual consiste en sumar una constante a cada color que constituye un pixel.

Matemáticamente, en este tipo de efecto se utiliza la función Harris Corner Detection para la detección de esquinas y de esta forma reconocer los objetos a los cuales se les aplicará el brillo. Luego se utiliza la función Brightness para darle brillo a los objetos.

#### 6.1.8. Efecto Polar

¿Qué sucede si tomamos una imagen tomada en 360° y la cerramos sobre sí misma, con el conocido efecto “Polar” o “Coordenadas Polares”? El resultado se puede apreciar en la siguiente imagen.

*Ilustración 20, Efecto polar*

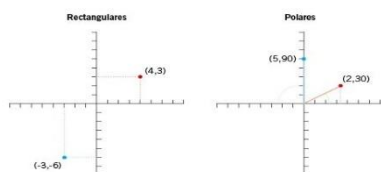


Fuente: <http://www.guerra-creativa.com/img/uploads/images/oc302hm0.jpg>

El efecto polar es capaz de convertir las coordenadas rectangulares de una selección en coordenadas polares y viceversa. O sea, este filtro sirve para transformar una imagen rectangular en una imagen cilíndrica, en la que la imagen distorsionada da el efecto de que se está viendo en un espejo cilíndrico.

Polar es un sistema de coordenadas en el que cada punto se determina en función de la distancia y el ángulo respecto al eje de coordenadas, mientras que en el sistema rectangular o cartesiano el punto se determina en función de la distancia al eje coordenadas. En la imagen inferior es posible comparar ambos sistemas. A la izquierda el sistema rectangular, en el que cada punto se determina indicando la distancia que lo separa de los ejes de coordenadas (estos pueden ser un eje o más de uno). El sistema rectangular puede determinar la posición de un punto en una, dos o tres dimensiones. Puede observarse como en el sistema polar tanto la distancia como el ángulo determinan la posición respecto del eje horizontal. El sistema de coordenadas polares sólo puede señalar la posición de los puntos en un plano bidimensional, para indicar un punto en un espacio tridimensional debe apoyarse en coordenadas esféricas y cilíndricas.

*Ilustración 21, Coordenadas rectangulares vs. polares*



Fuente: Elaboración propia

El valor de  $y$  del documento se convertirá en la distancia al centro al convertirse al sistema polar, siendo la parte superior del lienzo el punto de origen y la inferior el punto más alejado. Respecto al valor  $x$  se transforma en ángulo respecto al eje de coordenadas horizontal situado en el centro del documento al realizar la conversión. De izquierda a derecha empezando por  $90^\circ$ .

Como se puede observar en la imagen inferior una sencilla forma de conseguir círculos concéntricos es dibujar unas líneas horizontales y transformarlas a coordenadas polares.

*Ilustración 22, Ejemplo de conversión*



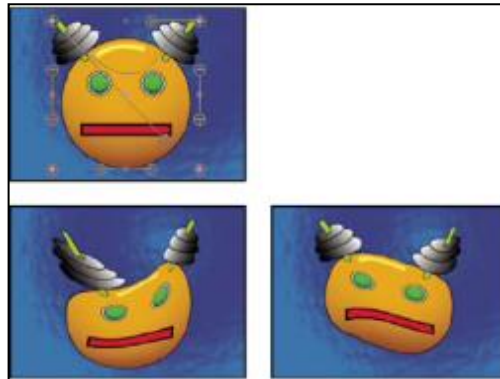
Fuente: Elaboración propia

En python gracias a las librerías PIL, numpy, scipy se puede lograr el efecto polar debido a que con las librerías matemáticas presentes en python tienen métodos con los que es posible poder modificar una matriz, por lo que solo se debe convertir la imagen a formato matriz para poder trabajarla.

#### 6.1.9. Efecto deformación Bézier

Se trata de dar forma a una imagen mediante una curva de Bézier cerrada a lo largo del límite de una capa. La curva se compone de cuatro segmentos. Cada segmento tiene tres puntos (un vértice y dos tangentes).

*Ilustración 23, Efecto de deformación Bézier*



Fuente: <https://helpx.adobe.com/es/after-effects/using/distort-effects.html#meshwarpeffect>

Imagen original (arriba a la izquierda) y con variaciones aplicadas del efecto Deformación Bézier (abajo a la izquierda y la derecha).

La algoritmia asociada, consiste en lo siguiente: Primero se deben reconocer los bordes (de la imagen completa) los cuales son representados en una región con 4 segmentos. Al mover los vértices de los segmentos la imagen se deforma dando al borde la forma de la Curva de Bézier generada por el movimiento del vértice.

De manera matemática, estas Curvas de Bézier, corresponden a un sistema que se desarrolló sobre 1960 para realizar dibujos técnicos en el ámbito profesional. Su nombre es en honor de quién ideó un método de descripción matemática de las curvas, Pierre Bézier.

Estas curvas pueden ser generadas por medio de la librería Spicy de Python.

#### 6.1.10. Efecto posicionar los bordes

Los bordes de una imagen digital se pueden definir como transiciones entre dos regiones de niveles de gris significativamente distintos. Suministran una valiosa información sobre las fronteras de los objetos y puede ser utilizada para segmentar la imagen, reconocer objetos, etc. La mayoría de las técnicas para detectar bordes emplean operadores locales basados en distintas aproximaciones discretas de la primera y segunda derivada de los niveles de grises de la imagen.

La derivada de una señal continua proporciona las variaciones locales con respecto a la variable, de forma que el valor de la derivada es mayor cuanto más rápidas son estas variaciones. En el caso de funciones bidimensionales  $f(x,y)$ , la derivada es un vector que apunta en la dirección de la máxima variación de  $f(x,y)$  y cuyo módulo es proporcional a dicha variación. Este vector se denomina gradiente y se define:

$$\nabla f(x,y) = \begin{bmatrix} \frac{\partial f(x,y)}{\partial x} \\ \frac{\partial f(x,y)}{\partial y} \end{bmatrix} \quad \text{Mag}[\nabla f(x,y)] = \sqrt{\left(\frac{\partial f(x,y)}{\partial x}\right)^2 + \left(\frac{\partial f(x,y)}{\partial y}\right)^2}$$

$$\theta = \arctan \frac{\frac{\partial f(x,y)}{\partial x}}{\frac{\partial f(x,y)}{\partial y}}$$

En el caso bidimensional discreto, las distintas aproximaciones del operador gradiente se basan en diferencias entre los niveles de grises de la imagen. La derivada parcial  $f_x(x,y)$  (gradiente de fila  $GF(i,j)$ ) puede aproximarse por la diferencia de píxeles adyacentes de la misma fila.

$$\frac{\partial f(x,y)}{\partial x} \approx \nabla_x f(x,y) = f(x,y) - f(x-1,y)$$

La discretización del vector gradiente en el eje Y ( $GC(i,j)$ ), será:

$$\frac{\partial f(x,y)}{\partial y} \approx \nabla_y f(x,y) = f(x,y) - f(x,y-1)$$



El gradiente de la fila GF y de columna GC en cada punto se obtienen mediante la convolución de la imagen con las máscaras HF y HC, esto es:

$$G_F(i, j) = F(i, j) \otimes H_F(i, j)$$

$$G_C(i, j) = F(i, j) \otimes H_C(i, j)$$

La magnitud y orientación del vector gradiente suele aproximarse por la expresión:

$$|G(i, j)| = \sqrt{G_F^2 + G_C^2} \approx |G_F(i, j)| + |G_C(i, j)|$$

Los operadores más utilizados son los de Roberts, Prewitt y Sobel y Frei-chen.

#### 6.1.10.1. Operador Roberts

Las máscaras utilizadas en este operador son:

Gradiente fila

0	0	0
0	0	1
0	-1	0

Gradiente columna

-1	0	0
0	1	0
0	0	0

Obtiene buena respuesta ante bordes diagonales. Ofrece buenas prestaciones en cuanto a localización. El gran inconveniente de este operador es su extremada sensibilidad al ruido y por tanto tiene pobres cualidades de detección.

#### 6.1.10.2. Operador Prewitt y Sobel y Frei-chen.

Los tres operadores pueden formularse de forma conjunta con las siguientes máscaras de convolución mostradas a continuación.

Gradiente fila

$\frac{1}{2+K}$	1	0	-1
	K	0	-K
	1	0	-1

Gradiente columna

$\frac{1}{2+K}$	-1	-K	-1
	0	1	0
	1	K	1

En el operador Prewitt (K=1) se involucran a los vecinos de filas / columnas adyacentes para proporcionar mayor inmunidad al ruido.

El operador Sobel ( $K=2$ ), se supone que es más sensible a los bordes diagonales que el de Prewitt, aunque en la práctica hay poca diferencia entre ellos.

Frei-Chen ( $K=2$ ), el gradiente es el mismo para bordes verticales, horizontales y diagonales.

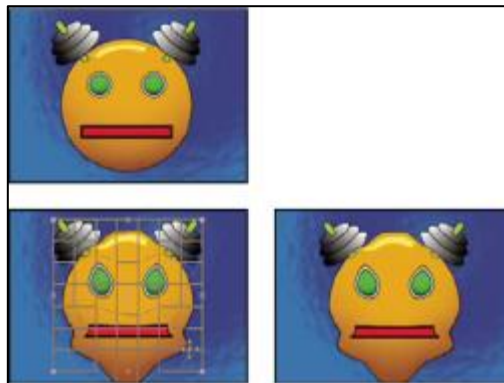
#### 6.1.11. Efecto deformación de malla

El efecto de deformación de malla, aplica una cuadrícula de parches Bézier sobre una capa, que se puede manipular para distorsionar zonas de una imagen. Cada esquina de un parche incluye un vértice y de dos a cuatro tangentes (puntos que controlan la curvatura del segmento de una línea que compone el borde del parche). El número de tangentes depende de si el vértice está en una esquina, en un borde o dentro de la cuadrícula. Al mover los vértices y las tangentes, se podrá manipular la forma de un segmento de líneas curvadas. Cuanto más definida esté la cuadrícula, más intensos serán los ajustes que se puedan realizar en el área de la imagen del interior del parche.

Este efecto se suele utilizar para convertir un par de imágenes con el fin de crear una transición de una imagen a otra.

Como se aprecia en la siguiente ilustración: Imagen original (arriba a la izquierda), con una malla de distorsión (abajo a la izquierda) y con el efecto Deformación de malla aplicado (abajo a la derecha).

*Ilustración 24, Efecto de deformación de malla*



Fuente: <https://helpx.adobe.com/es/after-effects/using/distort-effects.html#meshwarpeffect>

El algoritmo asociado se define de la siguiente manera, primero se deben reconocer los bordes. Luego, se forma una malla representativa de la imagen. Esta malla no es más que la repetición de la región que representa la imagen completa en el efecto deformación de Bézier. Al mover de su lugar un vértice de la malla, el borde se mueve según la Curva de Bézier generada con el movimiento del vértice.

Matemáticamente, al igual que el efecto deformación Bézier, se utilizan las Curvas de Bézier, que pueden ser generadas por medio de la librería Scipy de Python.

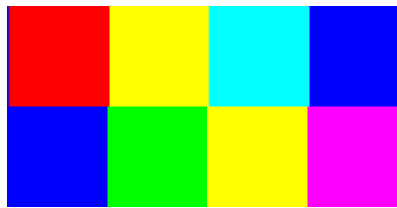
#### 6.1.12. Inversión de colores

El efecto de inversión de colores, básicamente, se trata de invertir todos los colores (valores) de los píxeles que afecta. Las áreas oscuras se vuelven claras y las claras, oscuras. En el caso de las tonalidades, estas son reemplazadas por el complemento de su color.

Este filtro solo funciona en imágenes RGB y escala de grises.

La teoría detrás de este efecto, como ya se mencionó con anterioridad, es calcular el complemento del color, sobre el píxel que se esté trabajando. Al tratarse de imágenes RGB, la intensidad de color de cada píxel viene dado por un rango comprendido entre cero y 255. Entonces: Dado un píxel con la siguiente configuración: (123, 34, 100) el complemento de este será (132, 221, 155)

*Ilustración 25, A la izquierda la imagen original, y a la derecha el filtro aplicado.*



Fuente: <https://docs.gimp.org/es/gimp-layer-invert.html>

#### 6.1.13. Escala de grises

La escala de grises es la representación de una imagen en la que cada pixel se dibuja usando un valor numérico individual que representa su luminancia, en una escala que se extiende entre blanco y negro.

Se trata de una forma supuestamente más profesional para designar lo que la gente no relacionada con las artes gráficas llama "fotos en blanco y negro". Con esta denominación se las distingue de las imágenes formadas por líneas que, en fotografía digital, se llaman "imágenes de mapa de bits". De hecho, las escalas de gris también son mapas de bits.

En tratamiento digital de las imágenes, el término "escala de grises" se aplica al modo de color en el que cada elemento (píxel o de otro tipo) se describe dentro de una serie limitada de valores de un mismo tono neutro. En el caso de imágenes con una profundidad de 8 bits, los valores posibles teóricos son 256.

En topografía y artes gráficas, la escala de gris es una serie de parches de tonos neutros que va aumentando de intensidad del más claro (blanco) al más oscuro (negro). Dependiendo de su uso, el número de parches suele variar en torno a los diez. Estas escalas de grises se usan, como las tiras de color, para evaluar la neutralidad y fidelidad de los tonos reproducidos.

En OpenCV, esto es aplicable con una función propia de esta librería, la cual consiste en cambiar una imagen en formato BGR (otro tipo de formato de imagen similar al RGB) a una a escala de grises.

#### 6.1.14. Efecto sepia

Dentro de toda la gama de efectos que se disponen, de la edición de imágenes digitales, este tipo de efectos es el más común y el mayormente utilizado. Su principal uso es para darle impacto y drama a las imágenes que dispongan. Antiguamente era utilizado para poder dar algo de color a las imágenes en blanco y negro.

La implementación el efecto sepia en OpenCV, viene dada por lo que se conoce como "sepia kernel", básicamente es una matriz con colores en BGR predefinidos con una tonalidad que se pueda considerar como sepia y este trabaja mediante todos los pixeles de imagen, modificando su tonalidad.

#### 6.1.15. Efecto RGB

Siglas de Red, Green, y Blue, (rojo, verde y azul). RGB es un modelo de color utilizado normalmente para presentar color en los sistemas de video, cámaras, y monitores de ordenadores. Representa todos los colores como combinaciones de rojo, verde y azul. RGB es el sistema de color más utilizado por la mayoría de los formatos actuales.

*Ilustración 26, Colores primarios*



Fuente:

Este diagrama muestra cómo la combinación de los colores primarios produce los colores secundarios, con el blanco en el centro. En términos resumidos:

1. Los colores primarios son el rojo, el verde y el azul.
2. Los colores secundarios son:
  - Amarillo (incluidos los tonos amarillos, anaranjados y pardos o marrones), que se obtiene mezclando el rojo y el verde.
  - Cian, que se obtiene mezclando el verde y el azul.
  - Magenta, que se obtiene mezclando el rojo y el azul.
3. El gris se obtiene mezclando tres colores (que no sean los colores primarios) en cantidades iguales.
4. El blanco se obtiene mezclando los tres colores primarios en sus cantidades máximas.
5. El negro es la ausencia de colores primarios.

Los colores primarios del modelo RGB son aditivos, es decir, para producir el resultado se suman las contribuciones individuales de cada primario. Como todos sabemos los colores básicos son el R RED (ROJO) G GREEN (VERDE) Y EL B BLUE (AZUL) y de la mezcla de estos y su aplicación con mayor o menor intensidad obtenemos el resto de colores; encontrándonos en los

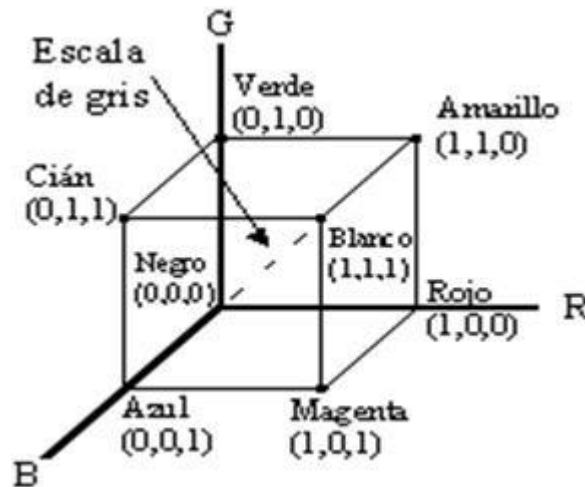
extremos con el blanco y el negro; es decir un color cuyo valor RGB es 0,0,0 es el color negro, y el color cuyo valor es 255, 255, 255 es el color blanco, en este punto cabe señalar que los valores RGB son expresados en una combinación de 3 cifras cuyos valores mínimo y máximo son 0 y 255 respectivamente. Mezclando los diferentes valores de un modo u otro obtendremos los diferentes colores como se describe

gráficamente a continuación: (255,0,0) ( 0,255,0) (0,0,255 ) (255,128,0) ( 125,23,46) ... y así todas las combinaciones posibles que podamos realizar con los 3 valores.

Representando el modelo en un cubo unitario, cada componente de la tripleta toma valores en el rango de 0 a 1. Así un color Cz se expresa:

$$Cz = RR + GG + BB$$

Ilustración 27, Representación de 3D de un pixel



Fuente:

Asumiendo que al utilizar números enteros en el rango 0...255 (un byte), es decir 256 valores posibles para cada color primario, entonces con tres bytes se puede representar 256 x 256 x 256 combinaciones diferentes de los colores primarios, es decir una gama de más de 16 millones de colores (frecuentemente referida como "color verdadero").

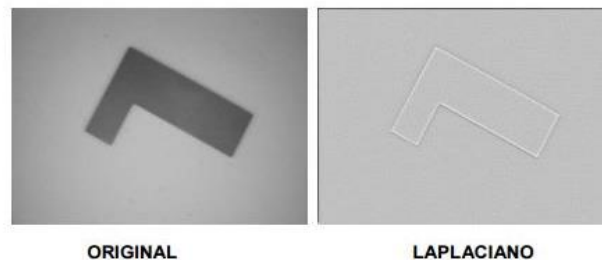
En Python el tratamiento matemático de matrices con la librería numpy y la librería PIL o Opencv permiten convertir una imagen a formato RGB gracias a los métodos .split o .convert, con los cuales separar una imagen convertida en array a la paleta de colores RGB se vuelve trivial para poder trabajarla en el formato anteriormente explicado (R, B, G) obteniendo de esta forma cualquier paleta o combinación de colores deseada.

### 6.1.16. Efecto LoG

El efecto log realza los bordes de una imagen digital en color. Este efecto implica una serie de efectos espaciales de suavizado en el espacio de color acromático y con colores oponentes para poder aproximar las funciones de sensibilidad al contraste del sistema visual humano. o sea, los filtros son combinaciones lineales de máscaras gaussianas.

El filtro Laplaciano es una medida 2-D isotrópica de la 2da derivada espacial de una imagen. El Laplaciano de una imagen destaca las regiones donde hay cambios bruscos de intensidad y por tanto se suele utilizar para detección de bordes. El Laplaciano se aplica frecuentemente a una imagen que previamente ha sido suavizada mediante un filtro Gaussiano de suavizado, con el fin de reducir su sensibilidad al ruido.

Ilustración 28, Efecto LoG



Fuente: Elaboración propia

Este tipo de filtro se basa en un operador derivativo, por lo que acentúa las zonas que tienen gran discontinuidad en la imagen

$$\frac{\partial f(x)}{\partial x} = f(x+1) - f(x)$$

$$\frac{\partial^2 f(x)}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$$

Si la función depende de dos variables

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

La derivada de segundo orden con respecto a la variable x:

$$\frac{\partial^2 f(x, y)}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$$

De forma similar, con respecto a y:

$$\frac{\partial^2 f(x, y)}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

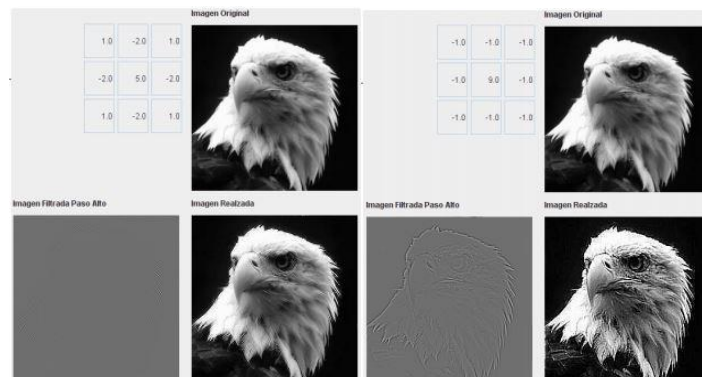
$$\nabla^2 f(x, y) = f(x+1, y) + f(x-1, y) - 2f(x, y) + f(x, y+1) + f(x, y-1) - 2f(x, y)$$

$$\nabla^2 f(x, y) = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

La anterior expresión es equivalente a aplicar una máscara definida por:

$$w = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$f(x-1, y-1)$	$f(x-1, y)$	$f(x-1, y+1)$
$f(x, y-1)$	$f(x, y)$	$f(x, y+1)$
$f(x+1, y-1)$	$f(x+1, y)$	$f(x+1, y+1)$



La forma para poder implementarlo en Python es con la ayuda de una librería llamada “Matlab”, a través de una serie de pasos primeramente se realiza un filtrado gaussiano a la imagen para luego realizar un filtrado Laplaciano.



#### 6.1.17. Efecto 3D de imágenes 2D

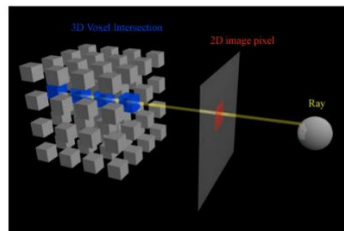
El término gráficos 3D por computadora (o por ordenador) se refiere a trabajos de arte gráfico que son creados con ayuda de software y programas especiales en general, o de términos, de técnicas y tecnología relacionadas con los gráficos de dimensiones altamente concentradas pueden ser grises o azul con rojo.

Un gráfico 3D difiere de uno bidimensional principalmente por la forma en que ha sido generado. Este tipo de gráficos se originan mediante un proceso de cálculos matemáticos sobre entidades geométricas tridimensionales producidas en un ordenador, y cuyo propósito es conseguir una proyección visual en dos dimensiones para ser mostrada en una pantalla o impresa en papel.

En general, el arte de los gráficos tridimensionales es similar a la escultura o la fotografía, mientras que el arte de los gráficos 2D es análogo a la pintura. En los programas de gráficos por computadora esta distinción es a veces difusa: algunas aplicaciones 2D utilizan técnicas 3D para alcanzar ciertos efectos como iluminación, mientras que algunas aplicaciones 3D primarias hacen uso de técnicas 2D.

Render volumétrico: La técnica utilizada para el renderizado de imágenes 3D con transparencia en la aplicación es un ray-casting. Por cada pixel de la pantalla, se tira un rayo de forma normal al plano que la forma, y se computa el color del pixel de salida, como la acumulación de transparencia en la de los pixeles del volumen que se encuentran en la dirección del rayo dado, tal como se esquematiza en la siguiente figura.

Ilustración 29, Ejemplo de renderizado volumétrico



Fuente: Elaboración propia

La etapa de modelado consiste en ir dando forma a objetos individuales que luego serán usados en la escena creada. Existen diversos tipos de geometría para modelador con NURBS y modelado poligonal o subdivisión de superficies. Además, aunque menos usado, existe otro tipo llamado "modelado basado en imágenes" o en inglés *image based modeling* (IBM). Consiste en convertir una fotografía a 3D mediante el uso de diversas técnicas, de las cuales, la más conocida es la fotogrametría cuyo principal impulsor es Paul Debevec.

Modelado de superficies este no tiene curvas en calculadas en cada línea ejemplo SketchUp.

#### 6.1.18. Efecto “zooming”

El efecto “zooming” es una técnica que consiste en hacer un barrido de distancias focales con el objetivo zoom de nuestra cámara mientras realizamos una fotografía con un tiempo de exposición medianamente largo.

Esto provoca una distorsión en el contorno de la fotografía con unas líneas convergentes que crean un efecto de movimiento hacia el centro del encuadre y que permiten transmitir desplazamiento, velocidad o simplemente centrar la atención del espectador.

*Ilustración 30, Ejemplo de efecto zooming*



Fuente: Elaboración propia

La manera de conseguir este efecto es solo de manera física, usando la cámara y el zoom de ella, consiste en seleccionar una velocidad baja, menor de 1/15 sg. y mover el zoom mientras se está disparando. Es conveniente usar el trípode y con esto se consigue que sólo quede enfocada la parte central y el resto quede desenfocado.

*Luego de investigar a fondo el concepto de zooming, se llegó a la conclusión de que no hay una librería que contenga este tipo de filtro de imagen en python con OpenCv, ya que no hay una operación matemática identificable detrás para lograr crear un algoritmo que haga tal efecto.*

*Existen otras técnicas que pueden generar un filtro con cierta similitud a lo que se pide con zooming, mas no es el efecto como tal.*

*Se encontró una forma de generar un efecto así extrapolando las fronteras de una imagen rotada en un ángulo distinto de 90° o 180° para que la frontera quede desencajada y se pueda extrapolar quedando un efecto como el que se muestra en la siguiente imagen.*

*Ilustración 31, Extrapolación de fronteras de imagen*



Fuente: Elaboración propia

## 6.2. Mejora de imagen

### 6.2.1. Mejora de brillo y contraste

La modificación de brillo y contraste es uno de los ajustes más sencillos que se pueden realizar en una fotografía y que nos puede dar unos resultados realmente buenos.

El brillo se puede definir como la tendencia al blanco o a la luz que tiene una imagen.

Si se aumenta demasiado el brillo, la imagen adopta un tono blanquecino. Si se baja, la imagen presentará un tono oscuro.

El contraste es la diferencia de luz entre zonas iluminadas y zonas oscuras.

Si se aumenta el contraste, las partes luminosas lo serán más, y las oscuras se oscurecerán más, creando una mayor diferencia de luminosidad que la imagen original. Al subir el contraste se perderán tonos intermedios y se pueden sobresaturar los colores, con lo que se alterarán los colores.

No conviene abusar de excesivos contrastes o la imagen presentará un aspecto irreal.

Si se baja demasiado el contraste, la imagen presentará un aspecto totalmente agrisado.

