

# Paquete de Python para el Procesamiento de Imágenes Usando Varios Modelos no Lineales

Carlos Toledo Silva

Tutor  
Damian Valdés Santiago

Universidad de La Habana

6 de diciembre de 2022

## Puntos claves:

- Bajo ciertas circunstancias el Procesamiento Lineal Clásico de Imágenes (CLIP) demuestra sus limitaciones.
- Alan V. Oppenheim introduce la teoría homomórfica.
- Michel Jourlin y Jean-Charles Pinoli proponen el primer modelo logarítmico para el procesamiento de imágenes definido desde un punto de vista físico.
- Vasile Patrascu y Vasile Buzuloiu proponen otro modelo logarítmico definido desde un punto de vista matemático.
- Constantin Vertan y colaboradores proponen un modelo pseudo-logarítmico.
- Se siguen desarrollando modelos no lineales, muchos inspirados en los anteriores.
- Con estos modelos se han desarrollado diversas aplicaciones: corrección de iluminación, mejora de contraste, mejora de imagen en color, ecualización de histogramas, mejora de rango dinámico, detección de bordes, etc.

# Introducción – Objetivos

## Objetivos generales:

- Diseño e implementación de un módulo de Python donde se encuentren los modelos no lineales para el procesamiento de imágenes abordados en este trabajo.
- Evaluación de los diferentes modelos en cuanto a las diferentes operaciones, tipos de imágenes y situaciones.

## Objetivos específicos:

- Revisar la literatura referente a los modelos no lineales para el procesamiento de imágenes.
- Implementación en Python de los diferentes modelos a partir de la bibliografía consultada.
- Creación de *datasets* con diferentes tipos de imágenes para evaluar los modelos.
- Selección de métricas para la evaluación de los modelos.
- Evaluación de los modelos.
- Arribar a conclusiones sobre la eficacia y utilidad de estos modelos.

# Estado del Arte – Antecedentes Matemáticos

## Definición

Sean  $(E, K, \oplus, \otimes)$  y  $(F, K, +, \cdot)$  dos estructuras algebraicas,  $K \subseteq \mathbb{R}$ , se denomina homomorfismo a la relación  $\varphi : E \rightarrow F$  que satisface:

$$\varphi(u \oplus v) = \varphi(u) + \varphi(v), \forall u, v \in E, \quad (1)$$

$$\varphi(\alpha \otimes u) = \alpha \cdot \varphi(u), \forall u \in E, \forall \alpha \in K. \quad (2)$$

La admisión de un inverso y la existencia de biyectividad indica la presencia de un isomorfismo.

En este trabajo:

- $E \subseteq \mathbb{R}$ , conjunto de valoración de las imágenes.
- $F \subseteq \mathbb{R}$ , conjunto de valoración de las estructuras vectoriales. Si  $u \in E$  entonces  $\varphi(u) \in F$
- $I(D, E)$  conjunto de imágenes definidas en  $D \subset \mathbb{R}^2$  y valoradas en  $E$ .
- $I(D, F)$  conjunto de las estructuras vectoriales definidas en  $D \subset \mathbb{R}^2$  y valoradas en  $F$ . Si  $f \in I(D, E)$  entonces  $\varphi(f) \in I(D, F)$ .

## Ejemplos:

- Si los valores de la imagen son intensidades, como cualquier plano en representación de color RGB, el conjunto  $E$  tiene la forma  $[0, M]$ .
- En el caso del espacio YUV, para los canales de diferencias de color, el conjunto  $E$  tiene una forma simétrica, como  $(-M/2, +M/2)$ .
- Una imagen en niveles de gris toma sus valores (niveles de gris) en la escala de grises  $[0, M]$ . Para imágenes de 8 bits,  $M = 256$  y los 256 niveles de gris están en la escala de números enteros  $[0, \dots, 255]$ .

# Estado del Arte – Antecedentes Matemáticos

Con respecto a la restricción de biyectividad y la existencia de  $\varphi^{-1}$ , las leyes de definición están determinadas por:

$$u \oplus v = \varphi^{-1}(\varphi(u) + \varphi(v)), \quad (3)$$

$$\alpha \otimes u = \varphi^{-1}(\alpha \cdot \varphi(u)). \quad (4)$$

Propiedad de cierre:

$$\forall u, v \in E, z = u \oplus v \Rightarrow z \in E, \quad (5)$$

$$\forall u \in E, \forall \alpha \in K, z = \alpha \otimes v \Rightarrow z \in E. \quad (6)$$

Estas propiedades se mantienen bajo la hipótesis de biyectividad asumida ya que:

$$z = \varphi^{-1}(\varphi(u) + \varphi(v)) \wedge \forall x \in F, \varphi^{-1}(x) \in E \Rightarrow z \in E. \quad (7)$$

$$z = \varphi^{-1}(\alpha \cdot \varphi(u)) \wedge \forall x \in F, \varphi^{-1}(x) \in E \Rightarrow z \in E. \quad (8)$$

## Definición

*Dadas las dos leyes operativas,  $\oplus$ ,  $\otimes$ , el conjunto vectorial  $E$  y el conjunto escalar  $K$ , la definición formal de espacio vectorial implica varias propiedades:*

- *La ley de la suma debe ser asociativa, conmutativa, debe tener elemento identidad y elemento inverso*
- *La distributividad debe mantenerse para la multiplicación escalar sobre la suma de vectores y para la multiplicación escalar en el espacio de los escalares.*
- *La multiplicación escalar debe tener elemento identidad y ser compatible con la multiplicación en el espacio de los escalares.*

Las propiedades conmutativas, asociativas y distributivas se pueden comprobar bajo la hipótesis de biyectividad y la admisión de inverso asumidas.

# Estado del Arte – Antecedentes Matemáticos

Elemento identidad con respecto a la suma:

$$\forall u \in E, \exists u_0 : u \oplus u_0 = u | \varphi(u_0) = 0. \quad (9)$$

Elemento inverso con respecto a la suma:

$$\forall u \in E, \exists u^- \Rightarrow u \oplus u^- = u_0 | \varphi(u) + \varphi(u^-) = 0 \Rightarrow \varphi(u^-) = -\varphi(u). \quad (10)$$

Elemento identidad de la multiplicación escalar tiene que ser 1:

$$\forall u \in E - \{u_0\}, \exists \alpha_1 : \alpha_1 \otimes u = u | \alpha_1 = 1. \quad (11)$$

## Definición

Sea  $V$  un espacio vectorial real y  $K$ , el conjunto escalar. Un cono  $C$  en  $V$  es un subconjunto de  $V$  que satisface:

- 1 si  $u_1, u_2 \in C$  y  $\alpha, \beta \in K : \alpha, \beta \geq 0$ , entonces  $\alpha u_1 + \beta u_2 \in C$ .
- 2  $C \cap (-C) = 0$ .

# Estado del Arte – Modelo LIP

Nombre: Modelo Logarítmico Clásico para el Procesamiento de Imágenes.

Propuesto por: Michel Jourlin y Jean-Charles Pinoli.

Dominio:  $E = [0, M] \vee E = (-\infty, M)$ .

Funciones de cambio:  $v = M - u$ ,  $u = M - v$ .

Suma:  $v_1 \oplus v_2 = v_1 + v_2 - \frac{v_1 v_2}{M}$ .

Multiplicación por un escalar:  $\lambda \otimes v = M - M \left(1 - \frac{v}{M}\right)^\lambda$ .

Opuesto:  $\ominus v = -\frac{v}{1 - \frac{v}{M}}$ .

Resta:  $v_1 \ominus v_2 = \frac{v_1 - v_2}{1 - \frac{v_2}{M}}$ .

Función del isomorfismo:  $\varphi(v) = -M \ln \left(1 - \frac{v}{M}\right)$ .

Inversa de la función del isomorfismo:  $\varphi^{-1}(x) = M \left(1 - e^{-\frac{x}{M}}\right)$ .

