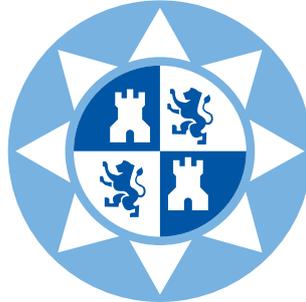


**UNIVERSIDAD POLITÉCNICA DE CARTAGENA**



**GRADO EN INGENIERÍA EN  
SISTEMAS DE TELECOMUNICACIÓN**

**Trabajo Fin de Grado**

**PROCESADO DE IMÁGENES RETINOGRÁFICAS  
MEDIANTE TÉCNICAS DE REGULARIZACIÓN  
POR VARIACIÓN TOTAL**

***Débora Meroño García***

*21 de diciembre de 2016*

Director: ***Dr. D. Rafael Verdú Monedero***



<b>1. Introducción</b>	<b>1</b>
1.1. Retina y retinografías . . . . .	1
<b>2. Método de Split Bregman</b>	<b>3</b>
2.1. Problemas de optimización con restricciones . . . . .	6
2.2. Iteración de Bregman . . . . .	6
2.2.1. Optimización con restricciones vía Iteración de Bregman . . . . .	7
2.2.2. Ventajas de la iteración de Bregman . . . . .	9
2.3. Split Bregman. Una mejor formulación para los problemas regularizados $l_1$ . . . . .	10
2.3.1. Minimización iterativa . . . . .	11
2.3.2. Implementación del algoritmo propuesto . . . . .	12
2.4. Aplicaciones . . . . .	13
2.4.1. Eliminación del ruido por Variación Total . . . . .	13
2.4.2. Compressed Sensing rápida para la reconstrucción de imágenes . . . . .	15
<b>3. Primeras pruebas</b>	<b>21</b>
3.1. Pruebas con ruido Gaussiano . . . . .	21
3.2. Efecto en las texturas . . . . .	38
<b>4. Pruebas en imágenes retinográficas</b>	<b>45</b>
<b>5. Conclusiones</b>	<b>51</b>
<b>Bibliografía</b>	<b>53</b>



# CAPÍTULO 1

---

## Introducción

---

En este trabajo se va a abordar la disminución del ruido presente en imágenes retinográficas con el fin de facilitar su uso en medicina y mejorar el posible diagnóstico de enfermedades a través de ellas. Por ello, será útil tener un concepto global de estos términos para así facilitar las posteriores explicaciones y conclusiones.

### **1.1. Retina y retinografías**

En primer lugar se darán unas pinceladas sobre las funciones y estructura de la retina.

La retina es una capa de tejido sensible a la luz situado en la superficie interior del ojo, en la parte posterior del globo ocular. Es similar a una tela en la que se proyectan las imágenes que pasan a través del cristalino del ojo. La luz que incide en la retina desencadena una serie de fenómenos químicos y eléctricos que finalmente se traducen en impulsos nerviosos que son enviados hacia el cerebro por el nervio óptico. Hablando de la estructura macroscópica de la retina se pueden apreciar distintas partes:

- Papila óptica: es la cabeza del nervio óptico, donde se agrupan todas las prolongaciones de las neuronas para llevar el impulso hacia el cerebro.

- Arteria y vena de la retina: son las encargadas de nutrir las capas más internas de la retina.
- Mácula: es la zona de la retina encargada de la visión central, de la visión fina de los detalles de las cosas como por ejemplo leer o enhebrar una aguja.
- Fóvea: es el centro de la visión, donde se enfocan los rayos de luz que llegan a la retina.
- Retina periférica: es la que nos permite la visión periférica.
- Vítreo: es el gel que ocupa la cavidad del globo ocular. Es transparente y está pegado a la retina.

En la Figura 1.1 se pueden apreciar las distintas partes de la retina que muestran claramente las imágenes retinográficas.