En OpenCV, este efecto es aplicado a cada pixel de la imagen a editar, siguiendo la siguiente fórmula matemática:

$$g(x) = \alpha f(x) + \beta$$

Donde el parámetro  $\alpha$  y  $\beta$  corresponden, respectivamente, a los niveles de brillo y contraste,  $f(x)$  es la imagen a editar y  $g(x)$  es a imagen de destino.

Convenientemente, la función anterior se puede representar (según el interés de este laboratorio) como:

$$g(i, j) = \alpha \cdot f(i, j) + \beta$$

Donde los parámetros  $i, j$  corresponden a la fila  $i$  y columna  $j$  de la matriz imagen.

### 6.2.2. Mejora de HDR

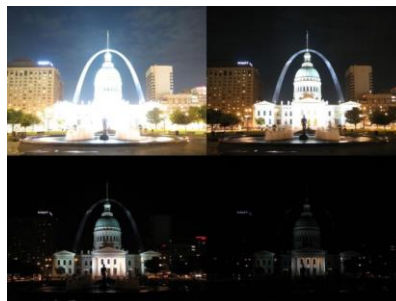
En procesamiento de imágenes, gráficos por ordenador y fotografía, las imágenes de alto rango dinámico (HDR) son un conjunto de técnicas que permiten un mejor rango dinámico de luminancias entre las zonas más claras y las más oscuras de una imagen del que las técnicas de imagen digital estándar o métodos fotográficos pueden ofrecer. La fotografía de alto rango dinámico permite obtener imágenes más acordes a las visualizadas por el ojo humano, que cuando observa el mundo que nos rodea es capaz de distinguir detalles en zonas con una diferencia de iluminación muy superior a la soportada por otros formatos como la película o los formatos de imágenes comprimidas. Este rango dinámico más extenso permite a las imágenes HDR representar con más exactitud el extenso rango de niveles de intensidad encontrados en escenas reales, que van desde luz solar directa hasta la débil luz de las estrellas.

Los dos principales orígenes de las imágenes HDR son el renderizado por ordenador y la mezcla de múltiples fotografías, que a su vez son conocidas como fotografías de bajo rango dinámico (LDR),<sup>2</sup> también llamadas de rango dinámico estándar (SDR).

Las técnicas de mapeo tonal (mapeo de tonos), que reducen todo el contraste para facilitar que dispositivos con menos rango dinámico muestren imágenes HDR, pueden aplicarse para producir imágenes conservando o exagerando el contraste localmente para realizar un efecto artístico.

Para poder realizar la fusión de las imágenes (que se muestran en la ilustración 26), OpenCV cuenta con un operador llamado Mertens fusión, el cual, de la mejor manera realiza esta operación.

*Ilustración 32, Imagen con distinto grado de exposición fotográfica*



Fuente: [http://docs.opencv.org/3.1.0/d2/df0/tutorial\\_py\\_hdr.html#gsc.tab=0](http://docs.opencv.org/3.1.0/d2/df0/tutorial_py_hdr.html#gsc.tab=0)

Una vez realizada la operación con la lista de imágenes, para este filtro, el resultado obtenido es como el que sigue la imagen:

*Ilustración 33, Resultado de filtro HDR*



Fuente: [http://docs.opencv.org/3.1.0/d2/df0/tutorial\\_py\\_hdr.html#gsc.tab=0](http://docs.opencv.org/3.1.0/d2/df0/tutorial_py_hdr.html#gsc.tab=0)

## 6.3. Movimiento de imagen

### 6.3.1. Suma de fotos

Para hacer una operación de suma de fotos, se necesitan 2 fotos del mismo tamaño y tipo, o na imagen y un valor escalar. Según la teoría de Thomas Young, en la retina disponemos de tres sensores para la percepción del color, cada uno de ellos, sensible, a una de estas tres longitudes de onda: rojo anaranjado, verde y azul violáceo. Estos son los tonos básicos o primarios Red, Green, Blue. Si representamos esto en un ordenador, podemos aplicar el modelo RGB computacionalmente representando cada pixel como un trio de números de 8 bits, lo que nos da un total de 24 bits por pixel (color verdadero.)

Como disponemos de 24 bits, y necesitamos representar 3 tonos básicos, cada uno de ellos requiere 8 bits ni más ni menos para ser representado. Esto conlleva que cada tono tiene 256 posibles intensidades (2 elevado a 8), al juntar los 8 bits de cada tono, obtenemos un color, o para nuestro caso: un pixel.

Como ya establecidos un modelo de representación, podemos hacer operaciones aritméticas con esa matriz que almacena los colores. Para la suma de matrices solo tenemos que sumar los valores de cada pixel de una matriz con los valores de la matriz de la otra imagen y formar una matriz nueva con dichos resultados.

A modo de ejemplo: si se tienen dos pixeles uno con valor #12 AC DE y se suman #D5 25 25, se obtiene como resultado #E7 D1 FF.

Ilustración 34, Adición de colores

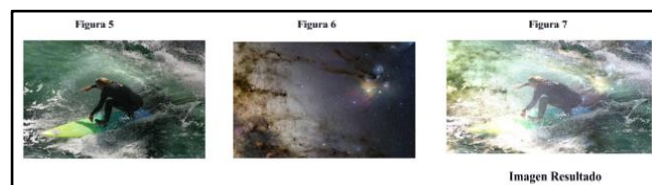


Fuente: Elaboración propia

Como se puede observar, en la suma de los “Blue” de esos 2 pixeles, en suma, hexadecimal DE+25 no es FF, sino que es 103, pero como nuestro sistema sólo contempla hasta el valor 255, se establece como resultado de DE+25 como FF.

Ahora un ejemplo del mismo concepto con 2 imágenes:

Ilustración 35, Adición de imágenes



Fuente: Elaboración propia

Nótese como la imagen final parece ser más brillante, esto es consecuencia de la suma de pixeles.

### 6.3.2. Time lapse

El time-lapse es una técnica fotográfica muy popular usada en cinematografía y fotografía para mostrar diferentes motivos o sucesos que por lo general suceden a velocidades muy lentas e imperceptibles al ojo humano. El efecto visual que se logra en el time-lapse consiste en que todo lo que se haya capturado se mueva muy rápidamente, como puede ser el movimiento de las nubes, la apertura de una flor, una puesta de sol, etc.

Existen dos métodos base para la realización de videos con la técnica time-lapse:

- Se filma una o varias secuencias de video con una cámara, para posteriormente acelerarla en cualquier programa de edición de video (posproducción).
- Se capturan varias fotografías o imágenes fijas a determinados intervalos de tiempo y en la postproducción se unen y se les asigna una velocidad específica. De esta forma se logra el efecto time-lapse.

Tomando en cuenta lo anterior, en la posproducción se pueden emplear libremente diferentes velocidades de fotogramas por segundo (FPS) para crear la ilusión del time-lapse de acuerdo a las necesidades de cada quien o del efecto que se quiera mostrar. Las siguientes son frecuencias de fotograma de algunos de los sistemas más conocidos:

- Cine mudo: 16–18 FPS
- Cine actual: 24 fps
- Televisión, normas europeas (PAL & SECAM): 25 FPS
- Televisión, norma estadounidense (NTSC): 30 FPS

Para poder construir este tipo de movimiento de imágenes, ya sea en OpenCV u otro tipo de librerías similares, basta con unir las imágenes según la velocidad que se desea. Este tipo de construcción, cuenta para los dos algoritmos siguientes, que se revisarán a continuación

#### 6.3.3. Bullet time

Bullet time es un concepto introducido recientemente en el cine y videojuegos actuales. Consiste en una extremada ralentización del tiempo para permitir ver movimientos o sucesos muy veloces (como el recorrido de una bala). También es usado para crear efectos dramáticos, como en la película Matrix.

En un principio el bullet time era creado mediante una serie de cámaras fotográficas fijas colocadas sobre una barra de sujeción de manera que apuntaran al sujeto u objeto a fotografiar desde diferentes situaciones y sin mucho espacio entre ellas. Estas cámaras realizaban múltiples fotografías y luego esas imágenes eran montadas como una sucesión de fotogramas en la que se intercalaban las imágenes que habían capturado las sucesivas cámaras. De esta manera se hacían varias fotografías para que posteriormente fueran montadas al software de edición fotográfica, intercalando las diferentes tomas al uso de un time lapse. Luego las imágenes se aceleraban a velocidades de 24 fotogramas por segundo para conseguir el efecto de ralentización adecuado, con el efecto de la cámara desplazándose lateralmente mientras enfoca el mismo sujeto.

Más tarde la técnica fue mejorada y perfeccionada utilizando cámaras de vídeo de alta definición y la postproducción digital, aunque en esencia la técnica sigue siendo la misma.

Esta técnica es mucho más fácil de usar con imágenes que han sido totalmente creadas por ordenador, como en videojuegos y películas de animación en 3D. Para conseguir efectos más espectaculares hay películas que mezclan las imágenes capturadas por las cámaras con imágenes creadas completamente por ordenador.

El único inconveniente para realizar esta técnica a nivel semi-profesional sería la necesidad imprescindible de una gran cantidad de cámaras fotográficas, ya que la sincronización no es un problema (porque podría utilizarse un sincronizador inalámbrica o un cable disparador sincronizado).

#### 6.3.4. Stop motion

El stop motion es una técnica de animación que consiste en aparentar el movimiento de objetos estáticos por medio de una serie de imágenes fijas sucesivas. Stop motion es un anglicismo que empezó a penetrar en el uso de la lengua española en tiempos relativamente recientes. Tradicionalmente se lo ha designado en español con muchos términos diferentes como, por ejemplo, fotoanimación, animación en volumen, parada de imagen, paso de manivela, animación foto a foto, animación fotograma por fotograma o animación cuadro por cuadro, por citar sólo algunos ejemplos.

En general se usan esos términos o se denomina «animaciones en stop motion» a las que no entran ni en la categoría del dibujo animado ni en la de la animación por ordenador; esto es, que no fueron dibujadas o pintadas ni a mano ni por ordenador, sino que fueron creadas tomando imágenes de la realidad. Así pues, el stop motion se utiliza para producir movimientos animados de cualquier objeto, ya sea rígido o maleable, como por ejemplo juguetes, bloques de construcción, muñecos articulados o personajes creados con plastilina.

Podría también definirse esta técnica como animación artesanal, debido a que se construye el movimiento manipulando un objeto, con las propias manos, fotograma a fotograma. Se trabajan diversos materiales (plastilina, arena, recortes de papel, tizas sobre suelos y muros...) de manera progresiva, hacia adelante, sin posibilidad alguna de retroceso.



## 6.4. Operaciones matemáticas

### 6.4.1. Dispersiones gaussianas

El desenfoque Gaussiano es un efecto de suavizado para mapas de bits generado por software de edición gráfica, que desenfoca objetos. Este filtro puede añadir brillos y reflejos realistas de los dibujos, así como un efecto suavizante en las imágenes que puede suavizar el ruido en las fotografías o quitar énfasis en el fondo de una imagen, lo que provoca una pérdida minuciosa en los detalles. Este filtro modifica el color de los píxeles estableciendo el valor de cada uno como el promedio entre los valores de todos los píxeles incluidos en forma de radio definido en el cuadro de diálogo (o sea, a mayor radio mayor desenfoque).

La intensidad del desenfoque se define internamente en términos del llamado "radio de desenfoque", que, para aquellos lectores con inclinaciones matemáticas, podemos decir que coincide con la desviación estándar de la distribución gaussiana. Matemáticamente, una función gaussiana se extiende hasta el infinito. sin embargo, por razones prácticas, el límite del desenfoque de un objeto está a una distancia doble del "radio de desenfoque" a partir de la caja de contorno del objeto, para una intensidad máxima del desenfoque.

La función de la distribución Gaussiana tiene la forma  $f(x) = ae^{-(x-b)^2/c^2}$  donde  $(x-b)$  es la distancia entre el píxel que está siendo modificado y el píxel del que se está tomando el valor,  $a$  y  $c$  valen 1, y  $f(x)$  resulta un número Real entre 0 y 1 que representa el porcentaje de influencia.

Y todo eso es sólo para calcular uno de los tres componentes (Rojo, Verde o Azul) de un sólo píxel. El efecto de desenfoque Gaussiano realiza una cantidad muy elevada de cálculos (que aumenta junto al Radio y al tamaño de la imagen).

La implementación del efecto gaussiano se puede utilizar en python utilizando la librería OPENCV.

#### 6.4.2. Transformada rápida de Fourier

FFT es la abreviatura usual (del inglés Fast Fourier Transform) de un eficiente algoritmo que permite calcular la transformada de Fourier discreta (DFT) y su inversa. La FFT es de gran importancia en una amplia variedad de aplicaciones, desde el tratamiento digital de señales y filtrado digital en general a la resolución de ecuaciones diferenciales parciales o los algoritmos de multiplicación rápida de grandes enteros. El algoritmo pone algunas limitaciones en la señal y en el espectro resultante.

Este es uno de los algoritmos aritméticos más ampliamente utilizados es la transformada rápida de Fourier, un medio eficaz de ejecutar un cálculo matemático básico y de frecuente empleo. La transformada rápida de Fourier es de importancia fundamental en el análisis matemático y ha sido objeto de numerosos estudios.

Las aplicaciones de la transformada rápida de Fourier son múltiples. Es la base de muchas operaciones fundamentales del procesamiento de señales, donde tiene amplia utilización. Además, proporciona un medio oportuno para mejorar el rendimiento de los algoritmos para un conjunto de problemas aritméticos comunes.

Partiendo del hecho de que es una función definida en el campo de los Reales y que tiene un periodo L entonces la serie de Fourier es:

$$\frac{1}{2}a_0 + \sum_{n=1}^{\infty} \left[ a_n \cos\left(\frac{n\pi x}{L}\right) + b_n \text{Sen}\left(\frac{n\pi x}{L}\right) \right]$$

Consideremos que esta expresión es integrable en un intervalo  $\left[-\frac{L}{2}, \frac{L}{2}\right]$  y llamemos  $w_0 = \frac{2\pi}{L}$  entonces:

$$\frac{1}{2}a_0 + \sum_{n=1}^{\infty} \left[ a_n \cos(nw_0 x) + b_n \text{Sen}(nw_0 x) \right]$$

Por otro lado, la forma de exponencial compleja de las funciones seno y coseno es:

$$\operatorname{sen}(nw_0x) = \frac{1}{2}(e^{inw_0x} + e^{-inw_0x})$$

$$\operatorname{cos}(nw_0x) = \frac{1}{2}(e^{inw_0x} - e^{-inw_0x})$$

Entonces:

$$\frac{1}{2}a_0 + \sum_{n=1}^{\infty} \left[ a_n \frac{1}{2}(e^{inw_0x} + e^{-inw_0x}) + b_n \frac{1}{2}(e^{inw_0x} - e^{-inw_0x}) \right]$$

Reorganizando la expresión:

$$\frac{1}{2}a_0 + \sum_{n=1}^{\infty} \left[ \frac{1}{2}(a_n - ib_n)e^{inw_0x} + \frac{1}{2}(a_n + ib_n)e^{-inw_0x} \right]$$

Tomemos ahora:

$$d_0 = \frac{1}{2}a_0$$

$$d_n = \frac{1}{2}(a_n - ib_n), \forall n \in \mathbb{Z}^+$$

$$\overline{d}_n = \frac{1}{2}(a_n + ib_n), \forall n \in \mathbb{Z}^+$$

Remplazando en la serie resulta:

$$\frac{1}{2}d_0 + \sum_{n=1}^{\infty} [d_n e^{inw_0x}] + \sum_{n=1}^{\infty} [\overline{d}_n e^{-inw_0x}]$$

Así:

$$d_0 = \frac{1}{2} a_0 = \frac{1}{L} \int_{-L/2}^{L/2} f(t) dt \text{ para } n = 1, 2, 3, \dots$$

Para  $d_n$  tenemos que:

$$d_n = \frac{1}{2} (a_n - ib_n)$$

$$d_n = \frac{1}{2L} \int_{-L/2}^{L/2} f(x) \cos(nw_0 x) dx - \frac{i}{2L} \int_{-L/2}^{L/2} f(x) \sin(nw_0 x) dx$$

$$d_n = \frac{1}{L} \int_{-L/2}^{L/2} f(x) [\cos(nw_0 x) - i \sin(nw_0 x)] dx$$

$$d_n = \frac{1}{L} \int_{-L/2}^{L/2} f(x) e^{-inw_0 x} dx$$

Por lo tanto, para:

$$\overline{d_n} = \frac{1}{L} \int_{-L/2}^{L/2} f(x) \overline{e^{-inw_0 x}} dx = \frac{1}{L} \int_{-L/2}^{L/2} f(x) e^{inw_0 x} dx$$

$$\overline{d_n} = d_{-n}$$

Con base en esto se tiene que:

$$f(x) = d_0 + \sum_{n=1}^{\infty} d_n e^{inw_0 x} + \sum_{n=1}^{\infty} \bar{d}_n e^{-inw_0 x}$$

$$f(x) = d_0 + \sum_{n=1}^{\infty} d_n e^{inw_0 x} + \sum_{n=1}^{\infty} d_{-n} e^{-inw_0 x}$$

$$f(x) = d_0 + \sum_{n=-\infty, n \neq 0}^{\infty} d_n e^{inw_0 x}$$

Que corresponde a la forma compleja de la serie de Fourier. Formalmente la definimos de la siguiente manera:

Sea  $f$  una función con periodo fundamental  $L$ . Sea  $w_0 = \frac{2\pi}{L}$ . Entonces la serie de Fourier compleja de es:

$$\sum_{n=-\infty}^{\infty} d_n e^{inw_0 x}$$

Donde:

$$d_n = \frac{1}{L} \int_{-\frac{L}{2}}^{\frac{L}{2}} f(x) e^{-inw_0 x} dx$$

Con  $n = 0, \pm 1, \pm 2, \pm 3$  valores de  $d_n$  son los coeficientes complejos de  $f$ .

#### 6.4.3. Convolución – Desconvolución

Una parte muy importante de los procesos de mejora se basan en la aplicación de funciones, cuyo resultado depende únicamente de los niveles de intensidad de cada píxel de la imagen, pero no de la posición dentro de la imagen, que permiten resaltar determinados elementos de la imagen (como resaltar los contornos de los objetos), mejorar el enfoque o reducir el ruido de fondo.

Utilizando los conceptos anteriores mediante la FT obtendremos las correspondientes transformadas a la imagen y al filtro.

- Imagen  $f(x, y)$   $\rightarrow F(u, v)$  su correspondiente FT
- Filtro  $g(u, v) \rightarrow G(u, v)$  su correspondiente FT

Cumpléndose por las propiedades de la Transformada de Fourier

$$f(x, y) \otimes g(u, v) = F(u, v) \cdot G(u, v)$$

Siendo bastante más sencillo obtener dicha convolución por la simple multiplicación de dos transformadas de Fourier y su posterior conversión al dominio espacial por la transformada inversa.

Un filtro de convolución, para una imagen digital, en el espacio real  $(X, Y)$ , puede representarse como una matriz cuadrada o rectangular (matriz de convolución), de dimensiones  $(M, N)$  mucho más pequeñas que la de la imagen. La matriz de convolución se desplaza sobre la imagen de tal forma que el elemento central de la matriz de convolución coincida con cada uno de los píxeles de la imagen. En cada posición, se multiplica el valor de cada píxel de la imagen, que coincide en posición con un elemento de la matriz de convolución, por el valor de éste. El píxel de la imagen, que coincide con el elemento central de la matriz de convolución, es substituido por la suma de los productos.

La mayoría de los filtros usan una matriz de convolución, se pueden crear filtros personalizados. Convolución es el tratamiento de una matriz por otra que se llama “kernel”.

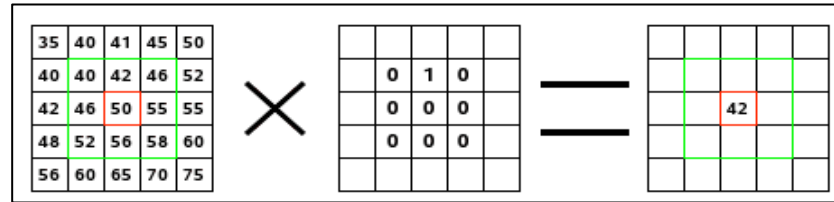
El filtro matriz de convolución usa una primera matriz que es la imagen que será tratada. La imagen es una colección bidimensional de píxeles en coordenada rectangular. El kernel que se usa depende del efecto deseado.

Se considerarán sólo las matrices  $3 \times 3$ , son las más usadas y son suficiente para los efectos deseados. Si todos los valores del borde de un kernel se seleccionan a cero, el sistema la considerará como una matriz  $3 \times 3$ .

El filtro examina, sucesivamente, cada píxel de la imagen. Para cada uno de ellos, que se llamará “píxel inicial”, se multiplica el valor de este píxel y los valores de los 8 circundantes por el valor correspondiente del kernel.

Entonces se añade el resultado, y el píxel inicial se regula en este valor resultante final.

Ilustración 36, Ejemplo de matriz de convolución



Fuente: Elaboración propia

A la izquierda, la imagen de la matriz: cada píxel está marcado con su valor. El píxel inicial tiene un borde rojo. El área de acción del kernel tiene un borde verde. En el medio, el kernel, y a la derecha, el resultado de convolución.

El filtro lee sucesivamente, de izquierda a derecha y de arriba a abajo, todos los píxeles del área de acción del kernel. Se multiplica el valor de cada uno de ellos por el valor correspondiente del kernel y se suman los resultados. El píxel inicial llega a 42:  $(40*0)+(42*1)+(46*0) + (46*0)+(50*0)+(55*0) + (52*0)+(56*0)+(58*0) = 42$ . (el filtro no trabaja sobre la imagen sino sobre una copia). Como resultado gráfico, el píxel inicial se movió un píxel hacia abajo.

Deconvolución no es más que el nombre de una operación matemática. En sus diversas implementaciones prácticas, se usa en restauración de señales para recuperar datos que han sido degradados por algún proceso físico que pueda describirse matemáticamente mediante la operación inversa, una convolución. Es el caso de imágenes de dispositivos ópticos como los empleados en microscopía o astronomía, pero también en otros tipos de sensores, como los de medidas sísmicas.

En microscopía, por ejemplo, la convolución modeliza matemáticamente el proceso de formación de una imagen que sufre degradación por desenfoque y ruido. El desenfoque aparece inevitablemente en el límite de resolución del dispositivo debido a la difracción de la luz en las lentes. El ruido es normalmente "ruido fotónico", un término que se refiere a las variaciones de flujo inherentes a las propiedades estadísticas de los fotones. Puede haber otros ruidos superpuestos, pero este es inevitable.

Para caracterizar la calidad óptica de un microscopio podemos determinar cuán borrosa se vuelve la imagen de un objeto luminoso casi puntual cuando se adquiere con él. Este grado de "emborronamiento" de un simple punto es una medida de la calidad del dispositivo óptico. La imagen tridimensional que se registra de tal fuente puntual se llama función de dispersión del punto (PSF, de Point Spread Function).

El proceso de formación de la imagen es lineal, y se puede describir mediante teoría de sistemas lineales. Esto significa que cuando las imágenes de dos objetos A y B se registran simultáneamente, el resultado es equivalente a la suma de sus imágenes registradas independientemente. En otras palabras: la imagen de A no se ve afectada por la de B, y viceversa.

Debido a esta propiedad de linealidad la imagen de cualquier objeto se puede "calcular" cortando el objeto en partes pequeñas, registrando sus imágenes independientemente, y después sumando todos los resultados. Cuando se divide el objeto en partes extremadamente minúsculas, en partículas puntuales de diferentes intensidades, la imagen se puede concebir como la suma de muchas PSF, cada una situada en la posición de cada partícula y ajustada su intensidad de acuerdo con la del punto correspondiente. La imagen total (borrosa) surge de la suma de todas las contribuciones de los puntos borrosos.

Resumiendo: la formación de la imagen en un dispositivo óptico queda descrita por completo mediante su PSF.

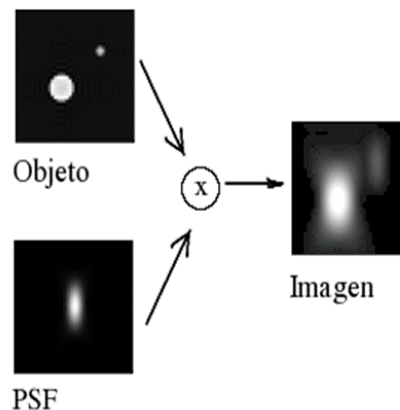
Podemos por lo tanto imaginar que la imagen se forma en el dispositivo óptico reemplazando cada fuente de luz puntual (o al menos de tamaño menor que la resolución del aparato) por su PSF, multiplicada por la intensidad correspondiente.

Este proceso se describe matemáticamente con una ecuación de convolución:

$$g = f * h$$

Donde la imagen  $g$  surge de la convolución de las fuentes reales de luz  $f$  (el objeto) y la PSF  $h$ . El operador de convolución  $*$  implica una integral en todo el espacio:

Ilustración 37, Aplicación del PSF



Fuente: Elaboración propia



$$h(\vec{x}) = (f * g)(\vec{x}) = \int f(\vec{r})g(\vec{x} - \vec{r})d^3\vec{r}$$

Se puede interpretar esta ecuación como sigue: la imagen registrada  $h(\vec{x})$  se compone de vóxeles (en el caso general tridimensional) o de píxeles (en el caso 2D de la imagen de un telescopio) situados en diversas coordenadas  $\vec{x} = (x, y, z)$ . La intensidad registrada en cada vóxel se debe a las distintas contribuciones de todos los puntos luminosos del objeto  $f$ , estando ponderadas sus intensidades por la PSF  $g$  en función de la distancia al punto considerado. Cuanto más lejos esté la fuente luminosa, menor será su contribución a la intensidad registrada localmente.