Propiedades:

- La suma  $\oplus$  es asociativa, conmutativa, tiene un elemento identidad: 0 y cada elemento tiene un opuesto.
- La multiplicación escalar  $\otimes$  es asociativa, tiene elemento identidad: 1 y la distributividad se mantiene para la multiplicación escalar sobre  $\oplus$  y para la multiplicación escalar en el espacio de los escalares.

Si  $E = [0, M]$  entonces  $(E, \mathbb{R}^+, \oplus, \otimes)$  es un cono positivo.

Si  $E = (-\infty, M)$  entonces  $(E, \mathbb{R}, \oplus, \otimes)$  es un espacio vectorial.

# Estado del Arte – Modelo HLIP

Nombre: Modelo Logarítmico Homomórfico para el Procesamiento de Imágenes.

Propuesto por: Vasile Patrascu y Vasile Buzuloiu.

Dominio:  $E = (-1, 1)$ .

Funciones de cambio:  $v = \frac{2}{M} \left(u - \frac{M}{2}\right)$ ,  $u = \frac{M}{2}(v + 1)$ .

Suma:  $v_1 \oplus v_2 = \frac{v_1 + v_2}{1 + v_1 v_2}$ .

Multiplicación por un escalar:  $\lambda \otimes v = \frac{(1+v)^\lambda - (1-v)^\lambda}{(1+v)^\lambda + (1-v)^\lambda}$ ,  $\lambda \in \mathbb{R}$ .

Opuesto:  $\ominus v = -v$ .

Resta:  $v_1 \ominus v_2 = \frac{v_1 - v_2}{1 - v_1 v_2}$ .

Función del isomorfismo:  $\varphi(v) = \frac{1}{2} \ln \left( \frac{1+v}{1-v} \right)$ .

Inversa de la función del isomorfismo:  $\varphi^{-1}(x) = \frac{e^{2x}-1}{e^{2x}+1}$ .

Propiedades:

- La suma  $\oplus$  es asociativa, conmutativa, tiene un elemento identidad: 0 y cada elemento tiene un opuesto.
- La multiplicación escalar  $\otimes$  es asociativa, tiene elemento identidad: 1 y la distributividad se mantiene para la multiplicación escalar sobre  $\oplus$  y para la multiplicación escalar en el espacio de los escalares.

La estructura  $(E, \mathbb{R}, \oplus, \otimes)$  es un espacio vectorial.

# Estado del Arte – Modelo PSLIP

Nombre: Modelo Pseudo-logarítmico para el Procesamiento de Imágenes.

Propuesto por: Constantin Vertan y colaboradores.

Dominio:  $E = [0, 1]$ .

Funciones de cambio:  $v = \frac{u}{M}$ ,  $u = M \cdot v$ .

Suma:  $v_1 \oplus v_2 = \frac{v_1 + v_2 - 2v_1 v_2}{1 - v_1 v_2}$ .

Multiplicación por un escalar:  $\lambda \otimes v = \frac{\lambda v}{1 + (\lambda - 1)v}$ ,  $\lambda \in \mathbb{R}^+$ .

Resta:  $v_1 \ominus v_2 = \frac{v_1 - v_2}{1 + v_1 v_2 - 2v_2}$  :  $v_1 \geq v_2$ .

Función del isomorfismo:  $\varphi(v) = \frac{v}{1-v}$ .

Inversa de la función del isomorfismo:  $\varphi^{-1}(x) = \frac{x}{1+x}$ .

Propiedades:

- La suma  $\oplus$  es asociativa, conmutativa y tiene un elemento identidad: 0.
- La multiplicación escalar  $\otimes$  es asociativa, tiene elemento identidad: 1 y la distributividad se mantiene para la multiplicación escalar sobre  $\oplus$  y para la multiplicación escalar en el espacio de los escalares.

La estructura  $(E, \mathbb{R}^+, \oplus, \otimes)$  es un cono positivo. La extensión a un espacio vectorial se puede lograr mediante el uso de la función generadora definida en  $(-1, 1) \rightarrow (-\infty, +\infty) : \varphi(v) = \frac{v}{1-|v|}$ .

# Estado del Arte – Modelo PLIP

Nombre: Modelo Logarítmico Parametrizado para el Procesamiento de Imágenes.

Propuesto por: Karen Panetta y colaboradores.

Dominio:  $E = [0, M] \cup E = (-\infty, M)$ .

Funciones de cambio:  $v = \mu(M) - u$ ,  $u = \mu(M) - v$ .

Suma:  $v_1 \oplus v_2 = v_1 + v_2 - \frac{v_1 v_2}{\gamma(M)}$ .

Multiplicación por un escalar:  $c \otimes v = \gamma(M) - \gamma(M)(1 - \frac{v}{\gamma(M)})^c$ .

Opuesto:  $\ominus v = -\frac{v}{1 - \frac{v}{\gamma(M)}}$ .

Resta:  $v_1 \ominus v_2 = \frac{v_1 - v_2}{1 - \frac{v_2}{k(M)}}$ .

Función del isomorfismo:  $\varphi(v) = -\lambda(M) \ln^\beta(1 - \frac{v}{\lambda(M)})$ .

Inversa de la función del isomorfismo:  $\varphi^{-1}(x) = \lambda(M)(1 - (e^{-\frac{x}{\lambda(M)}})^{\frac{1}{\beta}})$ .

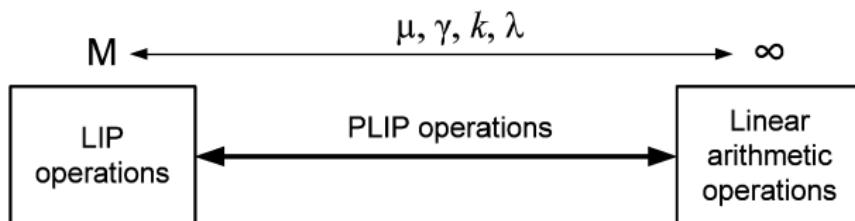


Figura: Operaciones en el modelo PLIP

## Propiedades del modelo PLIP:

- En el modelo PLIP, las operaciones aritméticas lineales se reemplazan con nuevas operaciones de la misma manera que en el modelo LIP tradicional.
- Las operaciones en el modelo PLIP son iguales a las del modelo tradicional LIP cuando  $\mu(M) = \gamma(M) = k(M) = \lambda(M) = M$  y  $\beta = 1$ .
- Las operaciones en el modelo PLIP se asemejan a las operaciones aritméticas lineales a medida que  $\gamma(M)$ ,  $k(M)$  y  $\lambda(M)$  se aproximan al infinito.
- Las operaciones PLIP pueden generar más casos entre los dos casos extremos del LIP y las operaciones aritméticas lineales cuando los parámetros  $\mu, \gamma, k$  y  $\lambda$  cambian dentro de  $[M, +\infty)$ .
- Las operaciones PLIP cumplen las leyes de asociatividad, conmutatividad, elemento identidad, elemento opuesto para la suma y las propiedades distributivas.

# Estado del Arte – Modelo SLIP

Nombre: Modelo Logarítmico Simétrico para el Procesamiento de Imágenes.

Propuesto por: Laurent Navarro y colaboradores.

Dominio:  $E = (-M, M)$ .

Suma:  $v_1 \oplus v_2 = M sgn(v_1 + v_2) \left[ 1 - \left( 1 - \frac{|v_1|}{M} \right)^{\gamma_1} \left( 1 - \frac{|v_2|}{M} \right)^{\gamma_2} \right],$

$\gamma_1 = \frac{sgn(v_1)}{sgn(v_1+v_2)}$ ,  $\gamma_2 = \frac{sgn(v_2)}{sgn(v_1+v_2)}$ .

Multiplicación por un escalar:  $\lambda \otimes v = M sgn(\lambda v) = \left[ 1 - \left( 1 - \frac{|v|}{M} \right)^{|\lambda|} \right].$

Opuesto:  $\ominus v = (-1) \otimes v$ .

Resta:  $v_1 \ominus v_2 = v_1 \oplus (-1) \otimes v_2$ .