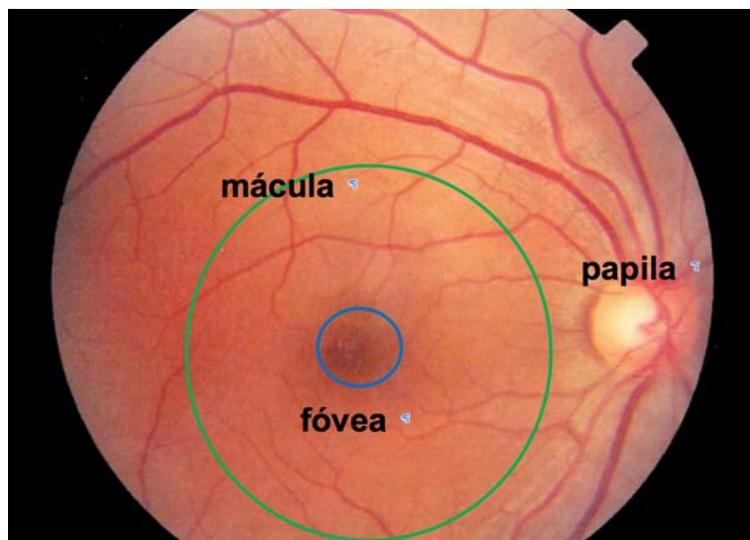


Figura 1.1: Imagen de retinografía.

Por otro lado, la retinografía es una técnica que se utiliza en medicina para obtener fotografías en color de la retina, tanto en imagen panorámica como de zonas más ampliadas, que ayudan a detectar diferentes enfermedades que afectan a la misma. Este tipo de fotografías son tomadas con ayuda de un instrumento llamado retinógrafo ocular.

---

### Método de Split Bregman

---

Muchos de los problemas importantes de ingeniería entran dentro de la categoría de problemas de regularización tipo l1, cuya forma general es:

$$\min_u |\Phi(u)| + H(u), \quad (2.1)$$

donde  $|\cdot|$  denota la norma l1 y tanto  $|\Phi(u)|$  como  $H(u)$  son funciones convexas definidas sobre un conjunto convexo. Muchos de los problemas importantes de procesamiento de imagen (y de otras áreas computacionales) se pueden exponer como un problema de optimización regularizado de tipo l1. Algunos ejemplos de esto podrían ser los siguientes:

$$\text{TVD (Total Variation Denoising)/ROF Denoising: } \min_u \|u\|_{BV} + \frac{\mu}{2} \|u - f\|_2^2, \quad (2.2)$$

$$\text{Basis Pursuit/Compressed Sensing: } \min_u \mathcal{J}(u) + \frac{\mu}{2} \|Au - f\|_2^2, \quad (2.3)$$

donde  $J(u)$  es un funcional cualquiera regularizado normalmente en forma de norma de Besov o norma BV.

A pesar de tener una forma muy simple, la función Rudin-Osher-Fatemi (ROF) que se representa en (2.2), es muy difícil de minimizar por métodos convencionales. La primera vez que se introdujo la recuperación de imágenes basada en la Variación Total en [22], los autores propusieron minimizar esta energía utilizando un método de proyección del gradiente. Este método es

simple pero muy lento debido a las pobres condiciones y a la no linealidad. Muchos autores han propuesto la mejora de los esquemas de tiempo que tiene como resultado un mejor rendimiento, como los presentados en [26] y [15]. Sin embargo, una clase de soluciones más eficientes son aquellas basadas en el método de Newton. Uno de esos algoritmos se presentó en [7], donde el método del gradiente conjugado preconditionado se usa para invertir la Hessiana a cada paso. En [25] se propuso una implementación más eficiente de un método de segundo orden, donde se usa una preconditionación multigrad algebraica para acelerar el método.

Sin embargo, la solución más eficiente a ROF es la propuesta por Darbon y Sigelle en [8], donde se aproximó la función utilizando una norma Besov (BV) anisotrópica. En [3] se demostró que con esta formulación, el problema resultante se resuelve muy rápidamente utilizando cortes de grafos.