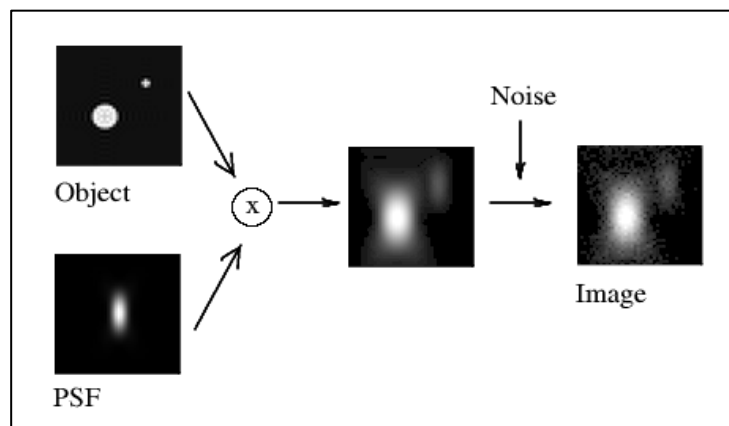
Por lo tanto, si la convolución implica reemplazar cada fuente luminosa puntual original por su correspondiente PSF para producir una imagen borrosa, el proceso de deconvolución sigue el camino inverso, recolectando toda la luz dispersa y poniéndola en su sitio de nuevo.

Esto produce una mejor representación del objeto real, más clara a nuestros ojos. (El brillo de la imagen parece disminuir mientras que aumenta el contraste entre objetos y fondo, y el rango dinámico se agranda).

En términos matemáticos, la deconvolución es simplemente la resolución de la ecuación mostrada más arriba, donde conocidas la imagen convuelta  $g$  y la PSF  $h$ , se obtendría la distribución de luz original  $f$ : una representación del objeto "real".

$$k = f * h + \epsilon$$

Ilustración 38, Desconvolución



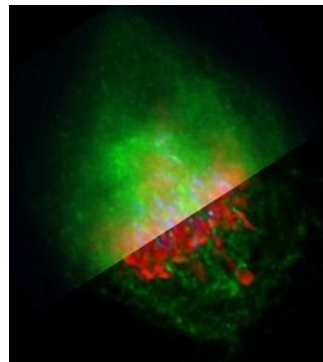
Fuente: Elaboración propia

Donde la imagen registrada  $k$  proviene de la convolución de las fuentes reales  $f$  y la PSF  $h$ , más el ruido fotónico  $\epsilon$ .

La división de  $\epsilon$  entre  $H$  conduciría a una amplificación extrema del ruido y a la aparición de artefactos, debido a las regiones de  $H$  en que ésta tiene valores próximos a cero. (Por otra parte, no podemos sustraer  $\epsilon$  de  $k$ , porque no conocemos cuál es la distribución exacta del ruido).

En conclusión, la inversión del filtro no nos permite recuperar la función del objeto real  $f$ . En vez de esto debemos encontrar una estimación de  $f$  que satisfaga algún criterio racional preestablecido, y que sea estable en presencia de ruido.

Ilustración 39, Resultado final



Fuente: Elaboración propia

#### 6.4.4. Laplaciano de la Gaussiana

Para comenzar, es necesario saber el por qué es importante el cálculo de derivadas en una imagen. Imaginemos que queremos detectar los bordes presentes en la siguiente imagen:

Ilustración 40, Imagen de ejemplo para la detección de bordes

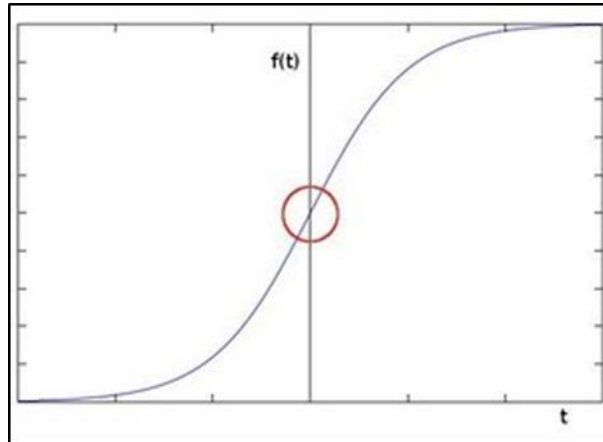


Fuente: Elaboración propia

Se puede notar fácilmente que en un borde (ver el círculo rojo), la intensidad de píxeles cambia de forma notoria. Una buena manera de expresar los cambios es mediante el uso de derivadas. Un alto cambio de pendiente indica un cambio importante en la imagen.

Supongamos que tenemos una imagen-1D. Un borde se muestra por el "salto" en la intensidad de la trama a continuación:

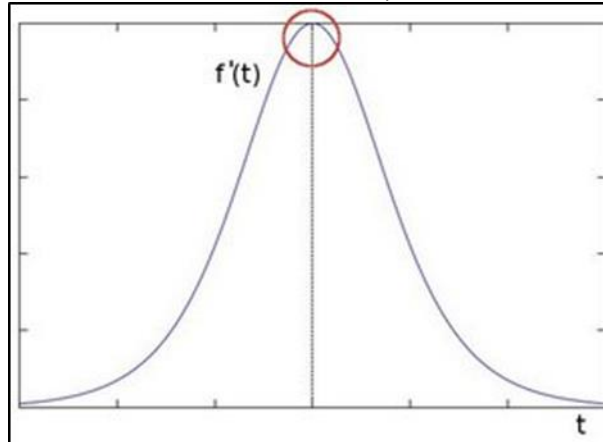
Ilustración 41, Representación matemática de un borde



Fuente: [http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/sobel\\_derivatives/sobel\\_derivatives.html](http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/sobel_derivatives/sobel_derivatives.html)

El borde "salto" se puede apreciar mejor si tomamos la primera derivada:

Ilustración 42, Primera derivada de la curva representada en la ilustración 26

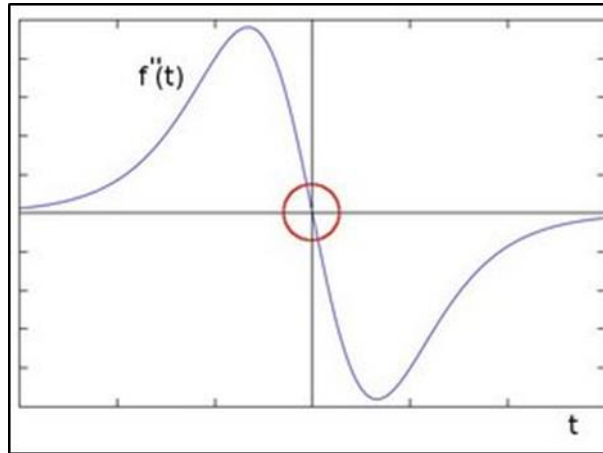


Fuente: [http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/sobel\\_derivatives/sobel\\_derivatives.html](http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/sobel_derivatives/sobel_derivatives.html)

Así, a partir de la explicación anterior, se puede deducir que un método para detectar bordes en una imagen se puede realizar mediante la localización de ubicaciones de píxel donde el gradiente es más alto que sus vecinos.

A partir de la primera derivada, se puede observar cómo un borde se caracteriza por un máximo, pero, ¿Qué ocurre si tomamos la segunda derivada?, como se muestra en la siguiente figura:

*Ilustración 43, Segunda derivada de la curva en la ilustración 9*



Fuente: [http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/laplace\\_operator/laplace\\_operator.html](http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/laplace_operator/laplace_operator.html)

Se puede notar que la segunda derivada es cero. Por lo tanto, también podemos utilizar este criterio para intentar detectar los bordes de una imagen. Sin embargo, debemos tener en cuenta que los ceros no sólo aparecerán en los bordes; esto puede ser resuelto mediante la aplicación de filtrado cuando sea necesario.

*Ilustración 44, Efecto del operador Laplaciano*



Fuente: [http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/laplace\\_operator/laplace\\_operator.html](http://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/laplace_operator/laplace_operator.html)

Este operador se basa en el reconocimiento de regiones, es decir, su objetivo es detectar puntos o regiones más claras o más oscuras en una imagen.

El algoritmo asociado es el siguiente: primero se carga la imagen. Posteriormente se elimina el ruido mediante un desenfoque gaussiano y luego se convierte la imagen original a escala de grises. Por último, se aplica un operador de Laplace a la imagen en escala de grises y ésta es almacenada.

Cabe destacar que el efecto de desenfoque gaussiano mezcla ligeramente los colores de los píxeles que estén vecinos el uno al otro en una imagen, lo que provoca que pierda algunos detalles minúsculos y de esta forma, ésta se vea más suave (aunque menos nítida o clara). Se genera un efecto similar al de una fotografía tomada con una cámara fotográfica desenfocada.

Cuya función matemática asociada es: Operador de Laplace. Como se dijo anteriormente, la segunda derivada se puede utilizar para detectar bordes. Dado que las imágenes están en “2D” (dos dimensiones), tendríamos que tomar la derivada en ambas dimensiones. En este caso, el operador laplaciano resulta muy útil.

El operador laplaciano se define por:

$$Laplace(f) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

## 7. Cronograma

Tabla 5, Cronograma de actividades

ID	Actividad	Plan Inicio	Plan Termino	Duración Planf.	
				Horas	Días
1	Reunion N°1	24-05-2016	24-05-2016	1	1
2	Planificacion de Actividades	24-05-2016	26-05-2016	18	3
3	Division de tareas e informacion a investigar	26-05-2016	26-05-2016	2	1
4	Investigacion para propuesta	26-05-2016	31-05-2016	24	3
5	Confeccion de la propuesta	26-05-2016	02-06-2016	48	6
6	Entrega de Informe de Propuesta	03-06-2016	03-06-2016	0	1
7	Reunion N°2	03-06-2016	03-06-2016	1	1
8	Instalacion de Ubuntu 14.04	06-06-2016	06-06-2016	4	1
9	Instalacion y configuracion del Software necesario	06-06-2016	06-06-2016	4	1
10	Construccion del cluster	07-06-2016	07-06-2016	4	1
11	Ejecutar programa de prueba en cluster	07-06-2016	07-06-2016	4	1
12	Investigación de diferentes algoritmos para la gestión de imagen	06-06-2016	07-06-2016	16	2
13	Reunión N°3	08-06-2016	08-06-2016	1	1
14	Modelar solucion de ambiente secuencial	08-06-2016	10-06-2016	24	3
15	Programacion solucion de ambiente secuencial	13-06-2016	13-06-2016	8	1
16	Optimizar solucion de ambiente secuencial	14-06-2016	14-06-2016	8	1
17	Calcular metricas de solucion de ambiente secuencial	15-06-2016	15-06-2016	4	1
18	Documentacion y analisis de QA de ambiente secuencial	15-06-2016	15-06-2016	4	1
19	Confeccion Informe Avance N°1	15-06-2016	16-06-2016	16	2
20	Entrega Informe Avance N°1	17-06-2016	17-06-2016	0	1
21	Reunión N°4	17-06-2016	17-06-2016	1	1
22	Modelar solucion de ambiente paralelo	17-06-2016	21-06-2016	20	3
23	Programacion solucion de ambiente paralelo	21-06-2016	22-06-2016	12	2
24	Optimizar solucion de ambiente paralelo	22-06-2016	23-06-2016	16	2
25	Confeccion Informe Avance N°2	21-06-2016	23-06-2016	24	3
26	Entrega Informe Avance N°2	24-06-2016	24-06-2016	0	1
27	Reunion N°5	24-06-2016	24-06-2016	1	1
28	Calcular metricas de solucion de ambiente paralelo	24-06-2016	24-06-2016	7	1
29	Documentacion y analisis de QA de ambiente paralelo	27-06-2016	27-06-2016	4	1
30	Comparar métricas algoritmo secuencial versus paralelo	27-06-2016	27-06-2016	4	1
31	Modelar plataforma web	24-06-2016	27-06-2016	16	2
32	Integración de la aplicación paralela a la plataforma web	28-06-2016	29-06-2016	16	2
33	Reunion N°6	30-06-2016	30-06-2016	1	1
34	Programación interna de plataforma web	30-06-2016	01-07-2016	16	2
35	Modelar Interfaz gráfica de plataforma web	01-07-2016	04-07-2016	16	2
36	Reunion N°7	04-07-2016	04-07-2016	1	1
37	Programación de la interfaz gráfica de plataforma web	04-07-2016	04-07-2016	7	1
38	Montar el servidor para la plataforma	05-07-2016	05-07-2016	5	1
39	Reunion N°8	05-07-2016	05-07-2016	1	1
40	Confeccion Informe Final de Laboratorio Fase II	05-07-2016	07-07-2016	24	3
41	Entrega Informe Final de Laboratorio Fase II	08-07-2016	08-07-2016	0	1

Fuente: Elaboración propia

## 8. Cuantificación de esfuerzo

Tabla 6, Tabla de Cuantificación de esfuerzos

ID	Actividad	Grupo LIDER															
		Felipe Canales				Camila Vera				Nelson Cariqueo				Diego García			
		% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	140	140	100%	1	130	130	100%	1	120	120	0%	1	120	120
2	Planificacion de Actividades	100%	18	140	25,20	33%	6	130	7,80	22%	4	120	4,80	22%	4	120	4,80
3	Division de tareas e informacion a investigar	100%	2	140	2,80	0%	0	130	0,00	0%	0	120	0,00	0%	0	120	0,00
4	Investigacion para propuesta	100%	24	140	33,60	33%	8	130	10,40	0%	0	120	0,00	0%	0	120	0,00
5	Confeccion de la propuesta	50%	24	140	33,60	50%	24	130	31,20	17%	8	120	9,60	0%	0	120	0,00
6	Entrega de Informe de Propuesta	-	-	140	-	-	-	130	-	-	-	120	-	-	-	-	-
7	Reunion N°2	100%	1	140	140	100%	1	130	130	100%	1	120	120	100%	1	120	120
8	Instalacion de Ubuntu 14.04	100%	4	140	5,60	0%	0	130	0,00	0%	0	120	0,00	100%	4	120	4,80
9	Instalacion y configuracion del Software necesario	100%	4	140	5,60	0%	0	130	0,00	0%	0	120	0,00	100%	4	120	4,80
10	Construccion del cluster	0%	0	140	0,00	0%	0	130	0,00	0%	0	120	0,00	100%	4	120	4,80
11	Ejecutar programa de prueba en cluster	0%	0	140	0,00	0%	0	130	0,00	0%	0	120	0,00	100%	4	120	4,80
12	Investigacion de diferentes algoritmos para la gestión de imagen	50%	8	140	11,20	0%	0	130	0,00	0%	0	120	0,00	25%	4	120	4,80
13	Reunión N°3	100%	1	140	140	100%	1	130	130	100%	1	120	120	100%	1	120	120
14	Modelar solucion de ambiente secuencial	100%	24	140	33,60	0%	0	130	0,00	67%	16	120	19,20	17%	4	120	4,80
15	Programacion solucion de ambiente secuencial	13%	1	140	140	0%	0	130	0,00	25%	2	120	2,40	100%	8	120	9,60
16	Optimizar solucion de ambiente secuencial	50%	4	140	5,60	0%	0	130	0,00	25%	2	120	2,40	50%	4	120	4,80
17	Calcular metricas de solucion de ambiente secuencial	0%	0	140	0,00	25%	1	130	130	25%	1	120	120	0%	0	120	0,00
18	Documentacion y analisis de QA de ambiente secuencial	25%	1	140	140	25%	1	130	130	50%	2	120	2,40	0%	0	120	0,00
19	Confeccion Informe Avance N°1	100%	16	140	22,40	63%	10	130	13,00	31%	5	120	6,00	0%	0	120	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	140	140	100%	1	130	130	100%	1	120	120	100%	1	120	120
22	Modelar solucion de ambiente paralelo	75%	15	140	21,00	0%	0	130	0,00	25%	5	120	6,00	100%	20	120	24,00
23	Programacion solucion de ambiente paralelo	8%	1	140	140	0%	0	130	0,00	17%	2	120	2,40	100%	12	120	14,40
24	Optimizar solucion de ambiente paralelo	25%	4	140	5,60	0%	0	130	0,00	31%	5	120	6,00	100%	16	120	19,20
25	Confeccion Informe Avance N°2	100%	24	140	33,60	83%	20	130	26,00	42%	10	120	12,00	0%	0	120	0,00
26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	140	140	100%	1	130	130	100%	1	120	120	100%	1	120	120
28	Calcular metricas de solucion de ambiente paralelo	14%	1	140	140	14%	1	130	130	14%	1	120	120	29%	2	120	2,40
29	Documentacion y analisis de QA de ambiente paralelo	50%	2	140	2,80	50%	2	130	2,60	25%	1	120	120	50%	2	120	2,40
30	Comparar métricas algoritmo secuencial versus paralelo	100%	4	140	5,60	100%	4	130	5,20	100%	4	120	4,80	50%	2	120	2,40
31	Modelar plataforma web	50%	8	140	11,20	0%	0	130	0,00	100%	16	120	19,20	25%	4	120	4,80
32	Integración de la aplicación paralela a la plataforma web	25%	4	140	5,60	0%	0	130	0,00	50%	8	120	9,60	50%	8	120	9,60
33	Reunion N°6	100%	1	140	140	100%	1	130	130	100%	1	120	120	100%	1	120	120
34	Programación interna de plataforma web	13%	2	140	2,80	0%	0	130	0,00	13%	2	120	2,40	13%	2	120	2,40
35	Modelar Interfaz gráfica de plataforma web	25%	4	140	5,60	0%	0	130	0,00	25%	4	120	4,80	100%	16	120	19,20
36	Reunion N°7	100%	1	140	140	100%	1	130	130	100%	1	120	120	100%	1	120	120
37	Programación de la interfaz gráfica de plataforma web	14%	1	140	140	0%	0	130	0,00	0%	0	120	0,00	0%	0	120	0,00
38	Montar el servidor para la plataforma	20%	1	140	140	0%	0	130	0,00	20%	1	120	120	20%	1	120	120
39	Reunion N°8	100%	1	140	140	100%	1	130	130	100%	1	120	120	100%	1	120	120
40	Confeccion Informe Final de Laboratorio Fase II	100%	24	140	33,60	83%	20	130	26,00	21%	5	120	6,00	0%	0	120	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
TOTAL			233,00		326,20		105,00		136,50		112,00		134,40		####		159,60

ID	Actividad	Grupo UNO															
		Elias Cáceres				Gabriel Sanhueza				Katherine Sáez				Mario Urbina			
		% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,70	0,70	100%	1	0,50	0,50
2	Planificación de Actividades	0%	0	1,00	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00
3	División de tareas e información a investigar	50%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00
4	Investigación para propuesta	33%	8	1,00	8,00	33%	8	0,50	4,00	33%	8	0,70	5,60	33%	8	0,50	4,00
5	Confección de la propuesta	2%	1	1,00	1,00	2%	1	0,50	0,50	2%	1	0,70	0,70	2%	1	0,50	0,50
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,70	0,70	100%	1	0,50	0,50
8	Instalación de Ubuntu 14.04	0%	0	1,00	0,00	100%	4	0,50	2,00	0%	0	0,70	0,00	100%	4	0,50	2,00
9	Instalación y configuración del Software necesario	0%	0	1,00	0,00	100%	4	0,50	2,00	0%	0	0,70	0,00	100%	4	0,50	2,00
10	Construcción del cluster	0%	0	1,00	0,00	100%	4	0,50	2,00	0%	0	0,70	0,00	100%	4	0,50	2,00
11	Ejecutar programa de prueba en cluster	0%	0	1,00	0,00	100%	4	0,50	2,00	0%	0	0,70	0,00	100%	4	0,50	2,00
12	Investigación de diferentes algoritmos para la gestión de imagen	50%	8	1,00	8,00	50%	8	0,50	4,00	50%	8	0,70	5,60	50%	8	0,50	4,00
13	Reunión N°3	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,70	0,70	100%	1	0,50	0,50
14	Modelar solución de ambiente secuencial	17%	4	1,00	4,00	0%	0	0,50	0,00	33%	8	0,70	5,60	0%	0	0,50	0,00
15	Programación solución de ambiente secuencial	50%	4	1,00	4,00	100%	8	0,50	4,00	25%	2	0,70	1,40	100%	8	0,50	4,00
16	Optimizar solución de ambiente secuencial	50%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00
17	Calcular métricas de solución de ambiente secuencial	100%	4	1,00	4,00	0%	0	0,50	0,00	100%	4	0,70	2,80	0%	0	0,50	0,00
18	Documentación y análisis de QA de ambiente secuencial	100%	4	1,00	4,00	0%	0	0,50	0,00	100%	4	0,70	2,80	0%	0	0,50	0,00
19	Confección Informe Avance N°1	6%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,70	0,70	100%	1	0,50	0,50
22	Modelar solución de ambiente paralelo	20%	4	1,00	4,00	0%	0	0,50	0,00	40%	8	0,70	5,60	0%	0	0,50	0,00
23	Programación solución de ambiente paralelo	17%	2	1,00	2,00	100%	12	0,50	6,00	0%	0	0,70	0,00	100%	12	0,50	6,00
24	Optimizar solución de ambiente paralelo	25%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00
25	Confección Informe Avance N°2	4%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00
26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,70	0,70	100%	1	0,50	0,50
28	Calcular métricas de solución de ambiente paralelo	57%	4	1,00	4,00	0%	0	0,50	0,00	57%	4	0,70	2,80	0%	0	0,50	0,00
29	Documentación y análisis de QA de ambiente paralelo	100%	4	1,00	4,00	0%	0	0,50	0,00	100%	4	0,70	2,80	0%	0	0,50	0,00
30	Comparar métricas algoritmo secuencial versus paralelo	25%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00
31	Modelar plataforma web	25%	4	1,00	4,00	0%	0	0,50	0,00	50%	8	0,70	5,60	0%	0	0,50	0,00
32	Integración de la aplicación paralela a la plataforma web	25%	4	1,00	4,00	13%	2	0,50	1,00	13%	2	0,70	1,40	13%	2	0,50	1,00
33	Reunion N°6	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,70	0,70	100%	1	0,50	0,50
34	Programación interna de plataforma web	13%	2	1,00	2,00	100%	16	0,50	8,00	0%	0	0,70	0,00	100%	16	0,50	8,00
35	Modelar Interfaz gráfica de plataforma web	13%	2	1,00	2,00	0%	0	0,50	0,00	50%	8	0,70	5,60	0%	0	0,50	0,00
36	Reunion N°7	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,70	0,70	100%	1	0,50	0,50
37	Programación de la interfaz gráfica de plataforma web	29%	2	1,00	2,00	100%	7	0,50	3,50	0%	0	0,70	0,00	100%	7	0,50	3,50
38	Montar el servidor para la plataforma	0%	0	1,00	0,00	100%	5	0,50	2,50	0%	0	0,70	0,00	100%	5	0,50	2,50
39	Reunion N°8	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,70	0,70	100%	1	0,50	0,50
40	Confección Informe Final de Laboratorio Fase II	4%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
TOTAL			82,00		82,00		91,00		45,50		77,00		53,90		91,00		45,50



ID	Actividad	Grupo DOS															
		Nicolás Soto				Alvaro Cabedo				Dominique Huenteleto				Boris Vásquez			
		% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	1,00	1,00	100%	1	0,70	0,70	100%	1	0,50	0,50	100%	1	0,50	0,50
2	Planificación de Actividades	0%	0	1,00	0,00	0%	0	0,70	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00
3	División de tareas e información a investigar	50%	1	1,00	1,00	0%	0	0,70	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00
4	Investigación para propuesta	33%	8	1,00	8,00	33%	8	0,70	5,60	33%	8	0,50	4,00	33%	8	0,50	4,00
5	Confección de la propuesta	2%	1	1,00	1,00	2%	1	0,70	0,70	2%	1	0,50	0,50	2%	1	0,50	0,50
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	1,00	1,00	100%	1	0,70	0,70	100%	1	0,50	0,50	100%	1	0,50	0,50
8	Instalación de Ubuntu 14.04	0%	0	1,00	0,00	0%	0	0,70	0,00	100%	4	0,50	2,00	100%	4	0,50	2,00
9	Instalación y configuración del Software necesario	0%	0	1,00	0,00	0%	0	0,70	0,00	100%	4	0,50	2,00	100%	4	0,50	2,00
10	Construcción del cluster	0%	0	1,00	0,00	0%	0	0,70	0,00	100%	4	0,50	2,00	100%	4	0,50	2,00
11	Ejecutar programa de prueba en cluster	0%	0	1,00	0,00	0%	0	0,70	0,00	100%	4	0,50	2,00	100%	4	0,50	2,00
12	Investigación de diferentes algoritmos para la gestión de imagen	50%	8	1,00	8,00	50%	8	0,70	5,60	50%	8	0,50	4,00	50%	8	0,50	4,00
13	Reunión N°3	100%	1	1,00	1,00	100%	1	0,70	0,70	100%	1	0,50	0,50	100%	1	0,50	0,50
14	Modelar solución de ambiente secuencial	17%	4	1,00	4,00	33%	8	0,70	5,60	0%	0	0,50	0,00	0%	0	0,50	0,00
15	Programación solución de ambiente secuencial	50%	4	1,00	4,00	25%	2	0,70	1,40	100%	8	0,50	4,00	100%	8	0,50	4,00
16	Optimizar solución de ambiente secuencial	50%	4	1,00	4,00	0%	0	0,70	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00
17	Calcular métricas de solución de ambiente secuencial	100%	4	1,00	4,00	100%	4	0,70	2,80	0%	0	0,50	0,00	0%	0	0,50	0,00
18	Documentación y análisis de QA de ambiente secuencial	100%	4	1,00	4,00	100%	4	0,70	2,80	0%	0	0,50	0,00	0%	0	0,50	0,00
19	Confección Informe Avance N°1	6%	1	1,00	1,00	0%	0	0,70	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	1,00	1,00	100%	1	0,70	0,70	100%	1	0,50	0,50	100%	1	0,50	0,50
22	Modelar solución de ambiente paralelo	20%	4	1,00	4,00	40%	8	0,70	5,60	0%	0	0,50	0,00	0%	0	0,50	0,00
23	Programación solución de ambiente paralelo	17%	2	1,00	2,00	0%	0	0,70	0,00	100%	12	0,50	6,00	100%	12	0,50	6,00
24	Optimizar solución de ambiente paralelo	25%	4	1,00	4,00	0%	0	0,70	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00
25	Confección Informe Avance N°2	4%	1	1,00	1,00	0%	0	0,70	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00
26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	1,00	1,00	100%	1	0,70	0,70	100%	1	0,50	0,50	100%	1	0,50	0,50
28	Calcular métricas de solución de ambiente paralelo	57%	4	1,00	4,00	57%	4	0,70	2,80	0%	0	0,50	0,00	0%	0	0,50	0,00
29	Documentación y análisis de QA de ambiente paralelo	100%	4	1,00	4,00	100%	4	0,70	2,80	0%	0	0,50	0,00	0%	0	0,50	0,00
30	Comparar métricas algoritmo secuencial versus paralelo	25%	1	1,00	1,00	0%	0	0,70	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00
31	Modelar plataforma web	25%	4	1,00	4,00	50%	8	0,70	5,60	0%	0	0,50	0,00	0%	0	0,50	0,00
32	Integración de la aplicación paralela a la plataforma web	25%	4	1,00	4,00	13%	2	0,70	1,40	13%	2	0,50	1,00	13%	2	0,50	1,00
33	Reunion N°6	100%	1	1,00	1,00	100%	1	0,70	0,70	100%	1	0,50	0,50	100%	1	0,50	0,50
34	Programación interna de plataforma web	13%	2	1,00	2,00	0%	0	0,70	0,00	100%	16	0,50	8,00	100%	16	0,50	8,00
35	Modelar Interfaz gráfica de plataforma web	13%	2	1,00	2,00	50%	8	0,70	5,60	0%	0	0,50	0,00	0%	0	0,50	0,00
36	Reunion N°7	100%	1	1,00	1,00	100%	1	0,70	0,70	100%	1	0,50	0,50	100%	1	0,50	0,50
37	Programación de la interfaz gráfica de plataforma web	29%	2	1,00	2,00	0%	0	0,70	0,00	100%	7	0,50	3,50	100%	7	0,50	3,50
38	Montar el servidor para la plataforma	0%	0	1,00	0,00	0%	0	0,70	0,00	100%	5	0,50	2,50	100%	5	0,50	2,50
39	Reunion N°8	100%	1	1,00	1,00	100%	1	0,70	0,70	100%	1	0,50	0,50	100%	1	0,50	0,50
40	Confección Informe Final de Laboratorio Fase II	4%	1	1,00	1,00	0%	0	0,70	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
TOTAL			82,00		82,00		77,00		53,90		91,00		45,50		91,00		45,50