Función del isomorfismo:  $\varphi(v) = -M sgn(v) \ln \left( 1 - \frac{|v|}{M} \right).$

Inversa de la función del isomorfismo:  $\varphi^{-1}(x) = M sgn(x) \left( 1 - e^{-\frac{|x|}{M}} \right).$

# Estado del Arte – Modelo SLIP

## Propiedades:

- La suma  $\oplus$  es asociativa, conmutativa, tiene un elemento identidad: 0 y cada elemento tiene un opuesto.
- La multiplicación escalar  $\otimes$  es asociativa, tiene elemento identidad: 1 y la distributividad se mantiene para la multiplicación escalar sobre  $\oplus$  y para la multiplicación escalar en el espacio de los escalares.

La estructura  $(E, \mathbb{R}, \oplus, \otimes)$  es un espacio vectorial.

## Relación Matemática entre los Modelos LIP y SLIP:

- Desde un punto de vista matemático, cuando  $v \in [0, M]$ , las dos funciones generadoras  $\varphi_{LIP}(v)$  y  $\varphi_{SLIP}(v)$  son exactamente las mismas (sin tener en cuenta la inversión de la escala de grises).
- El modelo  $SLIP(\odot)$  es el mismo que el modelo  $LIP(\square)$  para el rango de valores de píxeles  $[0, M]$
- Difieren entre sí en la forma en que tratan los valores negativos. Por ejemplo, cuando  $\lambda < 0$  y  $v > 0$ , tenemos  $\lambda \boxtimes v \in (-\infty, 0)$  y  $\lambda \otimes v \in (-M, 0)$ . Del mismo modo, cuando  $-\infty < v_1 < v_2 < M$  se tiene  $v_1 \boxminus v_2 \in (-\infty, 0)$  y cuando  $-M < v_1 < v_2 < M$ , se tiene  $v_1 \ominus v_2 \in (-M, 0)$ .

# Propuesta – Modelo PPSLIP

Nombre: Modelo Pseudo-logarítmico Parametrizado para el Procesamiento de Imágenes.

## Versión inicial: Parametrizar las funciones de cambio

Se parametrizan solo las funciones de cambio de la siguiente forma:

$v = \frac{u}{\delta(M)}$  y  $u = \delta(M)v$ , tal que  $\delta(M) \geq M$  y mantener el dominio y el resto de operaciones del modelo PSLIP.

Ventaja: Se parametriza el modelo para evitar la pérdida de información.

Desventaja: Las operaciones se parametrizan con el mismo parámetro.

## Versión final: Eliminar las funciones de cambio y hacerlos directamente

Dominio:  $E = [0, M]$ .

$$\text{Suma: } u_1 \oplus u_2 = \frac{u_1 + u_2 - \frac{2u_1 u_2}{\gamma(M)}}{1 - \frac{u_1 u_2}{\gamma(M)^2}}.$$

$$\text{Multiplicación por un escalar: } c \otimes u = \frac{cu}{1 + \frac{(c-1)u}{\gamma(M)}}.$$

$$\text{Resta: } u_1 \ominus u_2 = \frac{u_1 - u_2}{1 + \frac{u_1 u_2}{k(M)^2} - \frac{2u_2}{k(M)}} : u_1 \geq u_2.$$

$$\text{Función del isomorfismo: } \varphi(u) = \frac{u}{\lambda(M)-u}.$$

$$\text{Inversa de la función del isomorfismo: } \varphi^{-1}(x) = \lambda(M) \frac{x}{1+x}.$$

## Propiedades del modelo PPSLIP:

- En el modelo PPSLIP, las operaciones aritméticas lineales se reemplazan con nuevas operaciones de la misma manera que en el modelo PSLIP tradicional.
- Las operaciones en el modelo PPSLIP son iguales a las del modelo tradicional PSLIP cuando  $\gamma(M) = k(M) = \lambda(M) = M$ .
- Las operaciones en el modelo PLIP se asemejan a las operaciones aritméticas lineales a medida que  $\gamma(M)$ ,  $k(M)$  y  $\lambda(M)$  se aproximan al infinito.
- Las operaciones PPSLIP pueden generar más casos entre los dos casos extremos del PSLIP y las operaciones aritméticas lineales cuando los parámetros  $\gamma$ ,  $k$  y  $\lambda$  cambian dentro de  $[M, +\infty)$ .
- Las operaciones PPSLIP cumplen las leyes de asociatividad, comutatividad, elemento identidad y las propiedades distributivas.

# Propuesta – Módulo Implementado – Estructuras

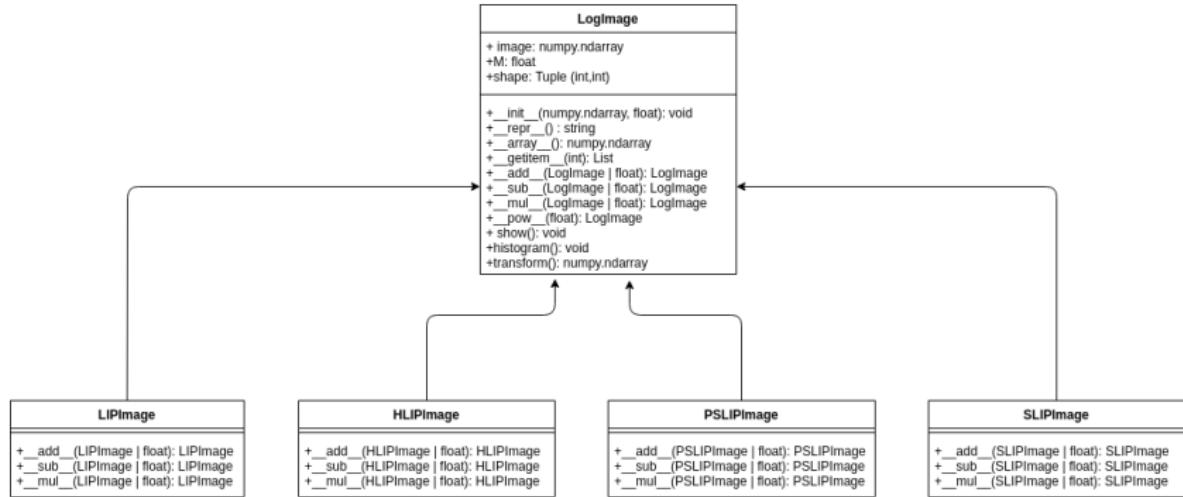


Figura: Diagrama de clases de las estructuras no lineales.