Los problemas de la forma (2.3) están recibiendo recientemente mucha atención debido a la introducción de las técnicas de detección comprimida (*compressed sensing*) que permiten la reconstrucción de imágenes y señales de alta resolución a partir de una pequeña cantidad de datos como se ve en [6], [5] y [9]. Se ha demostrado que esta formulación es útil en muchas áreas, incluyendo imagen médica [17], [18], radar [1] y otras aplicaciones de procesamiento de señal. La detección comprimida se basa en la idea de que una señal puede ser reconstruida a partir de un pequeño número de medidas siempre que dichas medidas se obtengan con las bases correctas. Una aplicación particular en la que haremos especial hincapié es la reconstrucción de imágenes de resonancia magnética dispersa (*Sparse MRI*) [17], [16]. El objetivo de la MRI es resolver la siguiente expresión:

$$\min_u \mathcal{J}(u) \text{ tal que } \|\mathcal{F}u - f\|_2 = 0,$$

donde  $f$  representa los datos de la detección comprimida, que consisten en las muestras de la transformada de Fourier de una imagen desconocida,  $\mathcal{F}$  es el subconjunto de filas de la matriz de Fourier,  $u$  la imagen desconocida que queremos reconstruir y  $\mathcal{J}$  el término de regularización  $l_1$  adecuadamente elegido. La formulación para este problema sin restricciones se introdujo en [16], donde se utiliza una aproximación iterativa de Bregman [4] para obtener soluciones a los problemas de reducción de ruido del tipo:

$$\min_u \mathcal{J}(u) \text{ tal que } \|Fu - f\|_2 < \sigma. \quad (2.4)$$

Los problemas de optimización de la forma (2.4), al presentar un término de regularización del tipo  $l_1$ , siguen siendo muy difíciles de resolver. Muchos autores han aplicado esquemas de optimización clásicos, como los métodos de punto interior, a este tipo de problemas.

En este caso especial donde el problema CS se puede escribir de la forma:

$$\min_u |u| \text{ tal que } \|Au - f\|_2 < \sigma, \quad (2.5)$$

se puede utilizar una nueva clase de métodos que reduce el problema a un conjunto de problemas más simples utilizando la linealización. El primero de estos métodos es el FPC (*Fixed Point*

*Continuation*) introducido en [13] y que resuelve el siguiente problema sin restricciones

$$\min_u |u| + \frac{\mu}{2} \|Au - f\|_2^2, \quad (2.6)$$

llevando a cabo de manera iterativa el descenso del gradiente.

También es posible resolver el problema con restricción (2.5) aplicando el esquema de la iteración de Bregman de [20] y [16], utilizando FPC/Bregman. Esto se hace resolviendo de manera iterativa el problema sin restricción (2.6) y modificando el valor de  $f$  usado en la siguiente iteración.

En lugar de resolver el problema sin restricciones y después llevar a cabo la actualización de Bregman de manera separada, estos dos pasos se combinan elegantemente en el algoritmo linearizado de Bregman [28], que resuelve el problema (2.5) resolviendo iterativamente

$$\begin{aligned} v^{k+1} &= v^k + A^T(f - Au^k), \\ u^{k+1} &= \delta * \text{shrink}(v^{k+1}, \frac{1}{\mu}), \end{aligned}$$

para  $k = 1, 2, \dots, n$ . El algoritmo termina cuando se cumple la restricción de reducción en ruido de (2.5). Se demostró en [11] y [21] que  $u^n$  será una aproximación adecuada a la solución de (2.5) siempre que se escojan los valores apropiados de los parámetros  $\mu$  y  $\sigma$ .

A pesar de que el FPC y el algoritmo linearizado de Bregman son extremadamente eficientes, sólo pueden resolver problemas que hayan sido puestos de la forma (2.6). Como el operador gradiente no es invertible, estos algoritmos no pueden resolver problemas que involucren normas BV. Además, estos esquemas tampoco pueden resolver problemas que tengan múltiples términos de regularización l1. Debido a estas razones es difícil aplicar el método FPC y el algoritmo linearizado de Bregman en problemas de procesamiento de imagen. Por ejemplo, varios autores [9], [24] han llegado a la conclusión de que la reconstrucción óptima de una resonancia magnética (MRI) se obtiene utilizando una combinación de la norma BV y de la norma de Besov,  $B_{1,1}$ . En este caso, este problema puede escribirse como:

$$\min_u \|u\|_{BV} + \|u\|_H + \frac{\mu}{2} \|\mathcal{F}u - f\|_2^2,$$

donde  $\|\cdot\|_{BV} = \|\nabla u\|_1$  es la norma "bounded variation" (variación acotada) y  $\|u\|_H = \|Hu\|_1$  denota la norma de Besov respecto a la transformada Wavelet de Haar. Más tarde se volverá a esta forma particular del problema de detección comprimida (*Compressed Sensing*).