		Grupo TRES															
ID	Actividad	Gladys Cerro				Felipe Abarca				David Martínez				Jonathan Acuña			
		% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)	% Dedic.	HH Esf.	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,50	0,50	100%	1	0,70	0,70
2	Planificacion de Actividades	0%	0	1,00	0,00	0%	0	0,50	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00
3	Division de tareas e informacion a investigar	50%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00
4	Investigacion para propuesta	33%	8	1,00	8,00	33%	8	0,50	4,00	33%	8	0,50	4,00	33%	8	0,70	5,60
5	Confeccion de la propuesta	2%	1	1,00	1,00	2%	1	0,50	0,50	2%	1	0,50	0,50	2%	1	0,70	0,70
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,50	0,50	100%	1	0,70	0,70
8	Instalacion de Ubuntu 14.04	0%	0	1,00	0,00	100%	4	0,50	2,00	100%	4	0,50	2,00	0%	0	0,70	0,00
9	Instalacion y configuracion del Software necesario	0%	0	1,00	0,00	100%	4	0,50	2,00	100%	4	0,50	2,00	0%	0	0,70	0,00
10	Construccion del cluster	0%	0	1,00	0,00	100%	4	0,50	2,00	100%	4	0,50	2,00	0%	0	0,70	0,00
11	Ejecutar programa de prueba en cluster	0%	0	1,00	0,00	100%	4	0,50	2,00	100%	4	0,50	2,00	0%	0	0,70	0,00
12	Investigacion de diferentes algoritmos para la gestión de imagen	50%	8	1,00	8,00	50%	8	0,50	4,00	50%	8	0,50	4,00	50%	8	0,70	5,60
13	Reunión N°3	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,50	0,50	100%	1	0,70	0,70
14	Modelar solucion de ambiente secuencial	17%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	33%	8	0,70	5,60
15	Programacion solucion de ambiente secuencial	50%	4	1,00	4,00	100%	8	0,50	4,00	100%	8	0,50	4,00	25%	2	0,70	1,40
16	Optimizar solucion de ambiente secuencial	50%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00
17	Calcular metricas de solucion de ambiente secuencial	100%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	100%	4	0,70	2,80
18	Documentacion y analisis de QA de ambiente secuencial	100%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	100%	4	0,70	2,80
19	Confeccion Informe Avance N°1	6%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,50	0,50	100%	1	0,70	0,70
22	Modelar solucion de ambiente paralelo	20%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	40%	8	0,70	5,60
23	Programacion solucion de ambiente paralelo	17%	2	1,00	2,00	100%	12	0,50	6,00	100%	12	0,50	6,00	0%	0	0,70	0,00
24	Optimizar solucion de ambiente paralelo	25%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00
25	Confeccion Informe Avance N°2	4%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00
26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,50	0,50	100%	1	0,70	0,70
28	Calcular metricas de solucion de ambiente paralelo	57%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	57%	4	0,70	2,80
29	Documentacion y analisis de QA de ambiente paralelo	100%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	100%	4	0,70	2,80
30	Comparar métricas algoritmo secuencial versus paralelo	25%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00
31	Modelar plataforma web	25%	4	1,00	4,00	0%	0	0,50	0,00	0%	0	0,50	0,00	50%	8	0,70	5,60
32	Integración de la aplicación paralela a la plataforma web	25%	4	1,00	4,00	13%	2	0,50	1,00	13%	2	0,50	1,00	13%	2	0,70	1,40
33	Reunion N°6	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,50	0,50	100%	1	0,70	0,70
34	Programación interna de plataforma web	13%	2	1,00	2,00	100%	16	0,50	8,00	100%	16	0,50	8,00	0%	0	0,70	0,00
35	Modelar Interfaz gráfica de plataforma web	13%	2	1,00	2,00	0%	0	0,50	0,00	0%	0	0,50	0,00	50%	8	0,70	5,60
36	Reunion N°7	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,50	0,50	100%	1	0,70	0,70
37	Programación de la interfaz gráfica de plataforma web	29%	2	1,00	2,00	100%	7	0,50	3,50	100%	7	0,50	3,50	0%	0	0,70	0,00
38	Montar el servidor para la plataforma	0%	0	1,00	0,00	100%	5	0,50	2,50	100%	5	0,50	2,50	0%	0	0,70	0,00
39	Reunion N°8	100%	1	1,00	1,00	100%	1	0,50	0,50	100%	1	0,50	0,50	100%	1	0,70	0,70
40	Confeccion Informe Final de Laboratorio Fase II	4%	1	1,00	1,00	0%	0	0,50	0,00	0%	0	0,50	0,00	0%	0	0,70	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
TOTAL			82,00		82,00		91,00		45,50		91,00		45,50		77,00		53,90

Equipo de Trabajo	Esfuerzo (HH)	UF*Hora	Costo Total
Felipe Canales	237	1,40	331,80
Camila Vera	105	1,30	136,50
Nelson Cariqueo	112	1,20	134,40
Diego Garcia	137	1,20	164,40
Elias Caceres	82	1,00	82,00
Nicolas Soto	82	1,00	82,00
Gladys Cerro	82	1,00	82,00
Katherine Saez	77	0,70	53,90
Alvaro Cabedo	77	0,70	53,90
Jonathan Acuña	77	0,70	53,90
Gabriel Sanhueza	91	0,50	45,50
Mario Urbina	91	0,50	45,50
Dominique Huenteleo	91	0,50	45,50
Boris Vasquez	91	0,50	45,50
Felipe Abarca	91	0,50	45,50
David Martinez	91	0,50	45,50
<b>Presupuesto Proyecto (Horas)</b>	<b>1614</b>	<b>-</b>	<b>-</b>
<b>Costo Capital Humano (UF)</b>	<b>-</b>	<b>-</b>	<b>1447,80</b>

## 9. Planificación

### 9.1. Control de avances

Para el desarrollo del laboratorio fase II, se implementará el uso de un tablero Kanban con el objetivo de controlar los avances realizados, tanto por los grupos de trabajo como los personales. De esta forma es posible tener en forma clara los puntos débiles y aplicar las medidas correctivas necesarias para corregirlos, también permite revisar si una actividad es necesario reprogramarla, etc.

El tablero estará construido de la siguiente manera:

Tabla 7, Tablero Kanban para la distribución de tareas

Grupo	Integrantes	Tarea(s) asignada(s)	Tarea(s) en desarrollo	Pruebas	Tarea(s) Finalizada(s)
Grupo 1	Elías Cáceres				
	Gabriel Sanhueza				
	Katherine Sáez				
	Mario Urbina				
Grupo 2	Nicolás Soto				
	Alvaro Cabedo				
	Dominique Huetaleo				
	Boris Vásquez				
Grupo 3	Gladys Cerro				
	Felipe Abarca				
	David Martínez				
	Jonathan Acuña				

Fuente: Elaboración propia

Sumado a esto, el formato de cada requerimiento, o tarea, deberá contener (a lo menos), los siguientes tres puntos:

- Fecha de creación.
- Prioridad, en escala de 1 a 5.
- Breve descripción.

## 9.2. Control de valor ganado

Para tener un control sobre el proyecto en todo momento, se realizará un seguimiento, en los cumplimientos de las tareas asignadas a cada grupo y, por ende, a cada miembro (que se encuentre trabajando actualmente en el proyecto) en cada etapa de control. Siendo estas etapas las fechas a las que correspondan las entregas de avances.

Es por esto, que es de vital importancia, que la persona que esté designada como Jefe de grupo tiene el deber y obligación de mostrar los indicadores válidos para demostrar, fehacientemente, que sus actividades no se encuentran con retraso alguno.

Con esta metodología es posible otorgar herramientas reales al área de control, para poder generar presión sobre los empleados y obtener resultados deseados.

## 10. Marco de evaluación

A continuación, se detallarán las medidas de control que se propondrán para evaluar el rendimiento, comportamiento y aspectos personales de cada integrante de los grupos. De esta manera se podrán obtener notas, las cuales serán promediadas consiguiendo un promedio final asociado a cada integrante.

En adición, cada evaluación (ya sea parcial o final) tendrá un factor numérico de importancia. De esta manera se evitarán casos como: El faltar a reuniones, pero lograr conseguir los avances de las tareas definidas.

### 10.1. Criterios de permanencia y amonestaciones

Como en todo ámbito laboral, es necesario definir criterios de permanencia para poder permanecer en el equipo de trabajo. A su vez, estos deben ser transparentes y de fácil entendimiento para que no existan problemas por parte de los trabajadores y no se produzcan dobles sentidos a la hora de transgredirlos.

A continuación, se definen 5 criterios que deben ser cumplidos para poder permanecer en el equipo de trabajo:

- Cumplimiento con las fechas de los plazos establecidos, si esto no es posible, es deber y obligación del jefe de grupo avisar con pertinencia y elevar una solicitud para re-agendar la actividad o asignación necesaria.  
Adjuntando todos los datos pertinentes que demuestren que es necesario realizar la reprogramación (ej. Tabla y gráficos de valor ganado de su grupo, tabla con porcentaje de avance, etc.).
- Es deber de la persona que esté a cargo de una tarea, mostrar el avance que se le solicite (en cualquier momento) y tener un claro conocimiento del mismo.
- El hecho de propinar reiteradas ofensas, entre y hacia, los integrantes de los grupos de trabajo.
- El hecho de no respetar la cadena de mando establecida.

Para que la solicitud de aplazo sea aprobada, esta debe tener la firma de los cuatro integrantes del grupo líder. Manteniendo una copia el grupo líder y otra el Jefe de grupo que eleva esta solicitud.

En caso de no cumplir los criterios de permanencia, al afectado (ya sea integrante del grupo o Jefe de grupo) se le otorgará una carta de amonestación detallando las causas del porqué tuvo que recibir dicho documento (conservando una copia firmada el afectado y otra el jefe de proyecto). Si la persona se niega a firmar, se expondrá su rebeldía al profesor.

Si se da la situación de que una persona acumule tres cartas de este tipo, se procederá a la expulsión inmediata del “grupo-curso”.

## 11. Desarrollo del proyecto

### 11.1. Herramientas utilizadas

- Sistema Operativo: Ubuntu 16.04 LTS, es un sistema operativo basado en GNU/LINUX y que distribuye como software libre, el cual incluye su propio entorno de escritorio denominado Unity.
- Lenguaje de programación: Python fue utilizado para configurar el Clúster, la elección de este es porque hace hincapié en una sintaxis que favorezca un código legible.
- Middleware: MPI es un estándar que define la sintaxis y la semántica de las funciones contenidas en una biblioteca de paso de mensajes diseñada para ser usada en programas que exploten la existencia de múltiples procesadores.
- IDE: Pycharm Community Edition, define la sintaxis y la semántica de las funciones contenidas en una de paso de mensajes diseñada para ser usada en programas que exploten la existencia de múltiples procesadores.
- Monitor de procesadores: Htop, programa de Ubuntu para monitorear la memoria.

## 11.2. Tiempos de Ejecución, Speed Up, Eficiencia y Costo

### 11.2.1. Grupo UNO

Tabla 8, Medición de tiempo de Efecto LoG

Efecto LoG					
Imagen	Tiempos (segundos)		Medidas de Rendimiento		
	secuencial	Paralelo	Speed Up	Eficiencia	Costo
píxeles	secuencial	N=4	N=4	N=4	N=4
900 x 600	0,483	0,184	2,625	0,656	10,500
1152 x 720	0,651	0,287	2,268	0,567	9,073
1280 x 800	0,781	0,368	2,122	0,531	8,489
1440 x 900	1,001	0,453	2,210	0,552	8,839
1920 x 1200	1,805	0,812	2,223	0,556	8,892

Fuente: Elaboración propia

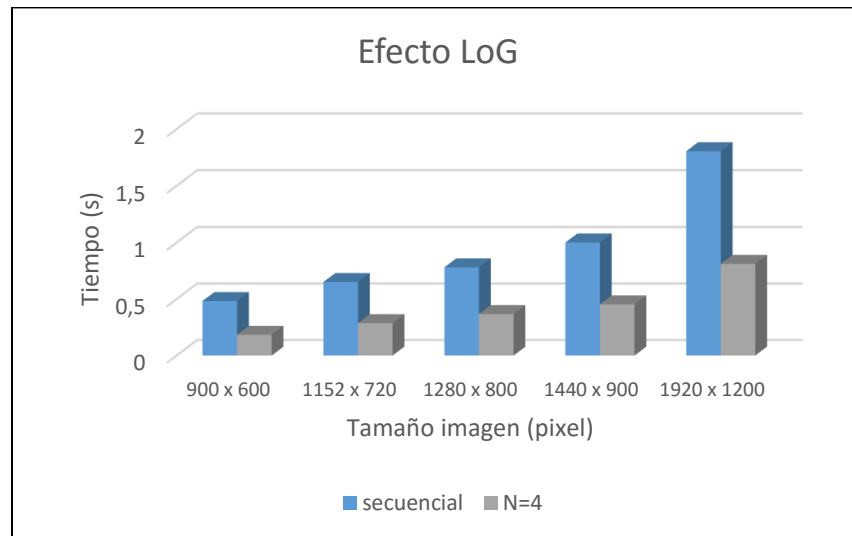


Tabla 9, Medición de tiempo de Dispersión Gaussiana

Dispersión Gaussiana					
Imagen	Tiempos (segundos)		Medidas de Rendimiento		
	secuencial	Paralelo	Speed Up	Eficiencia	Costo
píxeles	secuencial	N=4	N=4	N=4	N=4
900 x 600	0,412	0,175	2,354	0,589	9,417
1152 x 720	0,553	0,259	2,135	0,534	8,541
1280 x 800	0,722	0,321	2,249	0,562	8,997
1440 x 900	0,864	0,411	2,102	0,526	8,409
1920 x 1200	1,529	0,695	2,200	0,550	8,800

Fuente: Elaboración propia



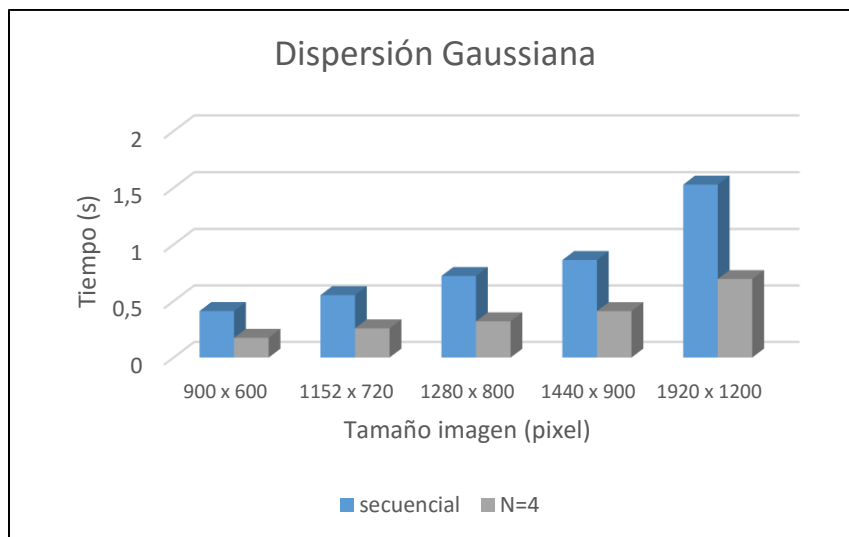


Tabla 10, Medición de tiempo de, Efecto RGB

Efecto RGB					
Imagen	Tiempos (segundos)		Medidas de Rendimiento		
	secuencial	Paralelo	Speed Up	Eficiencia	Costo
pixeles	secuencial	N=4	N=4	N=4	N=4
900 x 600	10,995	7,523	1,462	0,365	5,846
1152 x 720	16,813	10,355	1,624	0,406	6,495
1280 x 800	19,535	14,491	1,348	0,337	5,392
1440 x 900	25,208	18,273	1,380	0,345	5,518
1920 x 1200	43,427	31,584	1,375	0,344	5,500

Fuente: Elaboración propia

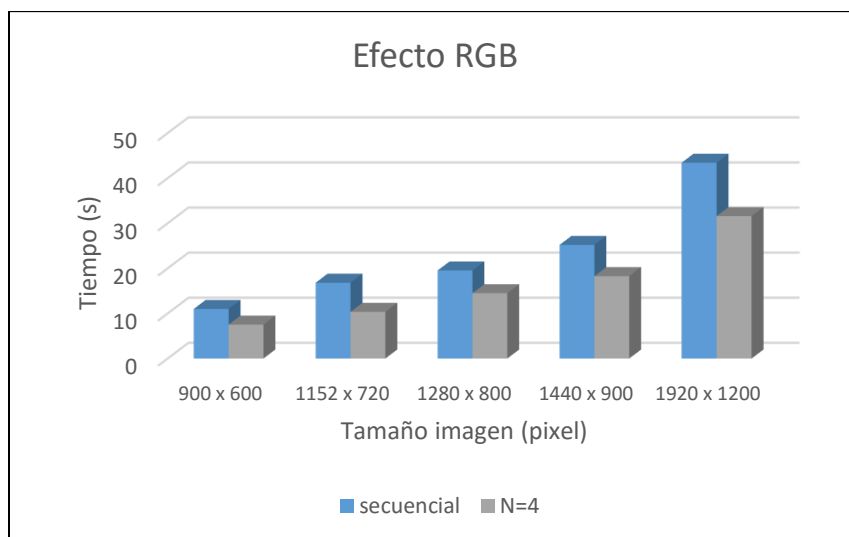


Tabla 11, Medición de tiempo de Efecto Polar

Efecto Polar					
Imagen	Tiempos (segundos)		Medidas de Rendimiento		
	secuencial	Paralelo	Speed Up	Eficiencia	Costo
pixeles	secuencial	N=4	N=4	N=4	N=4
900 x 600	13,645	En el algoritmo polar, debido a que la imagen es compacta y utiliza coordenadas polares, donde cada columna de la imagen depende de la otra, solo se pudo implementar el código de forma secuencial.			
1152 x 720	18,903				
1280 x 800	23,689				
1440 x 900	33,171				
1920 x 1200	53,377				

Fuente: Elaboración propia

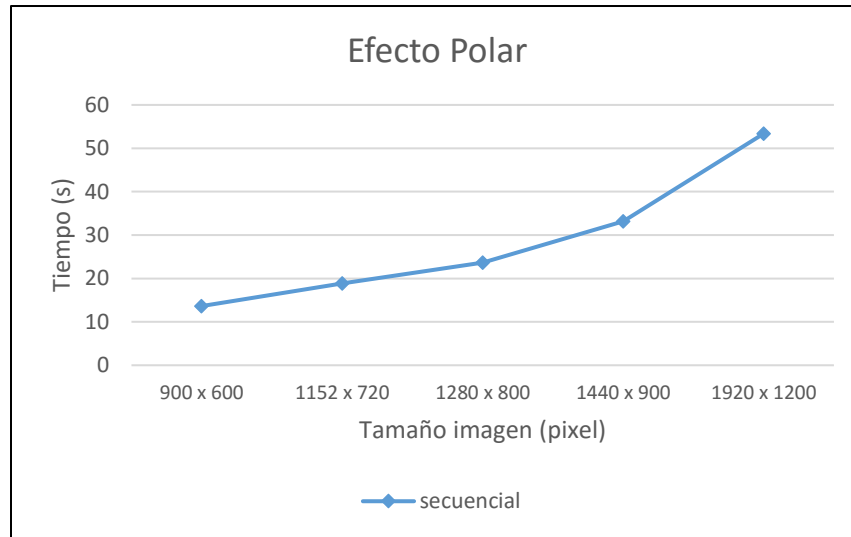
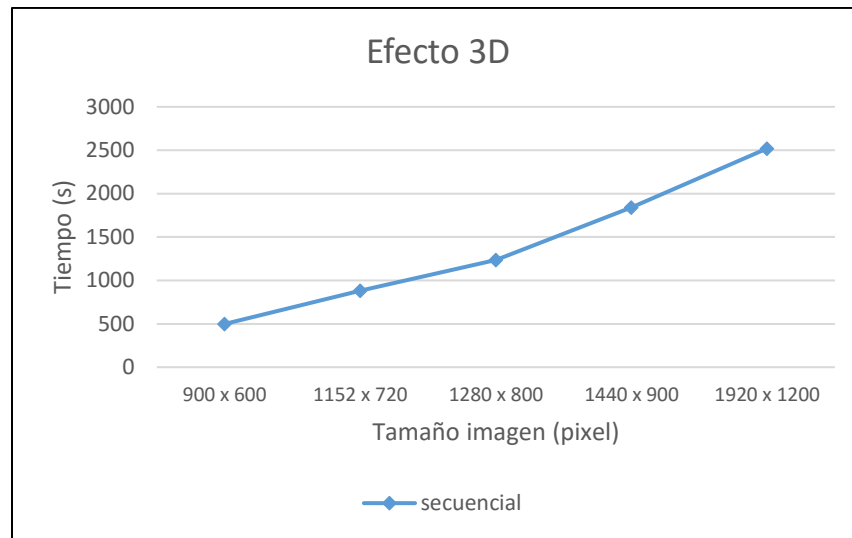


Tabla 12, Medición de tiempo de Efecto 3D

Efecto 3D					
Imagen	Tiempos (segundos)		Medidas de Rendimiento		
	secuencial	Paralelo	Speed Up	Eficiencia	Costo
pixeles	secuencial	N=4	N=4	N=4	N=4
900 x 600	498	Una de las razones por las que no se pudo paralelizar el algoritmo, fue que no se encontró la forma de dividir el problema en subproblemas para poder ser resuelto por el algoritmo de forma paralela, además de no encontrarse todo el algoritmo en lenguaje Python en su totalidad.			
1152 x 720	881				
1280 x 800	1234				
1440 x 900	1841				
1920 x 1200	2520				

Fuente: Elaboración propia



## 11.2.2. Grupo DOS

Tabla 13, Medición de tiempo de Efecto Espejo Horizontal

Efecto Espejo Horizontal													
Imagen	Tiempos (segundos)				Medidas de Rendimiento								
	secuencial	paralelos			Speed Up			Eficiencia			Costo		
pixeles	secuencial	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8
900 x 600	0,337	0,012	0,058	0,423	28,938	5,846	0,797	14,469	1,462	0,100	57,876	23,386	6,374
1152 x 720	0,506	0,016	0,069	0,410	31,089	7,375	1,235	15,545	1,844	0,154	62,179	29,499	9,882
1280 x 800	0,622	0,020	0,076	0,521	31,201	8,173	1,193	15,600	2,043	0,149	62,401	32,692	9,548
1440 x 900	0,787	0,024	0,106	0,494	32,367	7,442	1,592	16,183	1,861	0,199	64,733	29,770	12,735
1920 x 1200	1,383	0,041	0,141	0,458	33,624	9,837	3,021	16,812	2,459	0,378	67,248	39,349	24,166

Fuente: Elaboración propia

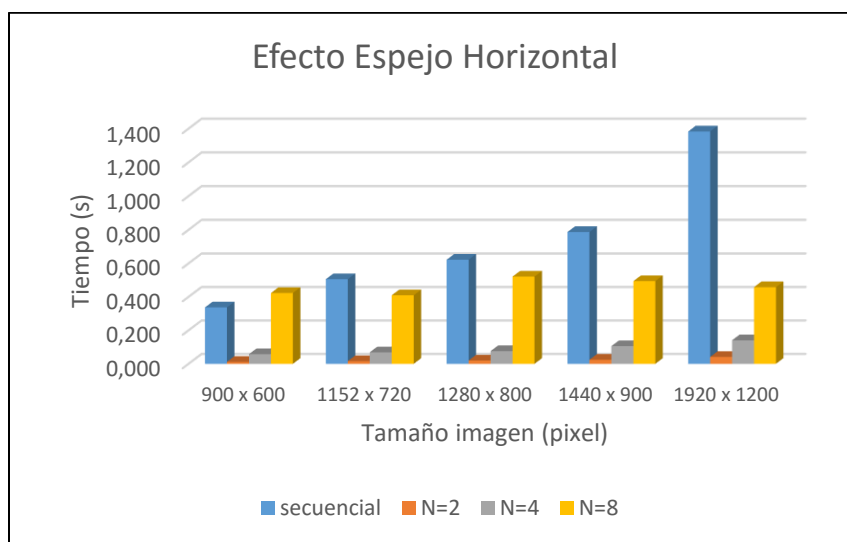


Tabla 14, Medición de tiempos de Espejo Vertical

Efecto Espejo Vertical													
Imagen	Tiempos (segundos)				Medidas de Rendimiento								
	secuencial	paralelos			Speed Up			Eficiencia			Costo		
pixeles	secuencial	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8
900 x 600	0,339	0,015	0,049	0,536	22,540	6,939	0,632	11,270	1,735	0,079	45,081	27,757	5,058
1152 x 720	0,499	0,015	0,076	0,397	33,971	6,565	1,257	16,986	1,641	0,157	67,943	26,261	10,054
1280 x 800	0,622	0,018	0,083	0,207	34,475	7,466	3,002	17,237	1,866	0,375	68,950	29,863	24,013
1440 x 900	0,752	0,022	0,087	0,340	33,742	8,643	2,211	16,871	2,161	0,276	67,484	34,571	17,686
1920 x 1200	1,432	0,035	0,146	0,490	41,143	9,822	2,924	20,572	2,456	0,365	82,286	39,288	23,390