# Propuesta – Módulo Implementado – Espacios

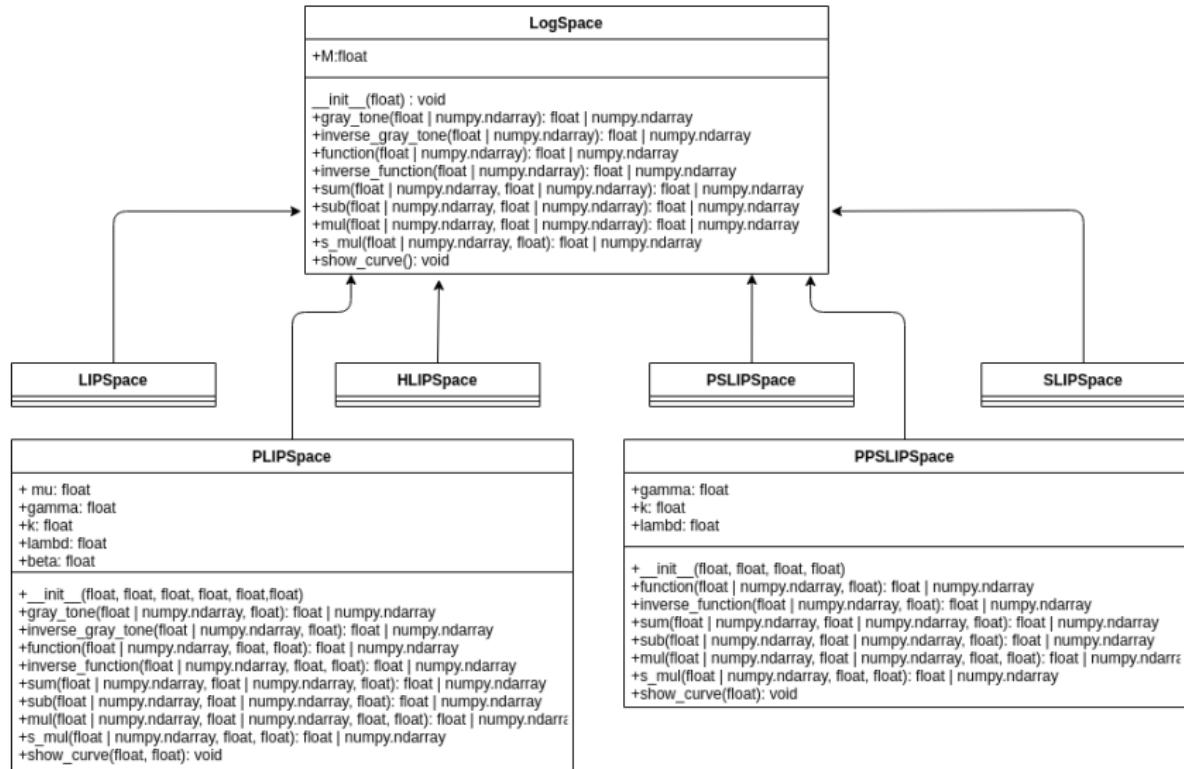


Figura: Diagrama de clases de los espacios no lineales.

La Medida de Mejora por Entropía se calcula dividiendo una imagen  $I$  en  $k_1 \times k_2$  bloques, obteniendo el máximo local  $I_{maxk,I}$  y el mínimo  $I_{mink,I}$  dentro de cada bloque individualmente, y luego procesándolos usando la siguiente ecuación:

$$EMEE_{\alpha, k_1, k_2} = \frac{1}{k_1 k_2} \sum_{l=1}^{k_1} \sum_{k=1}^{k_2} \alpha \left( \frac{I_{maxk,I}}{I_{mink,I}} \right)^\alpha \ln \left( \frac{I_{maxk,I}}{I_{mink,I}} \right), \quad (12)$$

donde  $\alpha$  es una constante que puede ayudar a seleccionar los parámetros. Se eligió  $\alpha = 1$  y el tamaño de bloque  $4 \times 4$ ,  $4 \times 5$ ,  $5 \times 4$  y  $5 \times 5$ , según las dimensiones de la imagen.

El mejor parámetro (óptimo) se obtiene si se cumple la siguiente condición:

$$EMEE_{optimal} = \max_{local}(EMEE(\alpha, \mu, \gamma, k, \lambda, \beta)) \quad (13)$$

Esta medida se utilizó para obtener las mejores combinaciones de parámetros en el modelo PLIP.

## Propuesta – Módulo Implementado – Métricas – $C_p$

En bibliografía consultada, denotando  $p_i = (x_i, y_i)$ , los pares de coordenadas que definen la posición espacial de un píxel en una imagen, el contraste absoluto entre dos píxeles distintos  $p_1, p_2 \in D$ , para una imagen,  $f \in I(D, E)$  se define por la relación:

$$C_A(p_1, p_2) = \frac{1}{d(p_1, p_2)} \cdot \frac{|f(p_1) - f(p_2)|}{1 - \frac{f(p_1) \cdot f(p_2)}{M^2}}, \quad (14)$$

Luego de la realización de una serie de experimentos, se decidió modificar la ecuación anterior, cambiando la definición a:

$$C_A(p_1, p_2) = \frac{|f(p_1) - f(p_2)|}{d(p_1, p_2)} \cdot \frac{256}{M} \quad (15)$$

El contraste para un píxel arbitrario  $p \in D$ , para una imagen  $f \in I(D, E)$ , se define por la media del contraste absoluto entre el píxel  $p$  y los píxeles  $(p_i)_{i=1, \dots, n}$  que pertenecen a una vecindad  $V$ :

$$C(p) = \frac{1}{n} \sum_{i=1}^n C_A(p, p_i). \quad (16)$$

Luego el valor de  $C_p$  de una imagen es el promedio de los valores de  $C(p)$  de los pixeles que la conforman.

## Detección de bordes

Para esto se pueden utilizar diferentes filtros como Sobel, Prewitt, Scharr, etc. En particular los tres mencionados anteriormente aparecen implementados en la librería `skimage.filters` de Python.

## Unsharp masking

*Unsharp Masking* es una técnica para mejorar la calidad de una imagen. Esta técnica consiste en determinar los bordes de una imagen y luego fusionar la imagen original con la imagen de bordes de dicha imagen.

## Transformación Afín

Considérense la transformación afín en el conjunto de imágenes  $I(D, E)$ , definida a continuación:  $\psi : I(D, E) \rightarrow I(D, E)$

$$\psi(f) = \lambda \otimes (f \oplus \tau), \quad (17)$$

donde  $\lambda \in \mathbb{R}$ ,  $\tau \in E$  y  $f \in I(D, E)$ .

Los parámetros  $(\lambda, \tau)$  se eligen de tal manera que se obtenga una nueva imagen muy cercana a una imagen con una distribución uniforme de los niveles de gris en el conjunto  $E = (a, b)$ .

Bajo este criterio la imagen mejorada debe tener su media  $\mu_u = \frac{a+b}{2}$  y su varianza  $\sigma_u^2 = \frac{(b-a)^2}{12}$ .

En estas condiciones para cualquier imagen  $f$  con la media  $\mu_f$  y la varianza  $\sigma_f^2$ , la transformación afín se convierte en:

$$\psi(f) = \frac{\sigma_u}{\sigma_f} \otimes (f \ominus \mu_f). \quad (18)$$

Importante: Este algoritmo solo se puede utilizar en espacios acotados.

# Detalles de Implementación – Estructura del Módulo

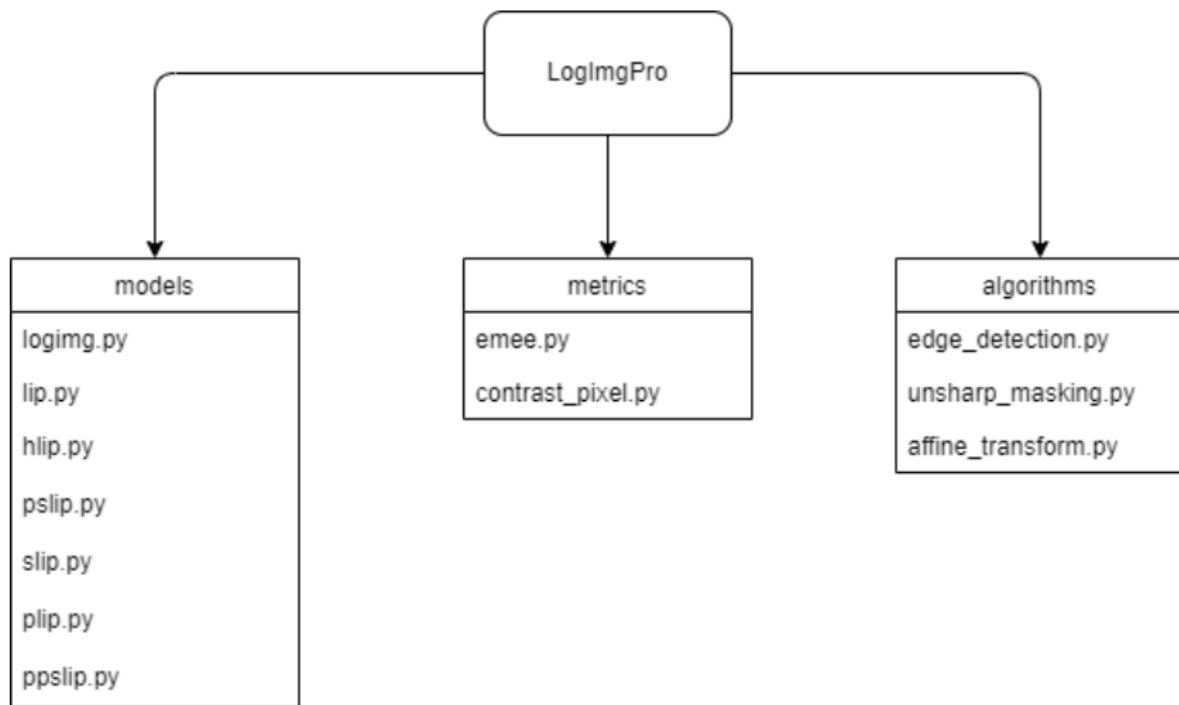


Figura: Diagrama de la estructura del módulo.