Fuente: Elaboración propia

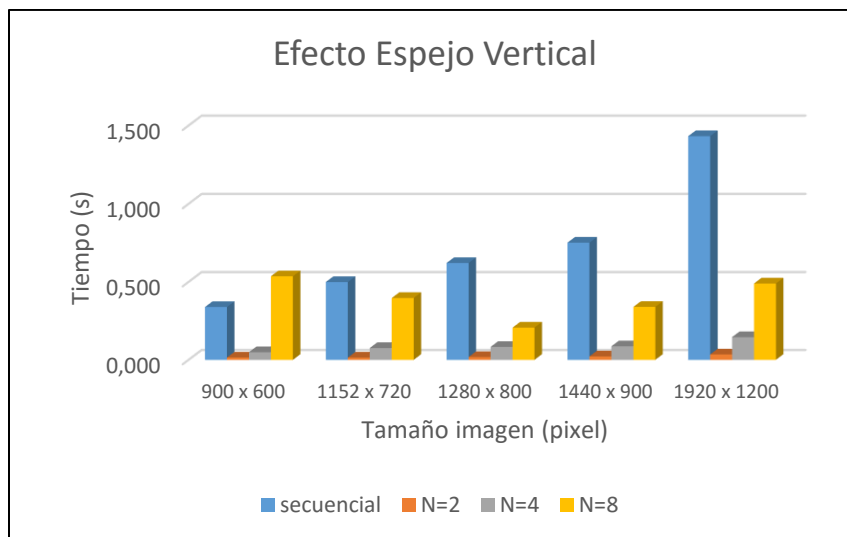


Tabla 15, Medición de tiempo de Efecto Luminosidad

Efecto Luminosidad													
Imagen	Tiempos (segundos)				Medidas de Rendimiento								
	secuencial	paralelos			Speed Up			Eficiencia			Costo		
pixeles	secuencial	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8
900 x 600	0,184	0,048	0,121	0,798	3,835	1,516	0,230	1,917	0,379	0,029	7,670	6,064	1,840
1152 x 720	0,141	0,075	0,169	1,395	1,868	0,835	0,101	0,934	0,209	0,013	3,735	3,341	0,808
1280 x 800	0,150	0,094	0,193	0,666	1,598	0,775	0,225	0,799	0,194	0,028	3,196	3,101	1,800
1440 x 900	0,160	0,118	0,235	1,028	1,358	0,680	0,155	0,679	0,170	0,019	2,715	2,720	1,242
1920 x 1200	0,209	0,153	0,398	1,451	1,367	0,525	0,144	0,683	0,131	0,018	2,734	2,099	1,152

Fuente: Elaboración propia

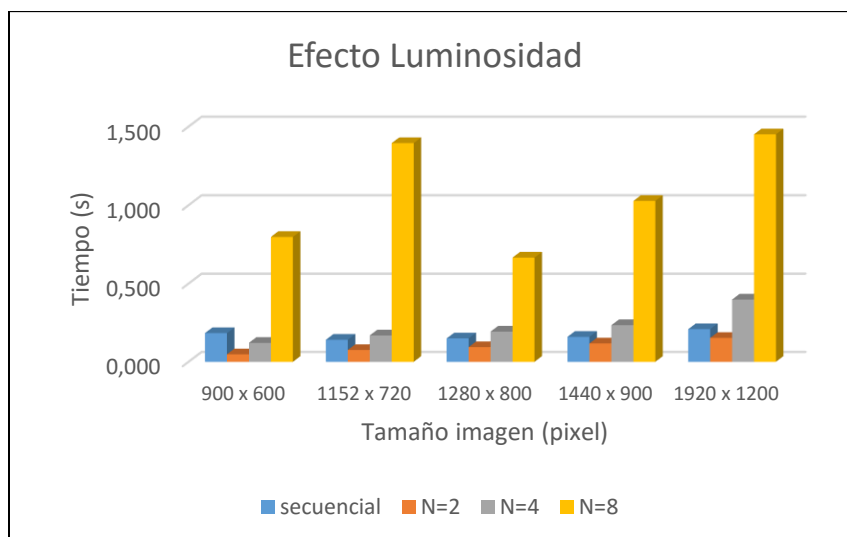
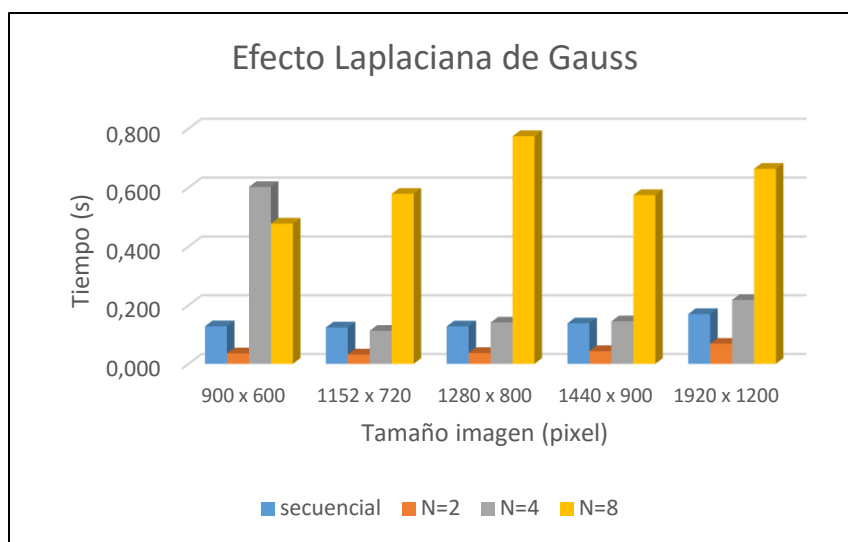


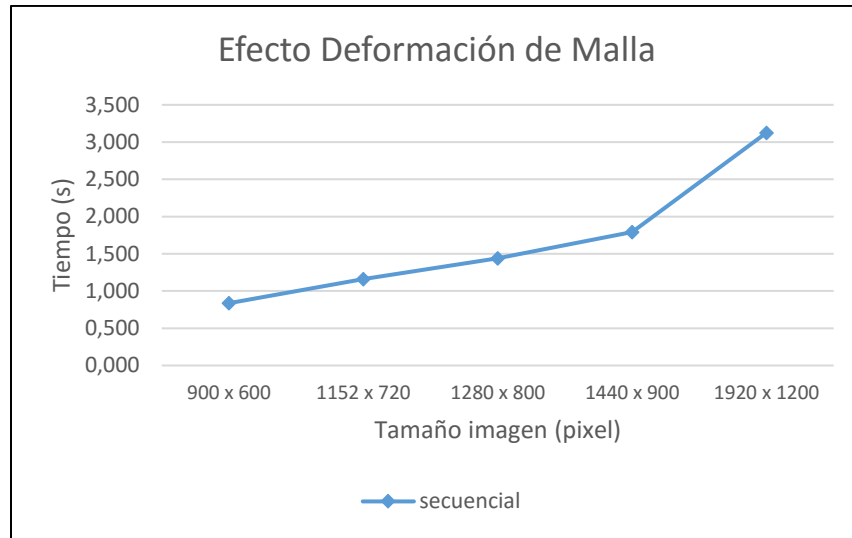
Tabla 16, Medición de tiempo de Laplaciana de Gauss

Efecto Laplaciana de Gauss													
Imagen	Tiempos (segundos)				Medidas de Rendimiento								
	secuencial	paralelos			Speed Up			Eficiencia			Costo		
pixeles	secuencial	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8
900 x 600	0,128	0,036	0,602	0,478	3,583	0,213	0,268	1,792	0,053	0,033	7,167	0,850	2,143
1152 x 720	0,124	0,032	0,113	0,579	3,890	1,100	0,214	1,945	0,275	0,027	7,780	4,399	1,713
1280 x 800	0,128	0,037	0,141	0,774	3,460	0,906	0,165	1,730	0,226	0,021	6,920	3,622	1,318
1440 x 900	0,138	0,043	0,145	0,575	3,183	0,952	0,240	1,591	0,238	0,030	6,366	3,810	1,923
1920 x 1200	0,170	0,069	0,218	0,664	2,477	0,780	0,256	1,238	0,195	0,032	4,954	3,120	2,046

Fuente: Elaboración propia



Efecto Deformación de Malla													
Imagen	Tiempos (segundos)				Medidas de Rendimiento								
	secuencial	paralelos			Speed Up			Eficiencia			Costo		
pixeles	secuencial	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8	N=2	N=4	N=8
900 x 600	0,837												
1152 x 720	1,162												
1280 x 800	1,440												
1440 x 900	1,793												
1920 x 1200	3,125												



### 11.2.3. Grupo TRES

Tabla 17, Medición de tiempo de Suma de fotos

Suma de fotos									
Imagen	Tiempos (segundos)			Medidas de Rendimiento					
	secuencial	paralelos		Speed Up		Eficiencia		Costo	
pixeles	secuencial	N=2	N=4	N=2	N=4	N=2	N=4	N=2	N=4
900 x 600	5,345	0,094	0,113	56,862	47,301	14,215	23,650	227,447	94,602
1152 x 720	7,342	0,126	0,144	58,270	50,986	14,567	25,493	233,079	101,972
1280 x 800	9,276	0,146	0,182	63,534	50,967	15,884	25,484	254,137	101,934
1440 x 900	11,367	0,181	0,202	62,801	56,272	15,700	28,136	251,204	112,545
1920 x 1200	20,553	0,284	0,324	72,370	63,435	18,092	31,718	289,479	126,870

Fuente: Elaboración propia

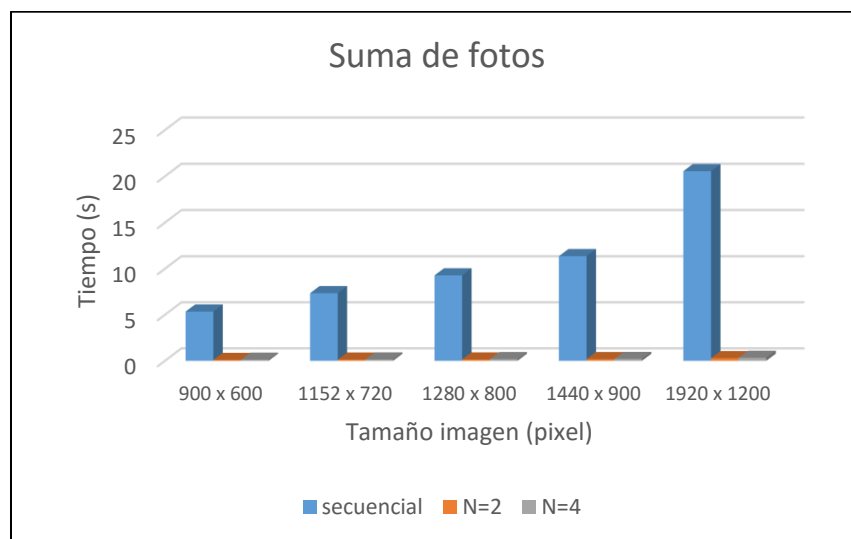


Tabla 18, Medición de tiempo de Efecto posicionar los bordes

Efecto posicionar los bordes									
Imagen	Tiempos (segundos)			Medidas de Rendimiento					
	secuencial	paralelos		Speed Up		Eficiencia		Costo	
pixeles	secuencial	N=2	N=4	N=2	N=4	N=2	N=4	N=2	N=4
900 x 600	5,012	4,641	3,085	1,080	1,625	0,270	0,812	4,320	3,249
1152 x 720	7,631	6,611	4,183	1,154	1,824	0,289	0,912	4,617	3,649
1280 x 800	8,574	8,327	5,078	1,030	1,688	0,257	0,844	4,119	3,377
1440 x 900	10,786	10,551	6,431	1,022	1,677	0,256	0,839	4,089	3,354
1920 x 1200	19,101	18,621	11,383	1,026	1,678	0,256	0,839	4,103	3,356

Fuente: Elaboración propia



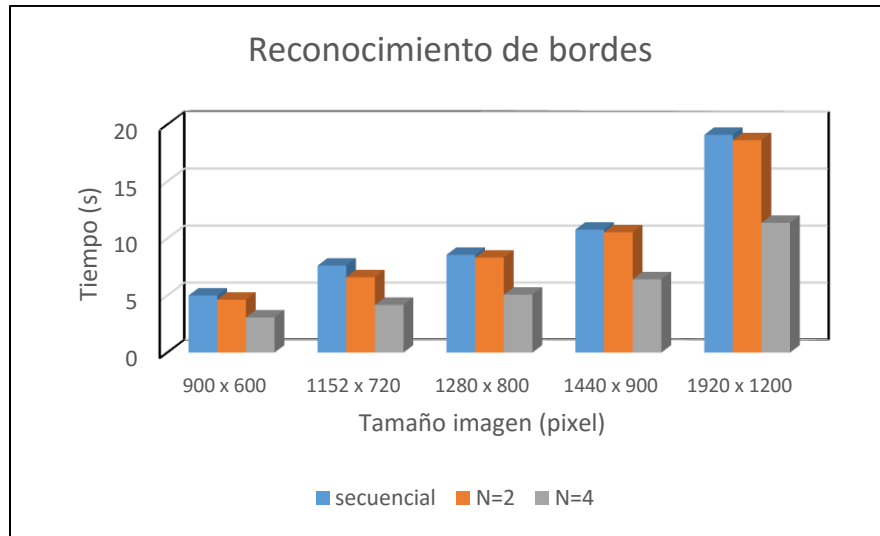


Tabla 19, Medición de tiempo de Transformada rápida de Fourier

Transformada rápida de Fourier									
Imagen	Tiempos (segundos)			Medidas de Rendimiento					
	secuencial	paralelos		Speed Up		Eficiencia		Costo	
pixeles	secuencial	N=2	N=4	N=2	N=4	N=2	N=4	N=2	N=4
900 x 600	2,621	2,593	2,293	1,011	1,143	0,253	0,572	4,043	2,286
1152 x 720	2,953	2,735	2,537	1,080	1,164	0,270	0,582	4,319	2,328
1280 x 800	3,22	3,192	2,771	1,009	1,162	0,252	0,581	4,035	2,324
1440 x 900	3,644	3,893	3,215	0,936	1,133	0,234	0,567	3,744	2,267
1920 x 1200	3,728	5,591	4,511	0,667	0,826	0,167	0,413	2,667	1,653

Fuente: Elaboración propia

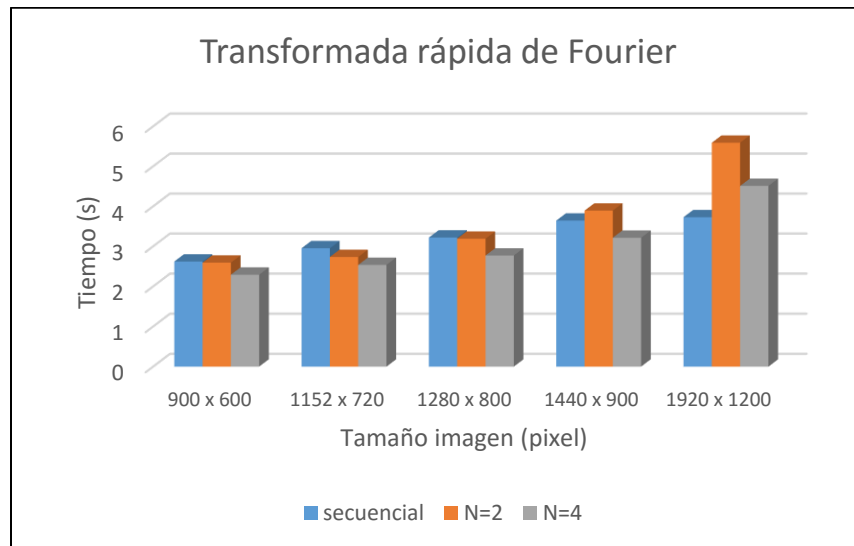


Tabla 20, Medición de tiempo de interpolación Bicubic

Interpolación Bicubic									
Imagen	Tiempos (segundos)			Medidas de Rendimiento					
	secuencial	paralelos		Speed Up		Eficiencia		Costo	
pixeles	secuencial	N=2	N=4	N=2	N=4	N=2	N=4	N=2	N=4
900 x 600	3,741	0,801	1,884	4,670	1,986	1,168	0,993	18,682	3,971
1152 x 720	5,862	0,826	1,929	7,097	3,039	1,774	1,519	28,387	6,078
1280 x 800	7,224	0,833	1,935	8,672	3,733	2,168	1,867	34,689	7,467
1440 x 900	9,043	0,863	1,942	10,479	4,657	2,620	2,328	41,914	9,313
1920 x 1200	16,526	0,908	1,947	18,200	8,488	4,550	4,244	72,802	16,976

Fuente: Elaboración propia

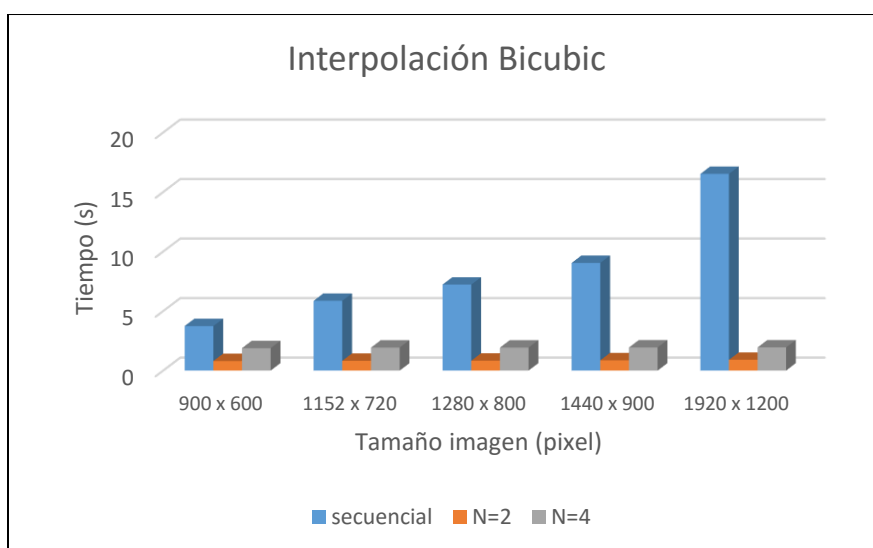


Tabla 21, Medición de tiempo de Interpolación Bilineal

Interpolación Bilineal									
Imagen	Tiempos (segundos)			Medidas de Rendimiento					
	secuencial	paralelos		Speed Up		Eficiencia		Costo	
pixeles	secuencial	N=2	N=4	N=2	N=4	N=2	N=4	N=2	N=4
900 x 600	0,205	0,807	2,237	0,254	0,092	0,064	0,046	1,016	0,183
1152 x 720	0,289	0,837	2,305	0,345	0,125	0,086	0,063	1,381	0,251
1280 x 800	0,348	0,841	2,315	0,414	0,150	0,103	0,075	1,655	0,301
1440 x 900	0,441	0,862	2,373	0,512	0,186	0,128	0,093	2,046	0,372
1920 x 1200	0,759	0,905	2,401	0,839	0,316	0,210	0,158	3,355	0,632

Fuente: Elaboración propia

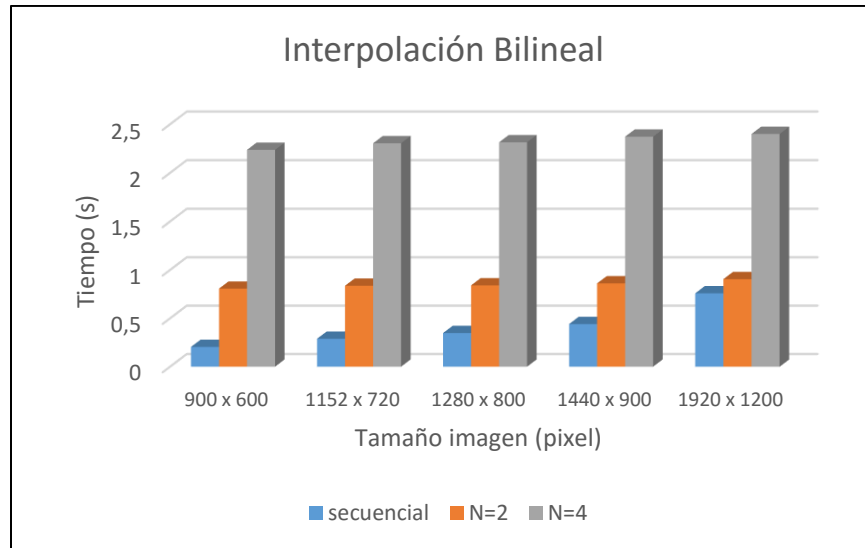


Tabla 22, Medición de tiempo de Interpolación Nearest Neighbor

Interpolación Nearest Neighbor									
Imagen	Tiempos (segundos)			Medidas de Rendimiento					
	secuencial	paralelos		Speed Up		Eficiencia		Costo	
pixeles	secuencial	N=2	N=4	N=2	N=4	N=2	N=4	N=2	N=4
900 x 600	0,271	0,879	2,301	0,308	0,118	0,077	0,059	1,233	0,236
1152 x 720	0,402	0,943	2,315	0,426	0,174	0,107	0,087	1,705	0,347
1280 x 800	0,573	0,966	2,318	0,593	0,247	0,148	0,124	2,373	0,494
1440 x 900	0,601	1,032	2,401	0,582	0,250	0,146	0,125	2,329	0,501
1920 x 1200	1,043	1,182	2,517	0,882	0,414	0,221	0,207	3,530	0,829

Fuente: Elaboración propia

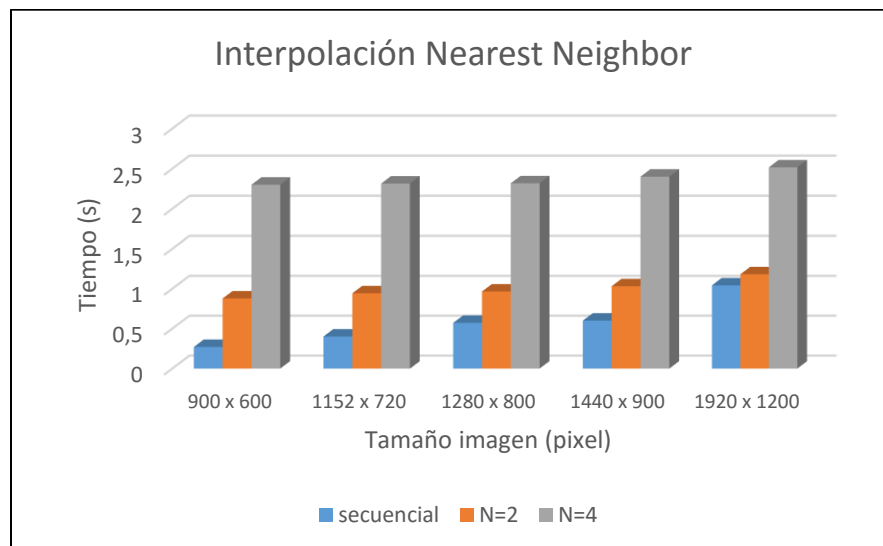


Tabla 23, Medición de tiempo de Convolución

Convolución									
Imagen	Tiempos (segundos)			Medidas de Rendimiento					
	secuencial	paralelos		Speed Up		Eficiencia		Costo	
pixeles	secuencial	N=2	N=4	N=2	N=4	N=2	N=4	N=2	N=4
900 x 600	10,877	10,628	6,732	1,023	1,616	0,256	0,808	4,094	3,231
1152 x 720	15,531	15,148	9,798	1,025	1,585	0,256	0,793	4,101	3,170
1280 x 800	19,107	18,314	12,171	1,043	1,570	0,261	0,785	4,173	3,140
1440 x 900	24,327	23,517	15,466	1,034	1,573	0,259	0,786	4,138	3,146
1920 x 1200	43,408	42,703	28,436	1,017	1,527	0,254	0,763	4,066	3,053

Fuente: Elaboración propia

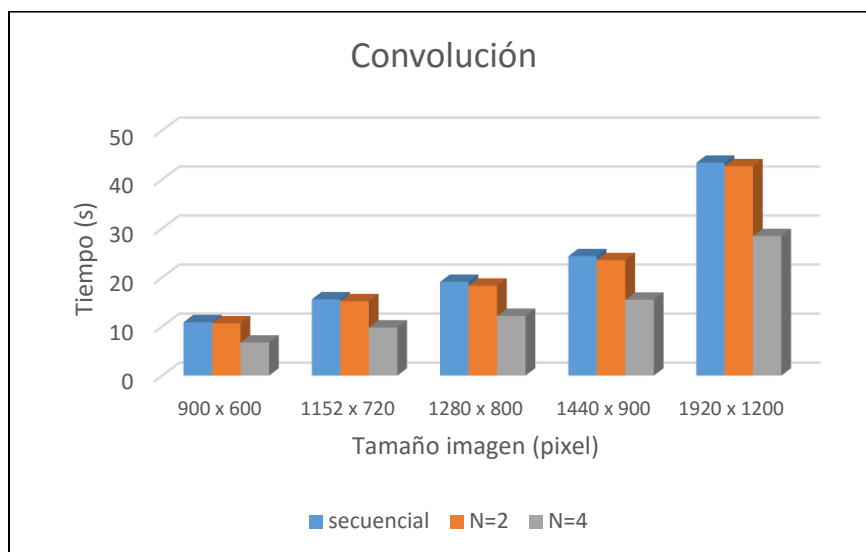
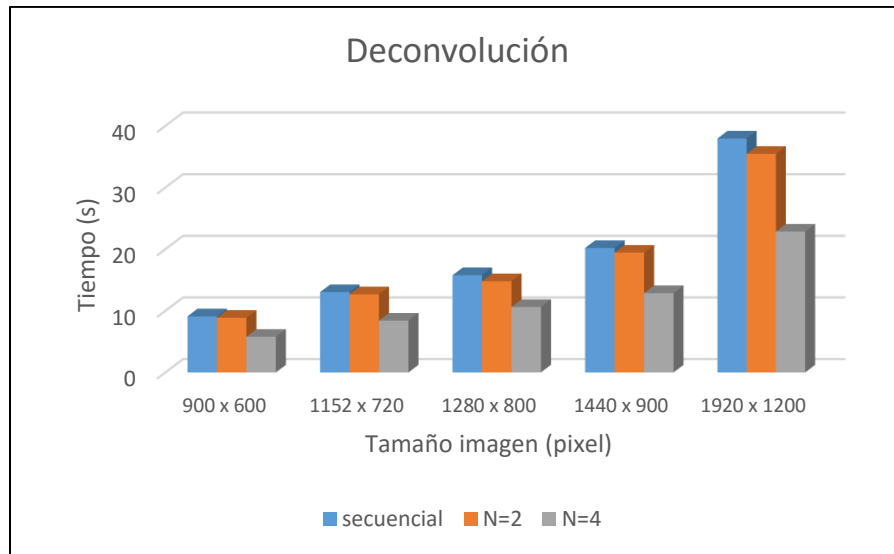


Tabla 24, Medición de tiempo de Desconvolución

Desconvolución									
Imagen	Tiempos (segundos)			Medidas de Rendimiento					
	secuencial	paralelos		Speed Up		Eficiencia		Costo	
pixeles	secuencial	N=2	N=4	N=2	N=4	N=2	N=4	N=2	N=4
900 x 600	9,097	8,862	5,783	1,027	1,573	0,257	0,787	4,106	3,146
1152 x 720	13,034	12,681	8,409	1,028	1,550	0,257	0,743	4,111	2,972
1280 x 800	15,784	14,792	10,622	1,067	1,486	0,267	0,784	4,268	3,136
1440 x 900	20,19	19,432	12,876	1,039	1,568	0,260	0,830	4,156	3,320
1920 x 1200	37,947	35,458	22,859	1,070	1,660	0,268	0,830	4,281	3,320

Fuente: Elaboración propia



## ANEXOS

---

## ANEXO A. Formato de carta de amonestación

### CARTA DE AMONESTACIÓN N° \_\_\_\_

Estimado Señor(a) \_\_\_\_\_.

Presente.

\_\_\_\_\_ de \_\_\_\_\_ del 2016.

Nos referimos a usted para informarle que su conducta observada, con relación al cumplimiento de sus deberes designados para el desarrollo del laboratorio fase II de Computación paralela. No ha sido satisfactoria. Demostrando una violación a los criterios de permanencia ya definidos.

Esto constituye una falta grave a sus obligaciones, siendo perjudicial tanto para usted como para sus propios compañeros. Situación, que por lo demás, se encuentra contemplada por su jefe de grupo y/o jefe de área \_\_\_\_\_.

Es por esto que, mediante esta carta de amonestación, requerimos que enmiende de manera inmediata su conducta y siga de manera estricta las instrucciones dadas por su jefe de área y/o su jefe de grupo. En caso contrario se adoptarán las medidas que se consideren necesarias, incluyendo la parcial o total exclusión del laboratorio fase II de Computación paralela.

Saluda atentamente.

\_\_\_\_\_  
Felipe S. Canales Saavedra

*Jefe de proyecto*

\_\_\_\_\_  
Camila A. Vera Castañeda

*Subjefe de proyecto*

\_\_\_\_\_  
Nelson F. Cariqueo Rojas

*Jefe área de diseño*

\_\_\_\_\_  
Diego I. García Castillo

*Jefe área de desarrollo*

\_\_\_\_\_  
Firma y Rut del afectado

OBSERVACIONES[illegible]



## ANEXO B. Formato de solicitud de aplazo

### SOLICITUD DE APLAZO

A la distinguida gerencia del proyecto.

Presente.

\_\_\_\_\_ de \_\_\_\_\_ del 2016.

Conforme a lo definido por los criterios de permanencia, como Jefe del grupo N° \_\_, es necesario solicitar una re-programación de la(s) tarea(s) \_\_\_\_\_.

Según los datos que manejamos, solicitamos una extensión del plazo por \_\_\_\_ (horas | días | semanas), por los siguientes motivos:

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_
4. \_\_\_\_\_
5. \_\_\_\_\_
6. \_\_\_\_\_
7. \_\_\_\_\_
8. \_\_\_\_\_
9. \_\_\_\_\_
10. \_\_\_\_\_

También, según lo definido en los criterios de permanencia, me permito a poner bajo su conocimiento los datos que avalan las razones anteriormente dichas.

En espera de su aprobación, se despide atentamente.

\_\_\_\_\_  
Nombre, Rut y firma del afectado

\_\_\_\_\_  
Jefe de proyecto

\_\_\_\_\_  
Subjefe de proyecto

\_\_\_\_\_  
Jefe área de diseño

\_\_\_\_\_  
Jefe área de desarrollo

## ANEXO C. Pauta de evaluaciones parciales

Tabla 25, Pauta de evaluaciones parciales

Área de interés	Puntaje (1-5)	Factor	Resultado
Es ordenado para trabajar.		3	0
Realiza lo asignado en el tiempo establecido.		5	0
Cumple con los objetivos.		5	0
Es proactivo y tiene improvisación a la hora de trabajar.		3	0
Menciona ideas constructivas sobre las tareas asignadas.		2	0
Es independiente para trabajar.		4	0
Reacciona de manera calmada a situaciones de estrés.		2	0
Mantiene a su Jefe de grupo informado de su progreso.		3	0
Asume responsabilidades y consecuencias negativas, demostrando deseos de mejorar.		3	0
Tiene conocimiento (a grandes rasgos) de las tareas que realizan los demás grupos.		3	0
Colabora con sus compañeros en sus momentos libres.		5	0
Promueve un clima positivo en el laboratorio.		2	0
Colabora y no discrimina a su compañero de tareas.		1	0
Respetar las decisiones de sus jefes.		2	0
Asiste a las reuniones.		3	0
<b>TOTAL</b>			<b>0</b>
<b>NOTA</b>			<b>0,0</b>

Fuente: Elaboración propia

## ANEXO D. Pauta de evaluación final

Tabla 26, Pauta de evaluación final

Área de interés	Puntaje (1-5)	Factor	Resultado
Demuestra compromiso para el proyecto.		5	0
Es confiable al realizar las tareas designadas.		3	0
Asertivo, claro y conciso al momento de expresarse.		5	0
Es capaz de aceptar críticas y/o sugerencias con madurez.		3	0
No demuestra problemas para trabajar bajo presión.		3	0
Demuestra calidad en su trabajo.		4	0
Cumple con los plazos establecidos.		5	0
Cumple con informar de sus avances.		3	0
Demuestra conocimientos acordes a su grado de estudios.		3	0
Mantiene un comportamiento acorde a su edad.		2	0
Tiene facilidad para el trabajo.		3	0
<b>Total</b>			<b>0</b>
<b>Nota</b>			<b>0,0</b>

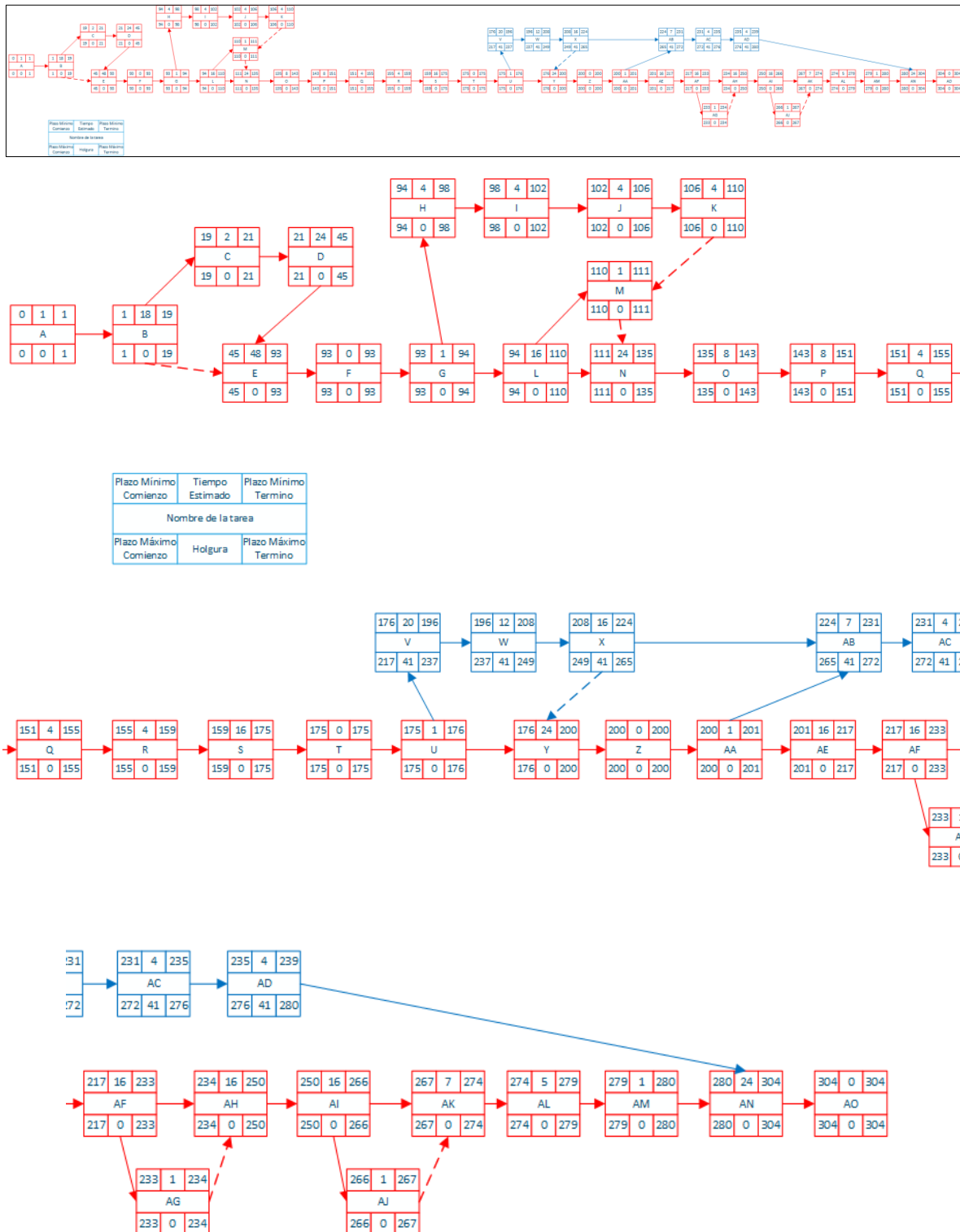
Fuente: Elaboración propia

## ANEXO E. CPM del proyecto

ID	Actividad	Act.	Predec.	Tiempo Estimado (Horas)	Plazo Mínimo		Plazo Máximo		Holgura	Ruta Crítica
					Comienzo	Término	Comienzo	Término		
1	Reunión N°1	A		1	0	1	0	1	0	
2	Planificación de Actividades	B	A	18	1	19	1	19	0	
3	División de tareas e información a investigar	C	B	2	19	21	19	21	0	
4	Investigación para propuesta	D	C	24	21	45	21	45	0	
5	Confección de la propuesta	E	B;D	48	45	93	45	93	0	
6	Entrega de Informe de Propuesta	F	E	0	93	93	93	93	0	
7	Reunión N°2	G	F	1	93	94	93	94	0	
8	Instalación de Ubuntu 14.04	H	G	4	94	98	94	98	0	
9	Instalación y configuración del Software necesario	I	H	4	98	102	98	102	0	
10	Construcción del clúster	J	I	4	102	106	102	106	0	
11	Ejecutar programa de prueba en clúster	K	J	4	106	110	106	110	0	
12	Investigación de diferentes algoritmos para la gestión de imagen	L	G	16	94	110	94	110	0	
13	Reunión N°3	M	K;L	1	110	111	110	111	0	
14	Modelar solución de ambiente secuencial	N	L;M	24	111	135	111	135	0	
15	Programación solución de ambiente secuencial	O	N	8	135	143	135	143	0	
16	Optimizar solución de ambiente secuencial	P	O	8	143	151	143	151	0	
17	Calcular métricas de solución de ambiente secuencial	Q	P	4	151	155	151	155	0	
18	Documentación y análisis de QA de ambiente secuencial	R	Q	4	155	159	155	159	0	
19	Confección Informe Avance N°1	S	R	16	159	175	159	175	0	
20	Entrega Informe Avance N°1	T	S	0	175	175	175	175	0	
21	Reunión N°4	U	T	1	175	176	175	176	0	
22	Modelar solución de ambiente paralelo	V	U	20	176	196	217	237	41	
23	Programación solución de ambiente paralelo	W	V	12	196	208	237	249	41	
24	Optimizar solución de ambiente paralelo	X	W	16	208	224	249	265	41	
25	Confección Informe Avance N°2	Y	U	24	176	200	176	200	0	
26	Entrega Informe Avance N°2	Z	Y	0	200	200	200	200	0	
27	Reunión N°5	AA	Z	1	200	201	200	201	0	
28	Calcular métricas de solución de ambiente paralelo	AB	X;AA	7	224	231	265	272	41	
29	Documentación y análisis de QA de ambiente paralelo	AC	AB	4	231	235	272	276	41	
30	Comparar métricas algoritmo secuencial versus paralelo	AD	AC	4	235	239	276	280	41	
31	Modelar plataforma web	AE	AA	16	201	217	201	217	0	
32	Integración de la aplicación paralela a la plataforma web	AF	AE	16	217	233	217	233	0	
33	Reunión N°6	AG	AF	1	233	234	233	234	0	
34	Programación interna de plataforma web	AH	AF;AG	16	234	250	234	250	0	
35	Modelar Interfaz gráfica de plataforma web	AI	AH	16	250	266	250	266	0	
36	Reunión N°7	AJ	AI	1	266	267	266	267	0	
37	Programación de la interfaz gráfica de plataforma web	AK	AI;AJ	7	267	274	267	274	0	

ID	Actividad	Act.	Predec.	Tiempo Estimado (Horas)	Plazo Mínimo		Plazo Máximo		Holgura	Ruta Crítica
					Comienzo	Término	Comienzo	Término		
39	Reunión N°8	AM	AL	1	279	280	279	280	0	
40	Confección Informe Final de Laboratorio Fase II	AN	AD;AM	24	280	304	280	304	0	
41	Entrega Informe Final de Laboratorio Fase II	AO	AN	0	304	304	304	304	0	

## ANEXO F. CPM, ruta crítica



## ANEXO E. WBS de la aplicación



## ANEXO F. Plan de hitos

Hito		Fecha	Avance Planificado	
N°	Descripción		según Costo (UF)	según Esfuerzo (HH)
1	Entrega Informe de Propuesta de Laboratorio Fase II	Viernes 03 de Junio	18%	16%
2	Entrega Informe de Avance N°1	Viernes 17 de Junio	48%	48%
3	Entrega Informe de Avance N°2	Viernes 24 de Junio	67%	66%
4	Entrega Informe Final de Laboratorio Fase II	Viernes 8 de Julio	100%	100%



## ANEXO G. Total de esfuerzos y costos

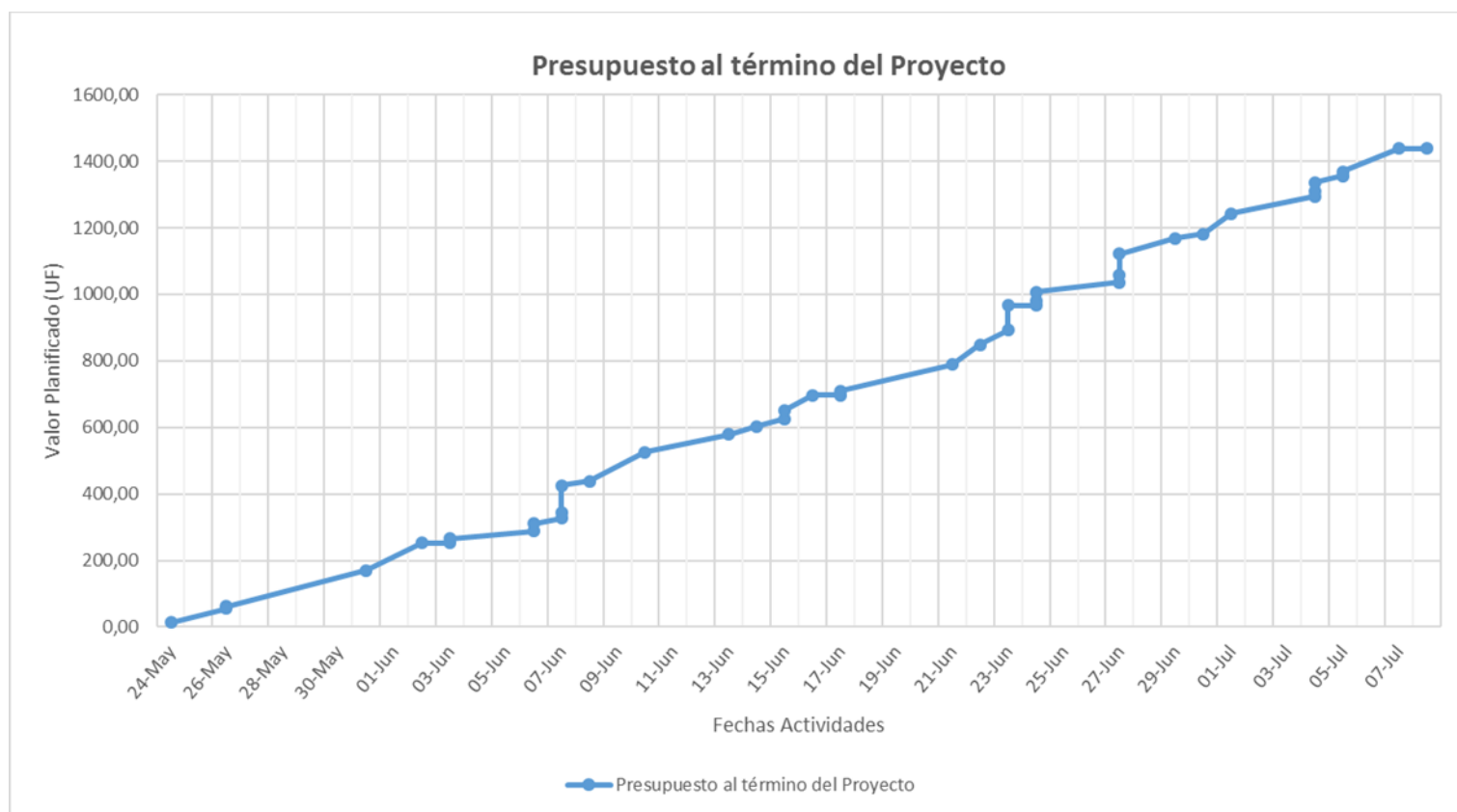
Tabla 27, Tabla de esfuerzos y costos

ID	Actividad	Plan Inicio	Plan Terminación	TOTAL					
				Costo (UF)	Costo Acumulado	Avance Plan. según Costo	Esfuerzo (HH)	Esfuerzo Acumulado	Avance Plan. según Esfuerzo
1	Reunion N°1	24-May	24-May	13,20	13,20		16	16	
2	Planificación de Actividades	24-May	26-May	42,60	55,80		32	48	
3	Division de tareas e informacion a investigar	26-May	26-May	5,80	61,60		5	53	
4	Investigacion para propuesta	26-May	31-May	108,80	170,40		128	181	
5	Confeccion de la propuesta	26-May	02-Jun	82,50	252,90		68	249	
6	Entrega de Informe de Propuesta	03-Jun	03-Jun	0,00	252,90	18%	0	249	16%
7	Reunion N°2	03-Jun	03-Jun	13,20	266,10		16	265	
8	Instalacion de Ubuntu 14.04	06-Jun	06-Jun	22,40	288,50		32	297	
9	Instalacion y configuracion del Software necesario	06-Jun	06-Jun	22,40	310,90		32	329	
10	Construccion del cluster	07-Jun	07-Jun	16,80	327,70		28	357	
11	Ejecutar programa de prueba en cluster	07-Jun	07-Jun	16,80	344,50		28	385	
12	Investigacion de diferentes algoritmos para la gestión de imagen	06-Jun	07-Jun	80,80	425,30		108	493	
13	Reunión N°3	08-Jun	08-Jun	13,20	438,50		16	509	
14	Modelar solucion de ambiente secuencial	08-Jun	10-Jun	86,40	524,90		80	589	
15	Programacion solucion de ambiente secuencial	13-Jun	13-Jun	53,60	578,50		77	666	
16	Optimizar solucion de ambiente secuencial	14-Jun	14-Jun	24,80	603,30		22	688	
17	Calcular metricas de solucion de ambiente secuencial	15-Jun	15-Jun	22,90	626,20		26	714	
18	Documentacion y analisis de QA de ambiente secuencial	15-Jun	15-Jun	25,50	651,70		28	742	
19	Confeccion Informe Avance N°1	15-Jun	16-Jun	44,40	696,10		34	776	
20	Entrega Informe Avance N°1	17-Jun	17-Jun	0,00	696,10	48%	0	776	48%
21	Reunión N°4	17-Jun	17-Jun	13,20	709,30		16	792	
22	Modelar solucion de ambiente paralelo	17-Jun	21-Jun	79,80	789,10		76	868	
23	Programacion solucion de ambiente paralelo	21-Jun	22-Jun	60,20	849,30		93	961	
24	Optimizar solucion de ambiente paralelo	22-Jun	23-Jun	42,80	892,10		37	998	
25	Confeccion Informe Avance N°2	21-Jun	23-Jun	74,60	966,70		57	1055	
26	Entrega Informe Avance N°2	24-Jun	24-Jun	0,00	966,70	67%	0	1055	66%

27	Reunion N°5	24-Jun	24-Jun	13,20	979,90		16	1071	
28	Calcular metricas de solucion de ambiente paralelo	24-Jun	24-Jun	26,70	1006,60		29	1100	
29	Documentacion y analisis de QA de ambiente paralelo	27-Jun	27-Jun	29,40	1036,00		31	1131	
30	Comparar métricas algoritmo secuencial versus paralelo	27-Jun	27-Jun	21,00	1057,00		17	1148	
31	Modelar plataforma web	24-Jun	27-Jun	64,00	1121,00		64	1212	
32	Integración de la aplicación paralela a la plataforma web	28-Jun	29-Jun	47,00	1168,00		50	1262	
33	Reunion N°6	30-Jun	30-Jun	13,20	1181,20		16	1278	
34	Programación interna de plataforma web	30-Jun	01-Jul	61,60	1242,80		108	1386	
35	Modelar Interfaz gráfica de plataforma web	01-Jul	04-Jul	52,40	1295,20		54	1440	
36	Reunion N°7	04-Jul	04-Jul	13,20	1308,40		16	1456	
37	Programación de la interfaz gráfica de plataforma web	04-Jul	04-Jul	28,40	1336,80		49	1505	
38	Montar el servidor para la plataforma	05-Jul	05-Jul	18,80	1355,60		33	1538	
39	Reunion N°8	05-Jul	05-Jul	13,20	1368,80		16	1554	
40	Confeccion Informe Final de Laboratorio Fase II	05-Jul	07-Jul	68,60	1437,40		52	1606	
41	Entrega Informe Final de Laboratorio Fase II	08-Jul	08-Jul	0,00	1437,40	100%	0	1606	100%
<b>TOTAL</b>				<b>1437,40</b>			<b>1606</b>		

Fuente: Elaboración propia

Ilustración 45, Gráfico de presupuesto al término del proyecto



Fuente: Elaboración propia

## ANEXO G. Evaluaciones

### Evaluaciones de etapa final

	GRUPO 1											
	Elías Cáceres			Gabriel Sanhueza			Katherine Sáez			Mario Urbina		
Área de interés	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado
Demuestra compromiso para el proyecto.	4	5	20	5	5	25	4	5	20	4	5	20
Es confiable al realizar las tareas designadas.	3	3	9	5	3	15	4	3	12	4	3	12
Asertivo, claro y conciso al momento de expresarse.	4	5	20	4	5	20	5	5	25	3	5	15
Es capaz de aceptar críticas y/o sugerencias con madurez.	5	3	15	5	3	15	5	3	15	4	3	12
No demuestra problemas para trabajar bajo presión.	4	3	12	5	3	15	5	3	15	3	3	9
Demuestra calidad en su trabajo.	3	4	12	5	4	20	4	4	16	3	4	12
Cumple con los plazos establecidos.	4	5	20	5	5	25	4	5	20	4	5	20
Cumple con informar de sus avances.	5	3	15	5	3	15	4	3	12	3	3	9
Demuestra conocimientos acorde a su grado de estudios.	3	3	9	5	3	15	5	3	15	4	3	12
Mantiene un comportamiento acorde a su edad.	5	2	10	5	2	10	5	2	10	4	2	8
Tiene facilidad para el trabajo.	4	3	12	5	3	15	5	3	15	4	3	12
<b>Total</b>			<b>154</b>			<b>190</b>			<b>175</b>			<b>141</b>
<b>Nota</b>			<b>5,5</b>			<b>6,8</b>			<b>6,3</b>			<b>5,1</b>

Ilustración 46, Tabla evaluaciones finales de grupo 1

Fuente: Elaboración propia

	GRUPO 2											
	Nicolás Soto			Álvaro Cabedo			Dominique Hueteleo			Boris Vásquez		
Área de interés	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado
Demuestra compromiso para el proyecto.	4	5	20	5	5	25	5	5	25	1	5	5
Es confiable al realizar las tareas designadas.	4	3	12	5	3	15	5	3	15	1	3	3
Asertivo, claro y conciso al momento de expresarse.	5	5	25	5	5	25	5	5	25	1	5	5
Es capaz de aceptar críticas y/o sugerencias con madurez.	5	3	15	5	3	15	5	3	15	3	3	9
No demuestra problemas para trabajar bajo presión.	4	3	12	5	3	15	5	3	15	2	3	6
Demuestra calidad en su trabajo.	4	4	16	5	4	20	4	4	16	1	4	4
Cumple con los plazos establecidos.	1	5	5	4	5	20	4	5	20	1	5	5
Cumple con informar de sus avances.	5	3	15	5	3	15	5	3	15	2	3	6
Demuestra conocimientos acorde a su grado de estudios.	4	3	12	4	3	12	4	3	12	3	3	9
Mantiene un comportamiento acorde a su edad.	5	2	10	5	2	10	5	2	10	4	2	8
Tiene facilidad para el trabajo.	5	3	15	4	3	12	4	3	12	1	3	3
<b>Total</b>			<b>157</b>			<b>184</b>			<b>180</b>			<b>63</b>
<b>Nota</b>			<b>5,6</b>			<b>6,6</b>			<b>6,5</b>			<b>2,3</b>