## Detalles de Implementación – Modelos

Las imágenes en niveles de gris se definen en el intervalo  $[0, M]$ .

Problema con los modelos LIP y HLIP pues, si  $u$  es un nivel de gris,  $v$  su tono de gris correspondiente y  $u = 0$  entonces:

- LIP:  $v = M - u = M - 0 = M$ . Contradicción  $E = [0, M]$ .
- HLIP:  $v = \frac{2}{M}(u - \frac{M}{2}) = \frac{2}{M}(0 - \frac{M}{2}) = \frac{2}{M}(-\frac{M}{2}) = -1$ . Contradicción  $E = (-1, 1)$

**Solución:** Antes de cambiar la imagen de niveles de gris a tonos de gris, esta primero se procesa de tal forma que el menor valor admisible sea un número cercano a cero, pero mayor que este. En esta implementación se decidió utilizar el número 0,0001.

Esto también se hizo para el modelo PLIP ya que este constituye una generalización del modelo LIP.

## Detalles de Implementación – Métricas – EMEE

`emee(numpy.ndarray, float, int, int, float): float`

**Entrada:** Una imagen, el valor de  $\alpha$ , las dimensiones de los bloques en los que se desea dividir la imagen y una constante  $c$ . Esta constante se suma tanto al numerador como al denominador de las fracciones que aparecen. De esta forma se redefinió el cálculo de EMEE como:

$$EMEE_{\alpha, k_1, k_2} = \frac{1}{k_1 k_2} \sum_{l=1}^{k_1} \sum_{k=1}^{k_2} \alpha \left( \frac{I_{maxk, l+c}}{I_{mink, l+c}} \right)^\alpha \ln \left( \frac{I_{maxk, l+c}}{I_{mink, l+c}} \right). \quad (19)$$

La constante utilizada fue 0,5.

**Salida:** El valor de EMEE de la imagen.

`find_min_max(numpy.ndarray, int, int, int, int): (float, float)`

**Entrada:** Una imagen y 4 enteros:  $n_1, n_2, m_1, m_2$ , tal que  $(n_1, m_1)$  son las coordenadas donde inicia el bloque y;  $n_2$  y  $m_2$  son las dimensiones horizontal y vertical del bloque, respectivamente.

**Salida:** Una tupla con el menor y el mayor valor de intensidad del bloque.

## Detalles de Implementación – Métricas – $C_p$

`abs_contrast_2_pixels(Tuple,Tuple,float): float`

**Entrada:** Dos tuplas de 3 elementos, tal que cada una representa un píxel (las coordenadas y el valor de intensidad del píxel), y el valor de  $M$ .

**Salida:** El contraste absoluto entre los 2 píxeles.

`contrast_pixel(Tuple, numpy.ndarray, float, int): float`

**Entrada:** Una tupla que representa un píxel, una imagen, el valor de  $M$  y el radio de la vecindad a considerarse.

**Salida:** El contraste del píxel en la imagen.

`contrast_img(numpy.ndarray, float, int): float`

**Entrada:** La imagen, el valor de  $M$  y el radio de la vecindad que se debe tener en cuenta alrededor de un píxel.

**Salida:** El contraste promedio de un píxel en la imagen, o sea  $C_p$ . El radio de la vecindad que se tomó para calcular el contraste de cada píxel fue de 2.

## D. de Implementación – Algoritmos – Detección de Bordes

`edge_detection(numpy.ndarray,filter): numpy.ndarray`

**Entrada:** Una imagen y un filtro.

**Salida:** El resultado de aplicarle el filtro a la imagen.

`space_edge_detection(numpy.ndarray,filter,LogSpace): numpy.ndarray`

**Entrada:** Los mismos parámetros que la anterior, además una instancia de tipo LogSpace.

**Salida:** El resultado de aplicarle el filtro a la imagen utilizando el modelo especificado.

`parameterized_space_edge_detection(numpy.ndarray,filter,LogSpace,*,float,float): numpy.ndarray`

**Entrada:** Los mismos parámetros que la función anterior, además de los valores de  $\lambda(M)$  y  $\mu(M)$ .

**Salida:** El resultado de aplicarle el filtro a la imagen utilizando el modelo y los parámetros especificados.

## D. de Implementación – Algoritmos – Unsharp Masking

`unsharp_masking(numpy.ndarray,filter,str): numpy.ndarray`

**Entrada:** Una imagen, un filtro y un *string*: ‘‘+’’ o ‘‘-’’.

**Salida:** El resultado de utilizar el filtro para obtener la imagen de bordes y fusionarla con la original utilizando la suma (‘‘+’’) o la resta (‘‘-’’).

`space_unsharp_masking(numpy.ndarray,filter,str,LogSpace): numpy.ndarray`

**Entrada:** Los mismos parámetros que la anterior, además una instancia de tipo LogSpace.

**Salida:** El resultado de utilizar el filtro para obtener la imagen de bordes y fusionarla con la original utilizando el modelo especificado.

`parameterized_space_unsharp_masking(numpy.ndarray,filter,str,LogSpace,*,float,float,float): numpy.ndarray`

**Entrada:** Los mismos parámetros que la función anterior, además de los valores de  $\gamma(M)$ ,  $\lambda(M)$  y  $\mu(M)$ .

**Salida:** El resultado de utilizar el filtro para obtener la imagen de bordes y fusionarla con la original utilizando el modelo y los parámetros dados.

```
space_affine_transform(numpy.ndarray, float, float, LogSpace):  
    numpy.ndarray
```

**Entrada:** Una imagen, los extremos del intervalo en el cual se va a realizar la transformación y una instancia de tipo LogSpace.