Ilustración 47, Tabla de evaluaciones finales de grupo 2

Fuente: Elaboración propia

	GRUPO 3											
	Gladys Cerro			Felipe Abarca			David Martinez			Jonathan Acuña		
Área de interés	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado
Demuestra compromiso para el proyecto.	5	5	25	5	5	25	5	5	25	5	5	25
Es confiable al realizar las tareas designadas.	5	3	15	5	3	15	5	3	15	5	3	15
Asertivo, claro y conciso al momento de expresarse.	5	5	25	5	5	25	5	5	25	5	5	25
Es capaz de aceptar críticas y/o sugerencias con madurez.	5	3	15	5	3	15	5	3	15	5	3	15
No demuestra problemas para trabajar bajo presión.	5	3	15	5	3	15	5	3	15	5	3	15
Demuestra calidad en su trabajo.	5	4	20	5	4	20	5	4	20	5	4	20
Cumple con los plazos establecidos.	5	5	25	5	5	25	5	5	25	5	5	25
Cumple con informar de sus avances.	5	3	15	5	3	15	4	3	12	4	3	12
Demuestra conocimientos acorde a su grado de estudios.	5	3	15	5	3	15	5	3	15	5	3	15
Mantiene un comportamiento acorde a su edad.	5	2	10	5	2	10	5	2	10	5	2	10
Tiene facilidad para el trabajo.	5	3	15	5	3	15	5	3	15	5	3	15
<b>Total</b>			<b>195</b>			<b>195</b>			<b>192</b>			<b>192</b>
<b>Nota</b>			<b>7,0</b>			<b>7,0</b>			<b>6,9</b>			<b>6,9</b>

Ilustración 49, Tabla de evaluaciones finales de grupo 3

Fuente: Elaboración propia

	GRUPO LIDER											
	Felipe Canales			Camila Vera			Nelson Carique			Diego Garcia		
Área de interés	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado	Puntaje (1-5)	Factor	Resultado
Demuestra compromiso para el proyecto.	5	5	25	5	5	25	5	5	25	5	5	25
Es confiable al realizar las tareas designadas.	5	3	15	5	3	15	5	3	15	5	3	15
Asertivo, claro y conciso al momento de expresarse.	5	5	25	5	5	25	5	5	25	5	5	25
Es capaz de aceptar críticas y/o sugerencias con madurez.	5	3	15	5	3	15	5	3	15	5	3	15
No demuestra problemas para trabajar bajo presión.	5	3	15	5	3	15	5	3	15	5	3	15
Demuestra calidad en su trabajo.	5	4	20	5	4	20	5	4	20	5	4	20
Cumple con los plazos establecidos.	5	5	25	5	5	25	5	5	25	5	5	25
Cumple con informar de sus avances.	5	3	15	5	3	15	5	3	15	5	3	15
Demuestra conocimientos acorde a su grado de estudios.	5	3	15	5	3	15	5	3	15	5	3	15
Mantiene un comportamiento acorde a su edad.	5	2	10	5	2	10	5	2	10	5	2	10
Tiene facilidad para el trabajo.	5	3	15	5	3	15	5	3	15	5	3	15
<b>Total</b>			<b>195</b>			<b>195</b>			<b>195</b>			<b>195</b>
<b>Nota</b>			<b>7,0</b>			<b>7,0</b>			<b>7,0</b>			<b>7,0</b>

Ilustración 48, Tabla de evaluaciones finales grupo líder

Fuente: Elaboración propia

## Evaluaciones finales

En la tabla a seguir, se detalla el resumen de las evaluaciones por las tres etapas logradas en este proyecto, incluyendo la evaluación final, la cual es el promedio aritmético de las tres anteriores.

*Tabla 28, Resumen de evaluación por etapas y la evaluación final*

Etapas	Elías Cáceres	Gabriel Sanhueza	Katherine Sáez	Mario Urbina
Etapas 1	6,8	6,8	4,8	4
Etapas 2	7	6,5	5,4	5
Etapas Final	5,5	6,8	6,3	5,1
<b>Evaluacion final</b>	<b>6,4</b>	<b>6,7</b>	<b>5,5</b>	<b>4,7</b>

Etapas	Nicolás Soto	Álvaro Cabedo	Dominique Hueteleo	Boris Vásquez
Etapas 1	7	6,8	6,6	4,1
Etapas 2	7	6,5	6,5	4
Etapas Final	5,6	6,6	6,5	2,3
<b>Evaluacion final</b>	<b>6,5</b>	<b>6,6</b>	<b>6,5</b>	<b>3,5</b>

Etapas	Gladys Cerro	Felipe Abarca	David Martinez	Jonathan Acuña
Etapas 1	7	6	7	7
Etapas 2	7	7	6,6	6,5
Etapas Final	5	7	6,9	6,9
<b>Evaluacion final</b>	<b>6,3</b>	<b>6,7</b>	<b>6,8</b>	<b>6,8</b>

Etapas	Felipe Canales	Camila vera	Nelson Cariqueo	Diego García
Etapas 1	6,9	7	7	7
Etapas 2	6,6	7	7	7
Etapas Final	7	7	7	7
<b>Evaluacion final</b>	<b>6,8</b>	<b>7,0</b>	<b>7,0</b>	<b>7,0</b>

Fuente: Elaboración propia

## ANEXO I. Cronograma final

Tabla 29, Cronograma final

ID	Actividad	Felipe Canales				Camila Vera			
		Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)	Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	1,40	1,40	100%	1	1,30	1,30
2	Planificacion de Actividades	100%	18	1,40	25,20	33%	6	1,30	7,80
3	Division de tareas e informacion a investigar	100%	2	1,40	2,80	0%	0	1,30	0,00
4	Investigacion para propuesta	100%	24	1,40	33,60	33%	8	1,30	10,40
5	Confeccion de la propuesta	50%	24	1,40	33,60	50%	24	1,30	31,20
6	Entrega de Informe de Propuesta	-	-	1,40	-	-	-	1,30	-
7	Reunion N°2	100%	1	1,40	1,40	100%	1	1,30	1,30
8	Instalacion de Ubuntu 14.04	100%	4	1,40	5,60	0%	0	1,30	0,00
9	Instalacion y configuracion del Software necesario	100%	4	1,40	5,60	0%	0	1,30	0,00
10	Construccion del cluster	0%	0	1,40	0,00	0%	0	1,30	0,00
11	Ejecutar programa de prueba en cluster	0%	0	1,40	0,00	0%	0	1,30	0,00
12	Investigacion de diferentes algoritmos para la gestión de imagen	50%	8	1,40	11,20	0%	0	1,30	0,00
13	Reunión N°3	100%	1	1,40	1,40	100%	1	1,30	1,30
14	Modelar solucion de ambiente secuencial	100%	24	1,40	33,60	0%	0	1,30	0,00
15	Programacion solucion de ambiente secuencial	13%	1	1,40	1,40	0%	0	1,30	0,00
16	Optimizar solucion de ambiente secuencial	50%	4	1,40	5,60	0%	0	1,30	0,00
17	Calcular metricas de solucion de ambiente secuencial	0%	0	1,40	0,00	25%	1	1,30	1,30
18	Documentacion y analisis de QA de ambiente secuencial	25%	1	1,40	1,40	25%	1	1,30	1,30
19	Confeccion Informe Avance N°1	100%	16	1,40	22,40	63%	10	1,30	13,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	1,40	1,40	100%	1	1,30	1,30

22	Modelar solución de ambiente paralelo	75%	15	1,40	21,00	0%	0	1,30	0,00
23	Programación solución de ambiente paralelo	8%	1	1,40	1,40	0%	0	1,30	0,00
24	Optimizar solución de ambiente paralelo	25%	4	1,40	5,60	0%	0	1,30	0,00
25	Confección Informe Avance N°2	100%	24	1,40	33,60	83%	20	1,30	26,00
26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-
27	Reunión N°5	100%	1	1,40	1,40	100%	1	1,30	1,30
28	Calcular métricas de solución de ambiente paralelo	14%	1	1,40	1,40	14%	1	1,30	1,30
29	Documentación y análisis de QA de ambiente paralelo	50%	2	1,40	2,80	50%	2	1,30	2,60
30	Comparar métricas algoritmo secuencial versus paralelo	100%	4	1,40	5,60	100%	4	1,30	5,20
31	Modelar plataforma web	50%	8	1,40	11,20	0%	0	1,30	0,00
32	Integración de la aplicación paralela a la plataforma web	25%	4	1,40	5,60	0%	0	1,30	0,00
33	Reunión N°6	100%	1	1,40	1,40	100%	1	1,30	1,30
34	Programación interna de plataforma web	13%	2	1,40	2,80	0%	0	1,30	0,00
35	Modelar Interfaz gráfica de plataforma web	25%	4	1,40	5,60	0%	0	1,30	0,00
36	Reunión N°7	100%	1	1,40	1,40	100%	1	1,30	1,30
37	Programación de la interfaz gráfica de plataforma web	14%	1	1,40	1,40	0%	0	1,30	0,00
38	Montar el servidor para la plataforma	20%	1	1,40	1,40	0%	0	1,30	0,00
39	Reunión N°8	100%	1	1,40	1,40	100%	1	1,30	1,30
40	Confección Informe Final de Laboratorio Fase II	100%	24	1,40	33,60	83%	20	1,30	26,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-
<b>TOTAL</b>			<b>233,00</b>		<b>326,20</b>		<b>105,00</b>		<b>136,50</b>



ID	Actividad	Nelson Cariqueo				Diego García			
		Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)	Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	1,20	1,20	0%	1	1,20	1,20
2	Planificacion de Actividades	22%	4	1,20	4,80	22%	4	1,20	4,80
3	Division de tareas e informacion a investigar	0%	0	1,20	0,00	0%	0	1,20	0,00
4	Investigacion para propuesta	0%	0	1,20	0,00	0%	0	1,20	0,00
5	Confeccion de la propuesta	17%	8	1,20	9,60	0%	0	1,20	0,00
6	Entrega de Informe de Propuesta	-	-	1,20	-	-	-	-	-
7	Reunion N°2	100%	1	1,20	1,20	100%	1	1,20	1,20
8	Instalacion de Ubuntu 14.04	0%	0	1,20	0,00	100%	4	1,20	4,80
9	Instalacion y configuracion del Software necesario	0%	0	1,20	0,00	100%	4	1,20	4,80
10	Construccion del cluster	0%	0	1,20	0,00	100%	4	1,20	4,80
11	Ejecutar programa de prueba en cluster	0%	0	1,20	0,00	100%	4	1,20	4,80
12	Investigacion de diferentes algoritmos para la gestión de imagen	0%	0	1,20	0,00	25%	4	1,20	4,80
13	Reunión N°3	100%	1	1,20	1,20	100%	1	1,20	1,20
14	Modelar solucion de ambiente secuencial	67%	16	1,20	19,20	17%	4	1,20	4,80
15	Programacion solucion de ambiente secuencial	25%	2	1,20	2,40	100%	8	1,20	9,60
16	Optimizar solucion de ambiente secuencial	25%	2	1,20	2,40	50%	4	1,20	4,80
17	Calcular metricas de solucion de ambiente secuencial	25%	1	1,20	1,20	0%	0	1,20	0,00
18	Documentacion y analisis de QA de ambiente secuencial	50%	2	1,20	2,40	0%	0	1,20	0,00
19	Confeccion Informe Avance N°1	31%	5	1,20	6,00	0%	0	1,20	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	1,20	1,20	100%	1	1,20	1,20

22	Modelar solución de ambiente paralelo	25%	5	1,20	6,00	100%	20	1,20	24,00
23	Programación solución de ambiente paralelo	17%	2	1,20	2,40	100%	12	1,20	14,40
24	Optimizar solución de ambiente paralelo	31%	5	1,20	6,00	100%	16	1,20	19,20
25	Confección Informe Avance N°2	42%	10	1,20	12,00	0%	0	1,20	0,00
26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-
27	Reunión N°5	100%	1	1,20	1,20	100%	1	1,20	1,20
28	Calcular métricas de solución de ambiente paralelo	14%	1	1,20	1,20	29%	2	1,20	2,40
29	Documentación y análisis de QA de ambiente paralelo	25%	1	1,20	1,20	50%	2	1,20	2,40
30	Comparar métricas algoritmo secuencial versus paralelo	100%	4	1,20	4,80	50%	2	1,20	2,40
31	Modelar plataforma web	100%	16	1,20	19,20	25%	4	1,20	4,80
32	Integración de la aplicación paralela a la plataforma web	50%	8	1,20	9,60	50%	8	1,20	9,60
33	Reunión N°6	100%	1	1,20	1,20	100%	1	1,20	1,20
34	Programación interna de plataforma web	13%	2	1,20	2,40	13%	2	1,20	2,40
35	Modelar Interfaz gráfica de plataforma web	25%	4	1,20	4,80	100%	16	1,20	19,20
36	Reunión N°7	100%	1	1,20	1,20	100%	1	1,20	1,20
37	Programación de la interfaz gráfica de plataforma web	0%	0	1,20	0,00	0%	0	1,20	0,00
38	Montar el servidor para la plataforma	20%	1	1,20	1,20	20%	1	1,20	1,20
39	Reunión N°8	100%	1	1,20	1,20	100%	1	1,20	1,20
40	Confección Informe Final de Laboratorio Fase II	21%	5	1,20	6,00	0%	0	1,20	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-
<b>TOTAL</b>			<b>112,00</b>		<b>134,40</b>		<b>133,00</b>		<b>159,60</b>

ID	Actividad	Elías Cáceres				Gabriel Sanhueza			
		Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)	Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	1,00	1,00	100%	1	0,50	0,50
2	Planificacion de Actividades	17%	3	1,00	3,00	0%	0	0,50	0,00
3	Division de tareas e informacion a investigar	50%	1	1,00	1,00	50%	1	0,50	0,50
4	Investigacion para propuesta	42%	10	1,00	10,00	21%	5	0,50	2,50
5	Confeccion de la propuesta	4%	2	1,00	2,00	4%	2	0,50	1,00
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	1,00	1,00	0%	0	0,50	0,00
8	Instalacion de Ubuntu 14.04	75%	3	1,00	3,00	75%	3	0,50	1,50
9	Instalacion y configuracion del Software necesario	75%	3	1,00	3,00	25%	1	0,50	0,50
10	Construccion del cluster	75%	3	1,00	3,00	25%	1	0,50	0,50
11	Ejecutar programa de prueba en cluster	50%	2	1,00	2,00	50%	2	0,50	1,00
12	Investigacion de diferentes algoritmos para la gestión de imagen	50%	8	1,00	8,00	50%	8	0,50	4,00
13	Reunión N°3	100%	1	1,00	1,00	100%	1	0,50	0,50
14	Modelar solucion de ambiente secuencial	17%	4	1,00	4,00	17%	4	0,50	2,00
15	Programacion solucion de ambiente secuencial	163%	13	1,00	13,00	313%	25	0,50	12,50
16	Optimizar solucion de ambiente secuencial	63%	5	1,00	5,00	63%	5	0,50	2,50
17	Calcular metricas de solucion de ambiente secuencial	100%	4	1,00	4,00	100%	4	0,50	2,00
18	Documentacion y analisis de QA de ambiente secuencial	125%	5	1,00	5,00	125%	5	0,50	2,50
19	Confeccion Informe Avance N°1	25%	4	1,00	4,00	0%	0	0,50	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	1,00	1,00	100%	1	0,50	0,50
22	Modelar solucion de ambiente paralelo	20%	4	1,00	4,00	0%	0	0,50	0,00
23	Programacion solucion de ambiente paralelo	100%	12	1,00	12,00	208%	25	0,50	12,50
24	Optimizar solucion de ambiente paralelo	50%	8	1,00	8,00	50%	8	0,50	4,00
25	Confeccion Informe Avance N°2	13%	3	1,00	3,00	8%	2	0,50	1,00

26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	1,00	1,00	100%	1	0,50	0,50
28	Calcular métricas de solución de ambiente paralelo	57%	4	1,00	4,00	0%	0	0,50	0,00
29	Documentación y análisis de QA de ambiente paralelo	100%	4	1,00	4,00	0%	0	0,50	0,00
30	Comparar métricas algoritmo secuencial versus paralelo	25%	1	1,00	1,00	0%	0	0,50	0,00
31	Modelar plataforma web	25%	4	1,00	4,00	0%	0	0,50	0,00
32	Integración de la aplicación paralela a la plataforma web	25%	4	1,00	4,00	13%	2	0,50	1,00
33	Reunion N°6	100%	1	1,00	1,00	100%	1	0,50	0,50
34	Programación interna de plataforma web	13%	2	1,00	2,00	100%	16	0,50	8,00
35	Modelar Interfaz gráfica de plataforma web	13%	2	1,00	2,00	0%	0	0,50	0,00
36	Reunion N°7	100%	1	1,00	1,00	100%	1	0,50	0,50
37	Programación de la interfaz gráfica de plataforma web	29%	2	1,00	2,00	100%	7	0,50	3,50
38	Montar el servidor para la plataforma	0%	0	1,00	0,00	100%	5	0,50	2,50
39	Reunion N°8	100%	1	1,00	1,00	100%	1	0,50	0,50
40	Confección Informe Final de Laboratorio Fase II	4%	1	1,00	1,00	0%	0	0,50	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-
<b>TOTAL</b>			<b>129,00</b>		<b>129,00</b>		<b>138,00</b>		<b>69,00</b>

ID	Actividad	Katherine Sáez				Mario Urbina			
		Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)	Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	0,70	0,70	100%	1	0,50	0,50
2	Planificacion de Actividades	0%	0	0,70	0,00	0%	0	0,50	0,00
3	Division de tareas e informacion a investigar	50%	1	0,70	0,70	50%	1	0,50	0,50
4	Investigacion para propuesta	21%	5	0,70	3,50	21%	5	0,50	2,50
5	Confeccion de la propuesta	4%	2	0,70	1,40	4%	2	0,50	1,00
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	0,70	0,70	100%	1	0,50	0,50
8	Instalacion de Ubuntu 14.04	25%	1	0,70	0,70	25%	1	0,50	0,50
9	Instalacion y configuracion del Software necesario	75%	3	0,70	2,10	75%	3	0,50	1,50
10	Construccion del cluster	75%	3	0,70	2,10	75%	3	0,50	1,50
11	Ejecutar programa de prueba en cluster	50%	2	0,70	1,40	50%	2	0,50	1,00
12	Investigacion de diferentes algoritmos para la gestión de imagen	50%	8	0,70	5,60	50%	8	0,50	4,00
13	Reunión N°3	100%	1	0,70	0,70	100%	1	0,50	0,50
14	Modelar solucion de ambiente secuencial	17%	4	0,70	2,80	17%	4	0,50	2,00
15	Programacion solucion de ambiente secuencial	88%	7	0,70	4,90	75%	6	0,50	3,00
16	Optimizar solucion de ambiente secuencial	38%	3	0,70	2,10	38%	3	0,50	1,50
17	Calcular metricas de solucion de ambiente secuencial	100%	4	0,70	2,80	100%	4	0,50	2,00
18	Documentacion y analisis de QA de ambiente secuencial	125%	5	0,70	3,50	125%	5	0,50	2,50
19	Confeccion Informe Avance N°1	25%	4	0,70	2,80	25%	4	0,50	2,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	0,70	0,70	100%	1	0,50	0,50
22	Modelar solucion de ambiente paralelo	0%	0	0,70	0,00	0%	0	0,50	0,00
23	Programacion solucion de ambiente paralelo	150%	18	0,70	12,60	208%	25	0,50	12,50
24	Optimizar solucion de ambiente paralelo	50%	8	0,70	5,60	50%	8	0,50	4,00
25	Confeccion Informe Avance N°2	8%	2	0,70	1,40	8%	2	0,50	1,00

26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	0,70	0,70	100%	1	0,50	0,50
28	Calcular métricas de solución de ambiente paralelo	57%	4	0,70	2,80	0%	0	0,50	0,00
29	Documentación y análisis de QA de ambiente paralelo	100%	4	0,70	2,80	0%	0	0,50	0,00
30	Comparar métricas algoritmo secuencial versus paralelo	0%	0	0,70	0,00	0%	0	0,50	0,00
31	Modelar plataforma web	50%	8	0,70	5,60	0%	0	0,50	0,00
32	Integración de la aplicación paralela a la plataforma web	13%	2	0,70	1,40	13%	2	0,50	1,00
33	Reunion N°6	100%	1	0,70	0,70	100%	1	0,50	0,50
34	Programación interna de plataforma web	0%	0	0,70	0,00	100%	16	0,50	8,00
35	Modelar Interfaz gráfica de plataforma web	50%	8	0,70	5,60	0%	0	0,50	0,00
36	Reunion N°7	100%	1	0,70	0,70	100%	1	0,50	0,50
37	Programación de la interfaz gráfica de plataforma web	0%	0	0,70	0,00	100%	7	0,50	3,50
38	Montar el servidor para la plataforma	0%	0	0,70	0,00	100%	5	0,50	2,50
39	Reunion N°8	100%	1	0,70	0,70	100%	1	0,50	0,50
40	Confección Informe Final de Laboratorio Fase II	0%	0	0,70	0,00	0%	0	0,50	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-
<b>TOTAL</b>			<b>114,00</b>		<b>79,80</b>		<b>124,00</b>		<b>62,00</b>

ID	Actividad	Nicolás Soto				Alvaro Cabedo			
		Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)	Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	1,00	1,00	100%	1	0,70	0,70
2	Planificacion de Actividades	56%	10	1,00	10,00	33%	6	0,70	4,20
3	Division de tareas e informacion a investigar	100%	2	1,00	2,00	0%	0	0,70	0,00
4	Investigacion para propuesta	40%	9,5	1,00	9,50	50%	12	0,70	8,40
5	Confeccion de la propuesta	13%	6	1,00	6,00	8%	4	0,70	2,80
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	1,00	1,00	100%	1	0,70	0,70
8	Instalacion de Ubuntu 14.04	0%	0	1,00	0,00	0%	0	0,70	0,00
9	Instalacion y configuracion del Software necesario	0%	0	1,00	0,00	0%	0	0,70	0,00
10	Construccion del cluster	0%	0	1,00	0,00	0%	0	0,70	0,00
11	Ejecutar programa de prueba en cluster	0%	0	1,00	0,00	0%	0	0,70	0,00
12	Investigacion de diferentes algoritmos para la gestión de imagen	75%	12	1,00	12,00	75%	12	0,70	8,40
13	Reunión N°3	100%	1	1,00	1,00	100%	1	0,70	0,70
14	Modelar solucion de ambiente secuencial	25%	6	1,00	6,00	42%	10	0,70	7,00
15	Programacion solucion de ambiente secuencial	75%	6	1,00	6,00	69%	5,5	0,70	3,85
16	Optimizar solucion de ambiente secuencial	75%	6	1,00	6,00	0%	0	0,70	0,00
17	Calcular metricas de solucion de ambiente secuencial	100%	4	1,00	4,00	100%	4	0,70	2,80
18	Documentacion y analisis de QA de ambiente secuencial	100%	4	1,00	4,00	100%	4	0,70	2,80
19	Confeccion Informe Avance N°1	25%	4	1,00	4,00	0%	0	0,70	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	1,00	1,00	100%	1	0,70	0,70
22	Modelar solucion de ambiente paralelo	15%	3	1,00	3,00	30%	6	0,70	4,20
23	Programacion solucion de ambiente paralelo	83%	10	1,00	10,00	92%	11	0,70	7,70
24	Optimizar solucion de ambiente paralelo	38%	6	1,00	6,00	0%	0	0,70	0,00
25	Confeccion Informe Avance N°2	17%	4	1,00	4,00	0%	0	0,70	0,00

26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	1,00	1,00	100%	1	0,70	0,70
28	Calcular métricas de solución de ambiente paralelo	57%	4	1,00	4,00	57%	4	0,70	2,80
29	Documentación y análisis de QA de ambiente paralelo	100%	4	1,00	4,00	100%	4	0,70	2,80
30	Comparar métricas algoritmo secuencial versus paralelo	50%	2	1,00	2,00	0%	0	0,70	0,00
31	Modelar plataforma web	25%	4	1,00	4,00	50%	8	0,70	5,60
32	Integración de la aplicación paralela a la plataforma web	25%	4	1,00	4,00	13%	2	0,70	1,40
33	Reunion N°6	100%	1	1,00	1,00	100%	1	0,70	0,70
34	Programación interna de plataforma web	13%	2	1,00	2,00	0%	0	0,70	0,00
35	Modelar Interfaz gráfica de plataforma web	13%	2	1,00	2,00	50%	8	0,70	5,60
36	Reunion N°7	100%	1	1,00	1,00	100%	1	0,70	0,70
37	Programación de la interfaz gráfica de plataforma web	29%	2	1,00	2,00	0%	0	0,70	0,00
38	Montar el servidor para la plataforma	0%	0	1,00	0,00	0%	0	0,70	0,00
39	Reunion N°8	100%	1	1,00	1,00	100%	1	0,70	0,70
40	Confección Informe Final de Laboratorio Fase II	17%	4	1,00	4,00	0%	0	0,70	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-
<b>TOTAL</b>			<b>128,50</b>		<b>128,50</b>		<b>108,50</b>		<b>75,95</b>