**Salida:** La transformación afín utilizando las operaciones aritméticas del modelo especificado.

# Experimentos – Curvas de los Modelos no Parametrizados

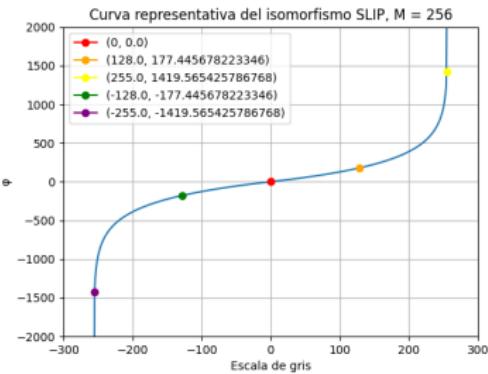
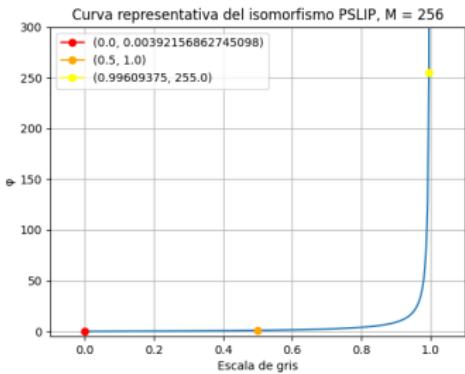
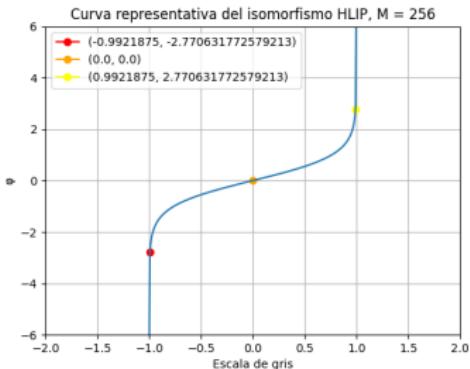
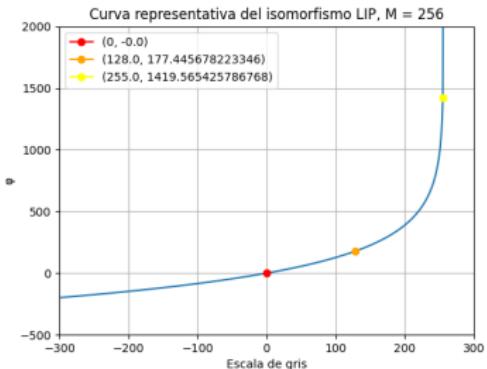
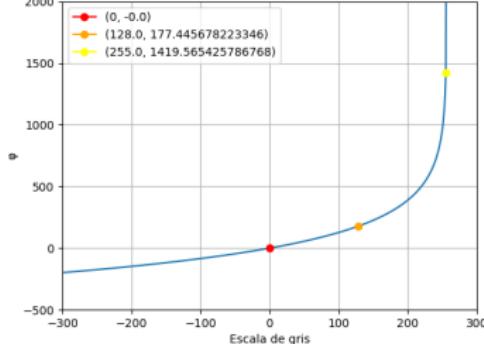


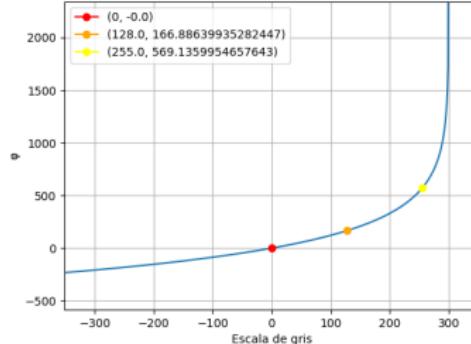
Figura: Curvas representativas de los modelos no parametrizados.

# Experimentos – Curvas del Modelo PLIP

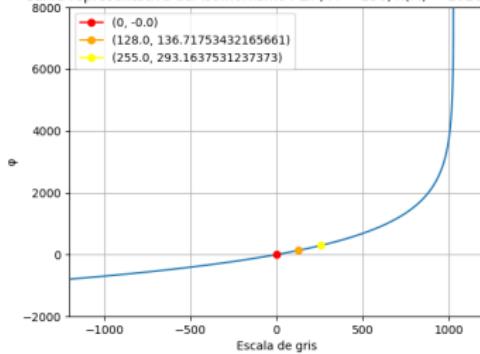
Curva representativa del isomorfismo PLIP,  $M = 256$ ,  $\lambda(M) = 256$ ,  $\beta = 1$



Curva representativa del isomorfismo PLIP,  $M = 256$ ,  $\lambda(M) = 300$ ,  $\beta = 1$



Curva representativa del isomorfismo PLIP,  $M = 256$ ,  $\lambda(M) = 1026$ ,  $\beta = 1$



Curva representativa del isomorfismo PLIP,  $M = 256$ ,  $\lambda(M) = 4100$ ,  $\beta = 1$

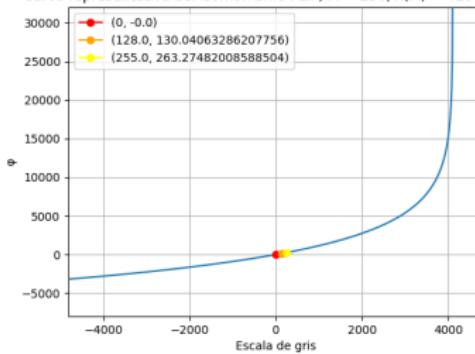


Figura: Curvas representativas del modelo PLIP con diferentes valores de  $\lambda(M)$ .

# Experimentos – Curvas del Modelo PPSLIP

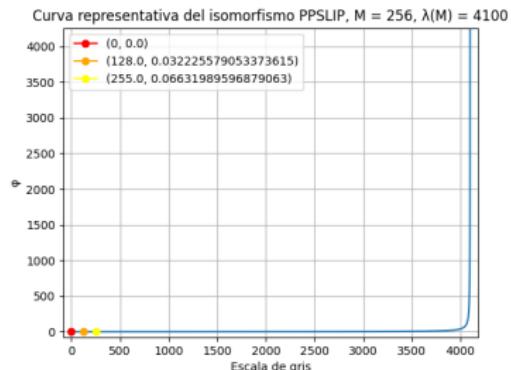
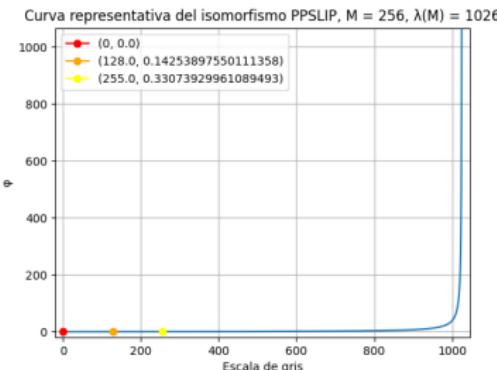
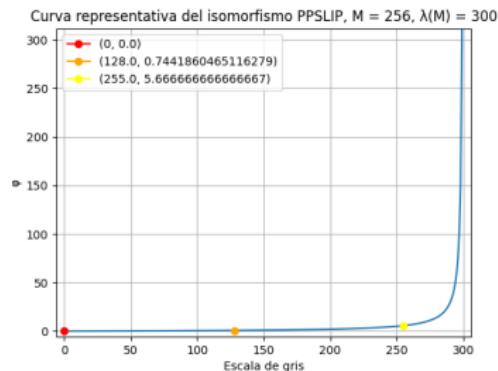
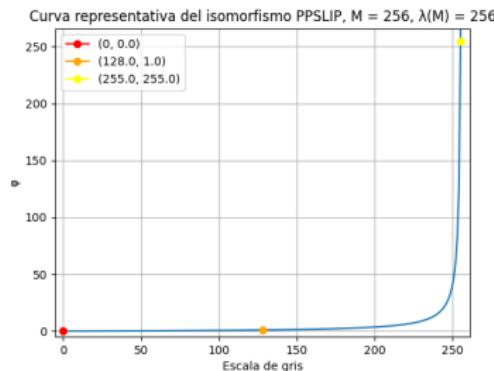


Figura: Curvas representativas del modelo PPSLIP, diferentes valores de  $\lambda(M)$ .

# Experimentos – Suma de Imágenes



(a) Playa



(b) Patineta



(c) Montaña



(d) Insecto

**Figura:** Imágenes para los experimentos de suma.

# Experimentos – Suma de Imágenes – Playa y Patineta



(a) Lineal  $C_p = 3,49$



(b) LIP  $C_p = 5,71$



(c) HLIP  $C_p = 4,96$



(d) PSLIP  $C_p = 1,39$



(e) SLIP  $C_p = 1,45$



(f) PLIP  $C_p = 5,71$



(g) PPSLIP  $C_p = 3,35$

# Experimentos – Suma de Imágenes – Montaña e Insecto



(h) Lineal  $C_p = 3,15$



(i) LIP  $C_p = 2,77$



(j) HLIP  $C_p = 4,53$



(k) PSLIP  $C_p = 3,47$



(l) SLIP  $C_p = 3,65$

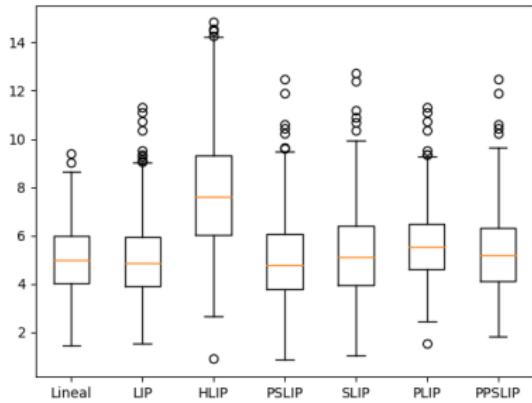


(m) PLIP  $C_p = 3,13$



(n) PPSLIP  $C_p = 3,47$

# Experimentos – Suma de Imágenes – Estadísticas



(a) Diagramas de caja para la medida  $C_p$ .

Modelo	Media	Mediana
Lineal	5,05	4,98
LIP	5,00	4,85
HLIP	7,73	7,59
PSLIP	4,95	4,78
SLIP	5,24	5,08
PLIP	5,60	5,54
PPSLIP	5,35	5,20

(b) Media y mediana del valor de  $C_p$ .

**Figura:** Análisis estadístico de la operación suma de los diferentes modelos utilizando imágenes naturales

# Experimentos – Imágenes Utilizadas



(a) Cámara  $C_p = 5,00$



(b) Tórax 1  $C_p = 2,26$

Figura: Imágenes para los experimentos de detección de bordes

# Experimentos – Detección de Bordes – Cámara



(a) Lineal  $C_p = 4,69$



(b) LIP  $C_p = 7,43$



(c) HLIP  $C_p = 8,05$



(d) PSLIP  $C_p = 9,41$



(e) SLIP  $C_p = 6,26$



(f) PLIP  $C_p = 7,43$



(g) PPSLIP  $C_p = 9,41$

**Figura:** Filtro de Scharr aplicado a la imagen Cámara con los diferentes modelos.

# Experimentos – Detección de Bordes – Tórax 1



(a) Lineal  $C_p = 2,89$



(b) LIP  $C_p = 3,66$



(c) HLIP  $C_p = 4,40$



(d) PSLIP  $C_p = 9,73$



(e) SLIP  $C_p = 6,85$



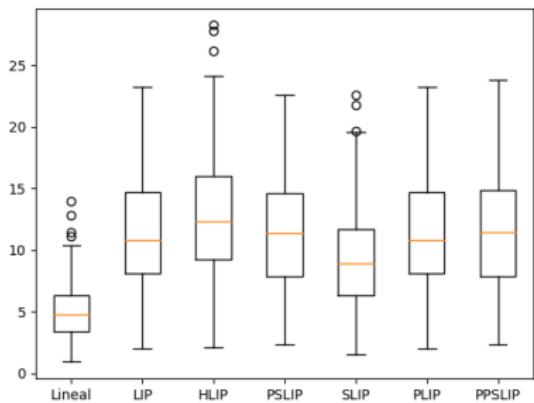
(f) PLIP  $C_p = 3,66$



(g) PPSLIP  $C_p = 9,73$

**Figura:** Filtro de Scharr aplicado a la imagen Tórax 1 con los diferentes modelos.

# Experimentos – Detección de Bordes – Estadísticas



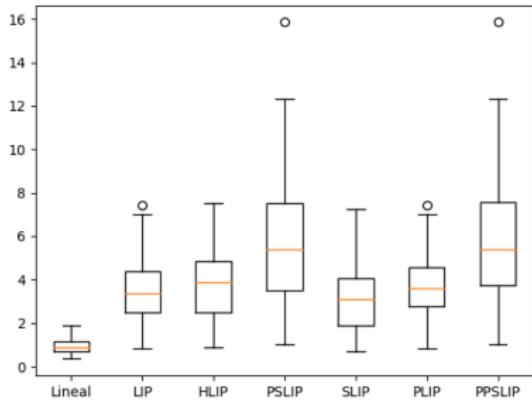
(a) Diagramas de caja para la medida  $C_p$ .

Modelo	Media	Mediana
Lineal	5,02	4,75
LIP	11,55	10,82
HLIP	12,81	12,36
PSLIP	11,57	11,38
SLIP	9,30	8,91
PLIP	11,59	10,82
PPSLIP	11,65	11,47

(b) Media y mediana del valor de  $C_p$ .

**Figura:** Análisis estadístico para la medida  $C_p$  utilizando los diferentes modelos para la detección de bordes en imágenes naturales.

# Experimentos – Detección de Bordes – Estadísticas



(a) Diagramas de caja para la medida  $C_p$ .

Modelo	Media	Mediana
Lineal	0,96	0,89
LIP	3,48	3,38
HLIP	3,69	3,85
PSLIP	5,55	5,38
SLIP	3,11	3,09
PLIP	3,68	3,60
PPSLIP	5,64	5,41

(b) Media y mediana del valor de  $C_p$ .

**Figura:** Análisis estadístico para la medida  $C_p$  utilizando los diferentes modelos para la detección de bordes en imágenes de radiografías de tórax.

# Experimentos – Unsharp Masking – Cámara



(a) Lineal  $C_p = 4,13$



(b) LIP  $C_p = 5,14$



(c) HLIP+  $C_p = 5,66$



(d) HLIP-  $C_p = 5,73$



(e) PSLIP  $C_p = 4,84$



(f) SLIP  $C_p = 5,16$



(g) PLIP  $C_p = 5,14$



(h) PPSLIP  $C_p = 5,80$

**Figura:** Unsharp masking aplicado a la imagen Cámara con los diferentes modelos.

# Experimentos – Unsharp Masking – Tórax 1



(a) Lineal  $C_p = 2,60$



(b) LIP  $C_p = 2,53$



(c) HLIP+  $C_p = 3,46$



(d) HLIP-  $C_p = 2,62$



(e) PSLIP  $C_p = 2,52$



(f) SLIP  $C_p = 2,59$



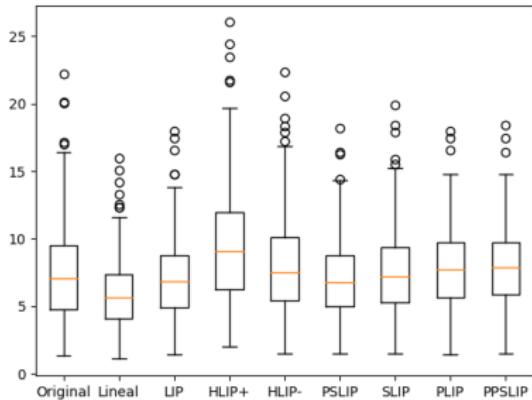
(g) PLIP  $C_p = 2,53$



(h) PPSLIP  $C_p = 4,97$

**Figura:** Unsharp masking aplicado a la imagen Tórax 1 con los diferentes modelos.

# Experimentos – Unsharp Masking – Estadísticas



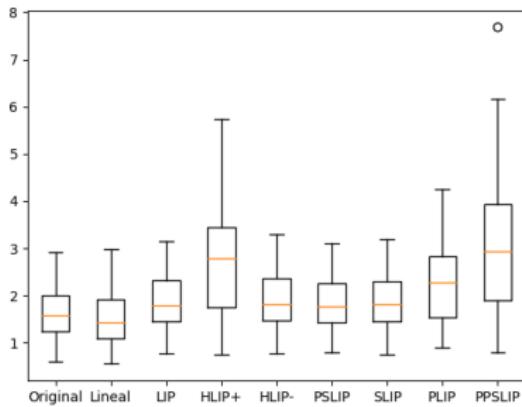
(a) Diagramas de caja para la medida  $C_p$ .

Modelo	Media	Mediana
Original	7,63	7,07
Lineal	6,04	5,66
LIP	7,24	6,81
HLIP+	9,67	9,07
HLIP-	8,23	7,52
PSLIP	7,10	6,73
SLIP	7,65	7,24
PLIP	7,95	7,72
PPSLIP	7,99	7,84

(b) Media y mediana del valor de  $C_p$ .