ID	Actividad	Dominique Huenteleto				Boris Vásquez			
		Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)	Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	0,50	0,50	100%	1	0,50	0,50
2	Planificacion de Actividades	22%	4	0,50	2,00	22%	4	0,50	2,00
3	Division de tareas e informacion a investigar	0%	0	0,50	0,00	0%	0	0,50	0,00
4	Investigacion para propuesta	35%	8,5	0,50	4,25	13%	3	0,50	1,50
5	Confeccion de la propuesta	8%	4	0,50	2,00	4%	2	0,50	1,00
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	0,50	0,50	100%	1	0,50	0,50
8	Instalacion de Ubuntu 14.04	100%	4	0,50	2,00	100%	4	0,50	2,00
9	Instalacion y configuracion del Software necesario	100%	4	0,50	2,00	100%	4	0,50	2,00
10	Construccion del cluster	100%	4	0,50	2,00	100%	4	0,50	2,00
11	Ejecutar programa de prueba en cluster	100%	4	0,50	2,00	100%	4	0,50	2,00
12	Investigacion de diferentes algoritmos para la gestión de imagen	75%	12	0,50	6,00	75%	12	0,50	6,00
13	Reunión N°3	100%	1	0,50	0,50	100%	1	0,50	0,50
14	Modelar solucion de ambiente secuencial	0%	0	0,50	0,00	0%	0	0,50	0,00
15	Programacion solucion de ambiente secuencial	150%	12	0,50	6,00	150%	12	0,50	6,00
16	Optimizar solucion de ambiente secuencial	0%	0	0,50	0,00	0%	0	0,50	0,00
17	Calcular metricas de solucion de ambiente secuencial	0%	0	0,50	0,00	0%	0	0,50	0,00
18	Documentacion y analisis de QA de ambiente secuencial	0%	0	0,50	0,00	0%	0	0,50	0,00
19	Confeccion Informe Avance N°1	0%	0	0,50	0,00	0%	0	0,50	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	0,50	0,50	100%	1	0,50	0,50
22	Modelar solucion de ambiente paralelo	0%	0	0,50	0,00	0%	0	0,50	0,00
23	Programacion solucion de ambiente paralelo	75%	9	0,50	4,50	83%	10	0,50	5,00
24	Optimizar solucion de ambiente paralelo	0%	0	0,50	0,00	0%	0	0,50	0,00
25	Confeccion Informe Avance N°2	0%	0	0,50	0,00	0%	0	0,50	0,00

26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	0,50	0,50	100%	1	0,50	0,50
28	Calcular métricas de solución de ambiente paralelo	0%	0	0,50	0,00	0%	0	0,50	0,00
29	Documentación y análisis de QA de ambiente paralelo	0%	0	0,50	0,00	0%	0	0,50	0,00
30	Comparar métricas algoritmo secuencial versus paralelo	0%	0	0,50	0,00	0%	0	0,50	0,00
31	Modelar plataforma web	0%	0	0,50	0,00	0%	0	0,50	0,00
32	Integración de la aplicación paralela a la plataforma web	13%	2	0,50	1,00	13%	2	0,50	1,00
33	Reunion N°6	100%	1	0,50	0,50	100%	1	0,50	0,50
34	Programación interna de plataforma web	100%	16	0,50	8,00	100%	16	0,50	8,00
35	Modelar Interfaz gráfica de plataforma web	0%	0	0,50	0,00	0%	0	0,50	0,00
36	Reunion N°7	100%	1	0,50	0,50	100%	1	0,50	0,50
37	Programación de la interfaz gráfica de plataforma web	100%	7	0,50	3,50	100%	7	0,50	3,50
38	Montar el servidor para la plataforma	100%	5	0,50	2,50	100%	5	0,50	2,50
39	Reunion N°8	100%	1	0,50	0,50	100%	1	0,50	0,50
40	Confección Informe Final de Laboratorio Fase II	0%	0	0,50	0,00	0%	0	0,50	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-
<b>TOTAL</b>			<b>103,50</b>		<b>51,75</b>		<b>97,00</b>		<b>48,50</b>

ID	Actividad	Gladys Cerro				Felipe Abarca			
		Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)	Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	1,00	1,00	100%	1	0,50	0,50
2	Planificacion de Actividades	0%	0	1,00	0,00	0%	0	0,50	0,00
3	Division de tareas e informacion a investigar	50%	1	1,00	1,00	0%	0	0,50	0,00
4	Investigacion para propuesta	33%	8	1,00	8,00	33%	8	0,50	4,00
5	Confeccion de la propuesta	2%	1	1,00	1,00	2%	1	0,50	0,50
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	1,00	1,00	100%	1	0,50	0,50
8	Instalacion de Ubuntu 14.04	0%	0	1,00	0,00	100%	4	0,50	2,00
9	Instalacion y configuracion del Software necesario	0%	0	1,00	0,00	100%	4	0,50	2,00
10	Construccion del cluster	0%	0	1,00	0,00	100%	4	0,50	2,00
11	Ejecutar programa de prueba en cluster	0%	0	1,00	0,00	100%	4	0,50	2,00
12	Investigacion de diferentes algoritmos para la gestión de imagen	50%	8	1,00	8,00	50%	8	0,50	4,00
13	Reunión N°3	100%	1	1,00	1,00	100%	1	0,50	0,50
14	Modelar solucion de ambiente secuencial	17%	4	1,00	4,00	0%	0	0,50	0,00
15	Programacion solucion de ambiente secuencial	50%	4	1,00	4,00	100%	8	0,50	4,00
16	Optimizar solucion de ambiente secuencial	50%	4	1,00	4,00	0%	0	0,50	0,00
17	Calcular metricas de solucion de ambiente secuencial	100%	4	1,00	4,00	0%	0	0,50	0,00
18	Documentacion y analisis de QA de ambiente secuencial	100%	4	1,00	4,00	0%	0	0,50	0,00
19	Confeccion Informe Avance N°1	6%	1	1,00	1,00	0%	0	0,50	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	1,00	1,00	100%	1	0,50	0,50
22	Modelar solucion de ambiente paralelo	20%	4	1,00	4,00	0%	0	0,50	0,00
23	Programacion solucion de ambiente paralelo	17%	2	1,00	2,00	100%	12	0,50	6,00
24	Optimizar solucion de ambiente paralelo	25%	4	1,00	4,00	0%	0	0,50	0,00
25	Confeccion Informe Avance N°2	4%	1	1,00	1,00	0%	0	0,50	0,00

26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	1,00	1,00	100%	1	0,50	0,50
28	Calcular métricas de solución de ambiente paralelo	57%	4	1,00	4,00	0%	0	0,50	0,00
29	Documentación y análisis de QA de ambiente paralelo	100%	4	1,00	4,00	0%	0	0,50	0,00
30	Comparar métricas algoritmo secuencial versus paralelo	25%	1	1,00	1,00	0%	0	0,50	0,00
31	Modelar plataforma web	25%	4	1,00	4,00	0%	0	0,50	0,00
32	Integración de la aplicación paralela a la plataforma web	13%	2	1,00	2,00	6%	1	0,50	0,50
33	Reunion N°6	100%	1	1,00	1,00	100%	1	0,50	0,50
34	Programación interna de plataforma web	13%	2	1,00	2,00	50%	8	0,50	4,00
35	Modelar Interfaz gráfica de plataforma web	13%	2	1,00	2,00	0%	0	0,50	0,00
36	Reunion N°7	100%	1	1,00	1,00	100%	1	0,50	0,50
37	Programación de la interfaz gráfica de plataforma web	0%	0	1,00	0,00	43%	3	0,50	1,50
38	Montar el servidor para la plataforma	0%	0	1,00	0,00	0%	0	0,50	0,00
39	Reunion N°8	100%	1	1,00	1,00	100%	1	0,50	0,50
40	Confección Informe Final de Laboratorio Fase II	4%	1	1,00	1,00	0%	0	0,50	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-
<b>TOTAL</b>			<b>78,00</b>		<b>78,00</b>		<b>73,00</b>		<b>36,50</b>

ID	Actividad	David Martínez				Jonathan Acuña			
		Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)	Dedicacion (%)	Esfuerzo (HH)	Valor Hora (UF)	Costo (UF)
1	Reunion N°1	100%	1	0,50	0,50	100%	1	0,70	0,70
2	Planificacion de Actividades	0%	0	0,50	0,00	0%	0	0,70	0,00
3	Division de tareas e informacion a investigar	0%	0	0,50	0,00	0%	0	0,70	0,00
4	Investigacion para propuesta	33%	8	0,50	4,00	33%	8	0,70	5,60
5	Confeccion de la propuesta	2%	1	0,50	0,50	2%	1	0,70	0,70
6	Entrega de Informe de Propuesta	-	-	-	-	-	-	-	-
7	Reunion N°2	100%	1	0,50	0,50	100%	1	0,70	0,70
8	Instalacion de Ubuntu 14.04	100%	4	0,50	2,00	0%	0	0,70	0,00
9	Instalacion y configuracion del Software necesario	100%	4	0,50	2,00	0%	0	0,70	0,00
10	Construccion del cluster	100%	4	0,50	2,00	0%	0	0,70	0,00
11	Ejecutar programa de prueba en cluster	100%	4	0,50	2,00	0%	0	0,70	0,00
12	Investigacion de diferentes algoritmos para la gestión de imagen	50%	8	0,50	4,00	50%	8	0,70	5,60
13	Reunión N°3	100%	1	0,50	0,50	100%	1	0,70	0,70
14	Modelar solucion de ambiente secuencial	0%	0	0,50	0,00	33%	8	0,70	5,60
15	Programacion solucion de ambiente secuencial	100%	8	0,50	4,00	25%	2	0,70	1,40
16	Optimizar solucion de ambiente secuencial	0%	0	0,50	0,00	0%	0	0,70	0,00
17	Calcular metricas de solucion de ambiente secuencial	0%	0	0,50	0,00	100%	4	0,70	2,80
18	Documentacion y analisis de QA de ambiente secuencial	0%	0	0,50	0,00	100%	4	0,70	2,80
19	Confeccion Informe Avance N°1	0%	0	0,50	0,00	0%	0	0,70	0,00
20	Entrega Informe Avance N°1	-	-	-	-	-	-	-	-
21	Reunión N°4	100%	1	0,50	0,50	100%	1	0,70	0,70
22	Modelar solucion de ambiente paralelo	0%	0	0,50	0,00	40%	8	0,70	5,60
23	Programacion solucion de ambiente paralelo	100%	12	0,50	6,00	0%	0	0,70	0,00
24	Optimizar solucion de ambiente paralelo	0%	0	0,50	0,00	0%	0	0,70	0,00
25	Confeccion Informe Avance N°2	0%	0	0,50	0,00	0%	0	0,70	0,00

26	Entrega Informe Avance N°2	-	-	-	-	-	-	-	-
27	Reunion N°5	100%	1	0,50	0,50	100%	1	0,70	0,70
28	Calcular métricas de solución de ambiente paralelo	0%	0	0,50	0,00	57%	4	0,70	2,80
29	Documentación y análisis de QA de ambiente paralelo	0%	0	0,50	0,00	100%	4	0,70	2,80
30	Comparar métricas algoritmo secuencial versus paralelo	0%	0	0,50	0,00	0%	0	0,70	0,00
31	Modelar plataforma web	0%	0	0,50	0,00	50%	8	0,70	5,60
32	Integración de la aplicación paralela a la plataforma web	6%	1	0,50	0,50	6%	1	0,70	0,70
33	Reunion N°6	100%	1	0,50	0,50	100%	1	0,70	0,70
34	Programación interna de plataforma web	50%	8	0,50	4,00	0%	0	0,70	0,00
35	Modelar Interfaz gráfica de plataforma web	0%	0	0,50	0,00	50%	8	0,70	5,60
36	Reunion N°7	100%	1	0,50	0,50	100%	1	0,70	0,70
37	Programación de la interfaz gráfica de plataforma web	43%	3	0,50	1,50	0%	0	0,70	0,00
38	Montar el servidor para la plataforma	0%	0	0,50	0,00	0%	0	0,70	0,00
39	Reunion N°8	100%	1	0,50	0,50	100%	1	0,70	0,70
40	Confección Informe Final de Laboratorio Fase II	0%	0	0,50	0,00	0%	0	0,70	0,00
41	Entrega Informe Final de Laboratorio Fase II	-	-	-	-	-	-	-	-
<b>TOTAL</b>			<b>73,00</b>		<b>36,50</b>		<b>76,00</b>		<b>53,20</b>

ID	Actividad	TOTAL				
		Esfuerzo (HH)	Esfuerzo Acumulado		Costo (UF)	Costo Acumulado
1	Reunion N°1	16	16	16	13,20	13,20
2	Planificacion de Actividades	59	75	32	63,80	77,00
3	Division de tareas e informacion a investigar	9	84	5	8,50	85,50
4	Investigacion para propuesta	122	206	128	107,75	193,25
5	Confeccion de la propuesta	84	290	68	94,30	287,55
6	Entrega de Informe de Propuesta	0	290	0	0,00	287,55
7	Reunion N°2	16	306	16	12,70	300,25
8	Instalacion de Ubuntu 14.04	32	338	32	24,10	324,35
9	Instalacion y configuracion del Software necesario	34	372	32	25,50	349,85
10	Construccion del cluster	30	402	28	19,90	369,75
11	Ejecutar programa de prueba en cluster	28	430	28	18,20	387,95
12	Investigacion de diferentes algoritmos para la gestión de imagen	124	554	108	91,60	479,55
13	Reunión N°3	16	570	16	13,20	492,75
14	Modelar solucion de ambiente secuencial	88	658	80	91,00	583,75
15	Programacion solucion de ambiente secuencial	119,5	777,5	77	82,05	665,80
16	Optimizar solucion de ambiente secuencial	36	813,5	22	33,90	699,70
17	Calcular metricas de solucion de ambiente secuencial	34	847,5	26	26,90	726,60
18	Documentacion y analisis de QA de ambiente secuencial	40	887,5	28	32,20	758,80
19	Confeccion Informe Avance N°1	48	935,5	34	55,20	814,00
20	Entrega Informe Avance N°1	0	935,5	0	0,00	814,00
21	Reunión N°4	16	951,5	16	13,20	827,20
22	Modelar solucion de ambiente paralelo	65	1016,5	76	71,80	899,00
23	Programacion solucion de ambiente paralelo	161	1177,5	93	109,00	1008,00
24	Optimizar solucion de ambiente paralelo	67	1244,5	37	62,40	1070,40

25	Confeccion Informe Avance N°2	68	1312,5	57	83,00	1153,40
26	Entrega Informe Avance N°2	0	1312,5	0	0,00	1153,40
27	Reunion N°5	16	1328,5	16	13,20	1166,60
28	Calcular metricas de solucion de ambiente paralelo	29	1357,5	29	26,70	1193,30
29	Documentacion y analisis de QA de ambiente paralelo	31	1388,5	31	29,40	1222,70
30	Comparar métricas algoritmo secuencial versus paralelo	18	1406,5	17	22,00	1244,70
31	Modelar plataforma web	64	1470,5	64	64,00	1308,70
32	Integración de la aplicación paralela a la plataforma web	45	1515,5	50	43,30	1352,00
33	Reunion N°6	16	1531,5	16	13,20	1365,20
34	Programación interna de plataforma web	92	1623,5	108	53,60	1418,80
35	Modelar Interfaz gráfica de plataforma web	54	1677,5	54	52,40	1471,20
36	Reunion N°7	16	1693,5	16	13,20	1484,40
37	Programación de la interfaz gráfica de plataforma web	39	1732,5	49	22,40	1506,80
38	Montar el servidor para la plataforma	23	1755,5	33	13,80	1520,60
39	Reunion N°8	16	1771,5	16	13,20	1533,80
40	Confeccion Informe Final de Laboratorio Fase II	55	1826,5	52	71,60	1605,40
41	Entrega Informe Final de Laboratorio Fase II	0	1826,5	0	0,00	1605,40
<b>TOTAL</b>		1826,50			1605,40	

Fuente: Elaboración propia



## ANEXO J. Costo del personal

*Tabla 30, Costo del personal, asociado al proyecto*

Equipo de Trabajo	Esfuerzo (HH)	UF*Hora	Costo Total
Felipe Canales	233	1,40	326,20
Camila Vera	105	1,30	136,50
Nelson Cariqueo	112	1,20	134,40
Diego García	133	1,20	159,60
Elias Caceres	129	1,00	129,00
Nicolas Soto	129	1,00	129,00
Gladys Cerro	78	1,00	129,00
Katherine Saez	114	0,70	79,80
Alvaro Cabedo	109	0,70	75,95
Jonathan Acuña	76	0,70	53,20
Gabriel Sanhueza	138	0,50	69,00
Mario Urbina	124	0,50	62,00
Dominique Huenteleto	104	0,50	51,75
Boris Vasquez	97	0,50	48,50
Felipe Abarca	73	0,50	36,50
David Martinez	73	0,50	36,50
<b>Presupuesto Proyecto (Horas)</b>	<b>1826</b>	<b>-</b>	<b>-</b>
<b>Costo Capital Humano (UF)</b>	<b>-</b>	<b>-</b>	<b>1656,90</b>

Fuente: Elaboración propia

## ANEXO K. Totales de esfuerzo y Costo

Tabla 31, Totales acumulados de esfuerzo y costo

ID	Actividad	TOTAL							
		UF Total por Actividad	PV	EV	EV AC	AC	AC AC	Esfuerzo (HH)	Esfuerzo Acumulado
1	Reunion N°1	13,20	13,20	13,20	13,20	13,20	13,20	16	16
2	Planificacion de Actividades	42,60	55,80	42,60	55,80	63,80	77,00	59	75
3	Division de tareas e informacion a investigar	5,80	61,60	5,80	61,60	8,50	85,50	9	84
4	Investigacion para propuesta	108,80	170,40	108,80	170,40	107,75	193,25	122	206
5	Confeccion de la propuesta	82,50	252,90	82,50	252,90	94,30	287,55	84	290
6	Entrega de Informe de Propuesta	0,00	252,90	0,00	252,90	0,00	287,55	0	290
7	Reunion N°2	13,20	266,10	13,20	266,10	12,70	300,25	16	306
8	Instalacion de Ubuntu 14.04	22,40	288,50	22,40	288,50	24,10	324,35	32	338
9	Instalacion y configuracion del Software necesario	22,40	310,90	22,40	310,90	25,50	349,85	34	372
10	Construccion del cluster	16,80	327,70	16,80	327,70	19,90	369,75	30	402
11	Ejecutar programa de prueba en cluster	16,80	344,50	16,80	344,50	18,20	387,95	28	430
12	Investigacion de diferentes algoritmos para la gestión de imagen	80,80	425,30	80,80	425,30	91,60	479,55	124	554
13	Reunión N°3	13,20	438,50	13,20	438,50	13,20	492,75	16	570
14	Modelar solucion de ambiente secuencial	86,40	524,90	86,40	524,90	91,00	583,75	88	658
15	Programacion solucion de ambiente secuencial	53,60	578,50	51,99	576,89	82,05	665,80	119,5	777,5
16	Optimizar solucion de ambiente secuencial	24,80	603,30	24,80	601,69	33,90	699,70	36	813,5
17	Calcular metricas de solucion de ambiente secuencial	22,90	626,20	22,90	624,59	26,90	726,60	34	847,5
18	Documentacion y analisis de QA de ambiente secuencial	25,50	651,70	25,50	650,09	32,20	758,80	40	887,5
19	Confeccion Informe Avance N°1	44,40	696,10	44,40	694,49	55,20	814,00	48	935,5
20	Entrega Informe Avance N°1	0,00	696,10	0,00	694,49	0,00	814,00	0	935,5
21	Reunión N°4	13,20	709,30	13,20	707,69	13,20	827,20	16	951,5
22	Modelar solucion de ambiente paralelo	79,80	789,10	79,80	787,49	71,80	899,00	65	1016,5

23	Programacion solucion de ambiente paralelo	60,20	849,30	55,99	843,48	109,00	1008,00	161	1177,5
24	Optimizar solucion de ambiente paralelo	42,80	892,10	42,80	886,28	62,40	1070,40	67	1244,5
25	Confeccion Informe Avance N°2	74,60	966,70	74,60	960,88	83,00	1153,40	68	1312,5
26	Entrega Informe Avance N°2	0,00	966,70	0,00	960,88	0,00	1153,40	0	1312,5
27	Reunion N°5	13,20	979,90	13,20	974,08	13,20	1166,60	16	1328,5
28	Calcular metricas de solucion de ambiente paralelo	26,70	1006,60	26,70	1000,78	26,70	1193,30	29	1357,5
29	Documentacion y analisis de QA de ambiente paralelo	29,40	1036,00	29,40	1030,18	29,40	1222,70	31	1388,5
30	Comparar métricas algoritmo secuencial versus paralelo	21,00	1057,00	21,00	1051,18	22,00	1244,70	18	1406,5
31	Modelar plataforma web	64,00	1121,00	64,00	1115,18	64,00	1308,70	64	1470,5
32	Integración de la aplicación paralela a la plataforma web	47,00	1168,00	47,00	1162,18	43,30	1352,00	45	1515,5
33	Reunion N°6	13,20	1181,20	13,20	1175,38	13,20	1365,20	16	1531,5
34	Programación interna de plataforma web	61,60	1242,80	61,60	1236,98	53,60	1418,80	92	1623,5
35	Modelar Interfaz gráfica de plataforma web	52,40	1295,20	52,40	1289,38	52,40	1471,20	54	1677,5
36	Reunion N°7	13,20	1308,40	13,20	1302,58	13,20	1484,40	16	1693,5
37	Programación de la interfaz gráfica de plataforma web	28,40	1336,80	28,40	1330,98	22,40	1506,80	39	1732,5
38	Montar el servidor para la plataforma	18,80	1355,60	18,80	1349,78	13,80	1520,60	23	1755,5
39	Reunion N°8	13,20	1368,80	13,20	1362,98	13,20	1533,80	16	1771,5
40	Confeccion Informe Final de Laboratorio Fase II	68,60	1437,40	68,60	1431,58	71,60	1605,40	55	1826,5
41	Entrega Informe Final de Laboratorio Fase II	0,00	1437,40	0,00	1431,58	0,00	1605,40	0	1826,5
<b>TOTAL</b>		<b>1437,40</b>		<b>1431,58</b>		<b>1605,40</b>		<b>1826,5</b>	

Fuente: Elaboración propia

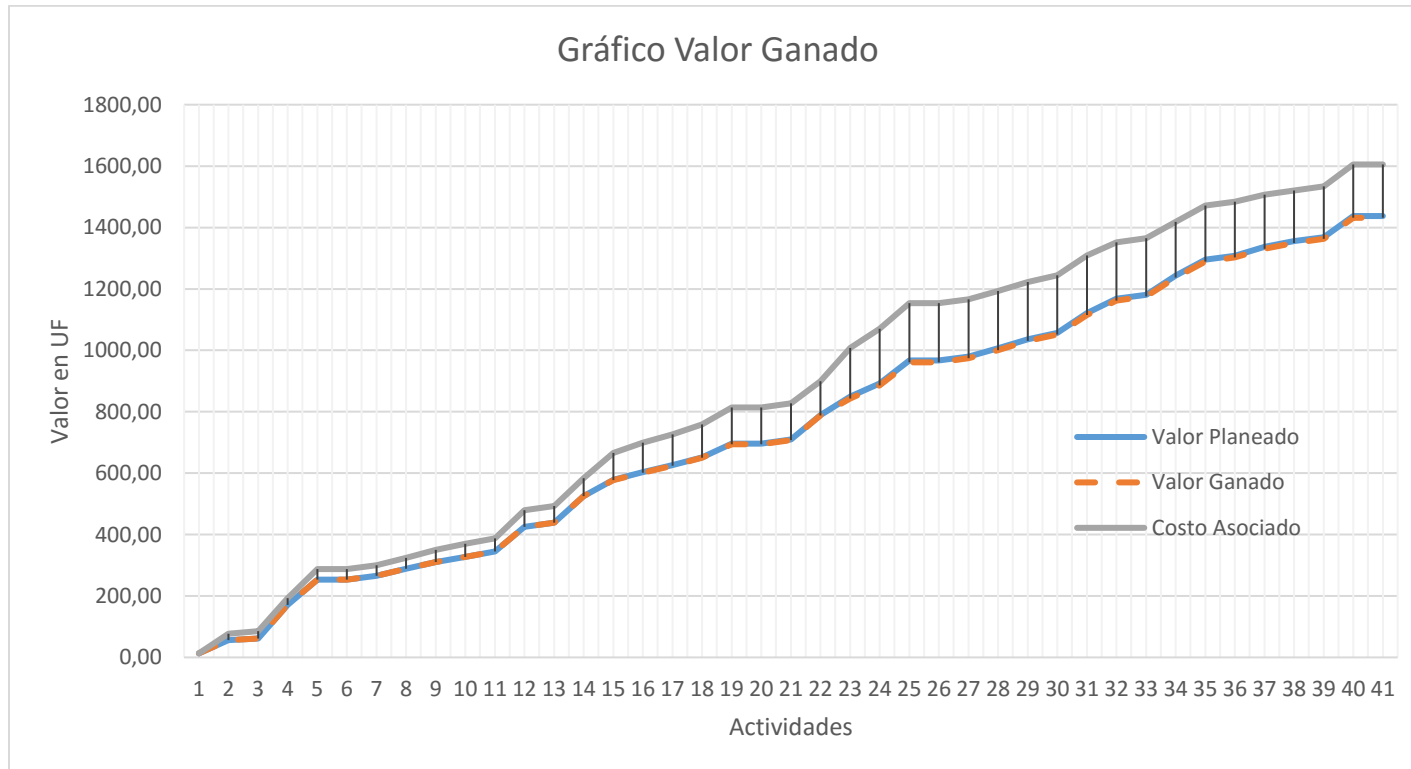


Ilustración 50, Gráfico de valor ganado y costo asociado al proyecto  
Fuente: Elaboración propia

## ANEXO L. Justificación Grupo 3

A la fecha 07 de julio del 2016 de acuerdo a la información observada anteriormente sobre el esfuerzo realizado como grupo 2 no se terminó de completar todos los algoritmos designados por el grupo líder, esto debido a las razones mencionadas a continuación:

- En primera instancia, se especifica que los algoritmos completados al cien por ciento (secuencial y paralelo) son: Laplaciano de la Gaussiana, Espejo y Luminosidad. Por otra parte, los algoritmos no realizados al cien por ciento son los de deformación, se logró implementar el algoritmo Deformación de Malla secuencial y la Deformación Bézier no fue posible realizarla.
- La razón principal que nos imposibilitó realizar los algoritmos de deformación es el grado de dificultad de éstos además de la poca información sobre las funcionalidades disponibles para éste tipo de efectos. Las librerías Opencv y PIL de Python, que son las que nos permitirían ésta tarea, no tienen funciones directas para este tipo de efecto. Por lo que al menos para Deformación de Malla se utilizó una manera más práctica de manipular el código para obtener la deformación de la imagen y para la Deformación Bézier no fue posible utilizar este método más práctico por lo que se concluye que las librerías Opencv y PIL de Python no nos fueron útiles a la hora de la implementación de dichos algoritmos.