**Figura:** Análisis estadístico para la medida  $C_p$  utilizando los diferentes modelos para el algoritmo *unsharp masking* en imágenes naturales.

# Experimentos – Unsharp Masking – Estadísticas



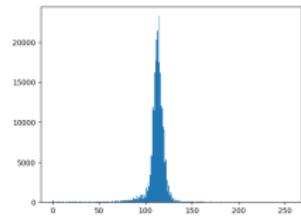
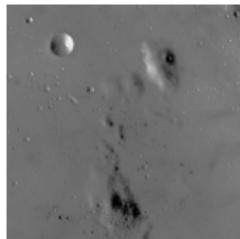
(a) Diagramas de caja para la medida  $C_p$ .

Modelo	Media	Mediana
Original	1,64	1,57
Lineal	1,49	1,42
LIP	1,88	1,78
HLIP+	2,71	2,78
HLIP-	1,91	1,82
PSLIP	1,84	1,78
SLIP	1,87	1,80
PLIP	2,21	2,27
PPSLIP	3,00	2,94

(b) Media y mediana del valor de  $C_p$ .

**Figura:** Análisis estadístico para la medida  $C_p$  utilizando los diferentes modelos para el algoritmo *unsharp masking* en imágenes de radiografías de tórax.

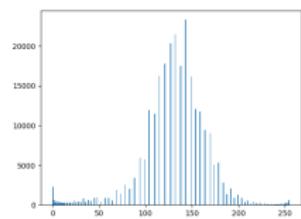
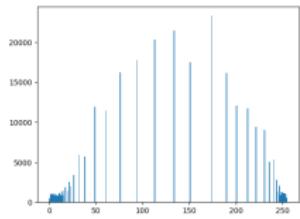
# Experimentos – Transformación Afín – Luna



(a) Original  
 $C_p = 1,30$



(c) Ecualizada  
 $C_p = 10,86$



(e) HLIP  
 $C_p = 5,35$



(g) SLIP  
 $C_p = 5,07$

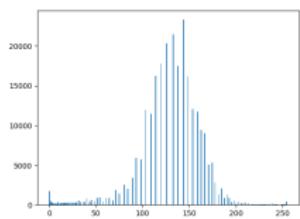
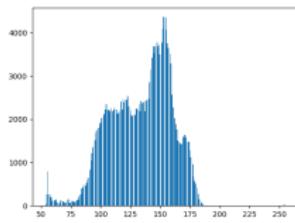


Figura: Diferentes técnicas de modificación del histograma de la imagen Luna.

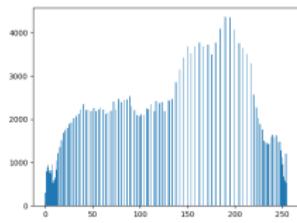
# Experimentos – Transformación Afín – Tórax 2



(a) Original  $C_p = 1,05$



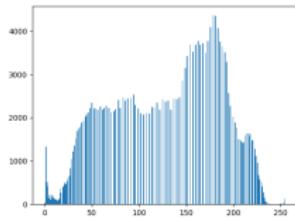
(c) Ecualizada  $C_p = 2,82$



(d) Ecualizada Histograma



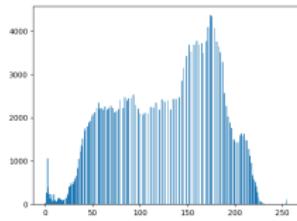
(e) HLIP  $C_p = 2,36$



(f) HLIP Histograma



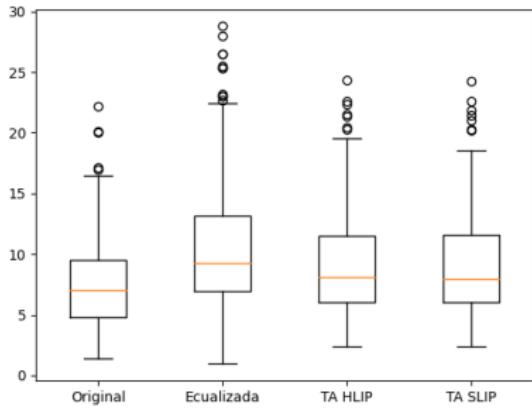
(g) SLIP  $C_p = 2,19$



(h) SLIP Histograma

**Figura:** Diferentes técnicas de modificación del histograma de la imagen Tórax 2.

# Experimentos – Transformación Afín – Estadísticas



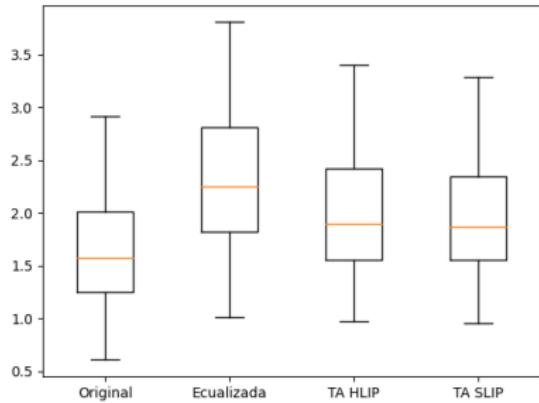
(a) Diagramas de caja para la medida  $C_p$ .

Modelo	Media	Mediana
Original	7,63	7,07
Ecualización	10,75	9,29
TA HLIP	9,20	8,07
TA SLIP	9,11	7,94

(b) Media y mediana del valor de  $C_p$ .

**Figura:** Análisis estadístico para la medida  $C_p$  utilizando los diferentes algoritmos para la modificación del histograma en imágenes naturales.

# Experimentos – Transformación Afín – Estadísticas



(a) Diagramas de caja para la medida  $C_p$ .

Modelo	Media	Mediana
Original	1,64	1,57
Ecualización	2,32	2,24
TA HLIP	2,01	1,89
TA SLIP	1,98	1,86

(b) Media y mediana del valor de  $C_p$ .

**Figura:** Análisis estadístico para la medida  $C_p$  utilizando los diferentes algoritmos para la modificación del histograma en imágenes de radiografías de tórax.

# Conclusiones

- 1 El modelo lineal muestra deficiencias en determinadas ocasiones, las cuales pueden ser resueltas por alguno de los modelos no lineales presentados.
- 2 El modelo LIP presenta sensibilidad hacia las tonalidades oscuras, mientras que el SLIP sin expansión presenta sensibilidad hacia las tonalidades claras.
- 3 El modelo HLIP demostró ser muy efectivo en los diferentes experimentos realizados dado su carácter simétrico.
- 4 El modelo PSLIP demostró ser muy efectivo para la detección de bordes, fundamentalmente en los niveles de mayor intensidad, quedando a deber en los de menor intensidad
- 5 La parametrización de los modelos demostró tener gran utilidad ya que permite cierta flexibilidad en favor de obtener mejores resultados.
- 6 El algoritmo de transformación afín con los dos modelos utilizados puede dar como resultado una imagen de mejor contraste que la imagen original y más natural que la imagen ecualizada.
- 7 La métrica  $C_p$  utilizada para evaluar los experimentos puede ser utilizada para la evaluación de distintos algoritmos para el procesamiento de imágenes.

# Recomendaciones

- 1 Utilización de los modelos no lineales abordados en sustitución del modelo lineal en los casos estudiados donde se evidencia un mejor resultado.
- 2 Continuar el estudio de estos modelos
- 3 Implementar algoritmos que combinen dos o más modelos aprovechando las ventajas que ofrece cada uno.
- 4 En los futuros estudios que se realicen, incluir también, como medida subjetiva, la opinión de diferentes usuarios con respecto a la calidad de los resultados obtenidos.
- 5 Implementación de un algoritmo que permita estimar para un modelo parametrizado el mejor o los mejores parámetros para la realización de una determinada operación

# Paquete de Python para el Procesamiento de Imágenes Usando Varios Modelos no Lineales

Carlos Toledo Silva

Tutor  
Damian Valdés Santiago

Universidad de La Habana

6 de diciembre de 2022