A continuación se va a presentar una técnica general que pueda usarse para resolver eficientemente los problemas de regularización l1 más comunes. Además, el esquema de optimización que se empleará se puede generalizar para resolver un rango muy amplio de problemas de optimización, algunos de los cuales pueden ser muy difíciles de resolver usando técnicas existentes.

## 2.1. Problemas de optimización con restricciones

Se considera una función de energía convexa,  $\mathcal{E}$ , y un operador lineal,  $A : \mathbb{R}^n \rightarrow \mathbb{R}^m$ . Se desea resolver un problema de optimización con restricciones generalizado

$$\min_u \mathcal{E}(u) \text{ sujeto a } Au = b. \quad (2.7)$$

Este problema puede ser muy difícil de resolver directamente si  $\mathcal{E}$  es no diferenciable. Para que (2.7) sea más sencillo de resolver, se desea convertirlo en un problema de optimización sin restricciones. Un método común para hacer esto es utilizando una función de penalización/método de continuación, que aproxima (2.7) a un problema de la forma

$$\min_u \mathcal{E}(u) + \frac{\lambda_k}{2} \|Au - b\|_2^2, \quad (2.8)$$

donde  $\lambda_1 < \lambda_2 < \dots < \lambda_N$  es una secuencia creciente de pesos de la función de penalización [2, 19]. Para reforzar el hecho de que  $H(u) \approx 0$ , se debe escoger  $\lambda_N$  para que sea un valor extremadamente grande.

Desafortunadamente, en muchos problemas, escoger un valor grande de  $\lambda$  hace a (2.8) extremadamente difícil de resolver numéricamente. A menudo se desea resolver (2.8) por un método tipo Newton, que requiere invertir la Hessiana de la función objetivo. Sin embargo, conforme  $\lambda_k \rightarrow \infty$ , el número de condiciones de la Hessiana se aproxima a infinito, haciéndolo poco práctico para el uso de métodos iterativos rápidos (como el método del gradiente conjugado o el método de Gauss-Seidel)[2, 19]. Además, para muchas aplicaciones,  $\lambda_k$  debe ser incrementada en pasos muy pequeños, haciendo el método menos eficiente.

En el siguiente apartado se verá que la iteración de Bregman se puede usar para reducir (2.7) a una secuencia corta de problemas sin restricción. En este sentido, la iteración de Bregman es una alternativa a los métodos convencionales de función de penalización.

## 2.2. Iteración de Bregman

La iteración de Bregman es un concepto que se originó en el análisis de funciones para encontrar los extremos de funciones convexas [4]. La iteración de Bregman se usó por primera vez en procesamiento de imagen en [20], donde se aplicó al modelo ROF para la reducción de ruido por técnicas de variación total. También ha sido utilizada para resolver problemas de búsqueda de base (*Basis Pursuit*) en [28], [11] y [21], y en consecuencia aplicada en problemas de imagen médica en [16]. En lugar de centrarnos en una aplicación específica, se va a presentar una formulación general de esta técnica.

Empezamos con el concepto de "distancia de Bregman". La distancia de Bregman asociada a una función convexa  $\mathcal{E}$  en el punto  $v$  es

$$D_{\mathcal{E}}^p(u, v) = \mathcal{E}(u) - \mathcal{E}(v) - \langle p, u - v \rangle,$$

donde  $p$  es el subgradiente de  $\mathcal{E}$  en  $v$ . Claramente, ésta no es una distancia en el sentido habitual ya que no es simétrica en general. Sin embargo sí que mide cercanía en el sentido de que  $D_{\mathcal{E}}^p(u, v) \geq 0$ , y  $D_{\mathcal{E}}^p(u, v) \geq D_{\mathcal{E}}^p(w, v)$  con  $w$  en la línea segmento entre  $u$  y  $v$ .

De nuevo, considerando dos funciones de energía convexas,  $\mathcal{E}$  y  $\mathcal{H}$ , definidas sobre  $\mathbb{R}^n$  con  $\min_{u \in \mathbb{R}^n} \mathcal{H}(u) = 0$ . El problema de minimización sin restricciones asociado es

$$\min_u \mathcal{E}(u) + \lambda \mathcal{H}(u). \quad (2.9)$$

Se puede modificar este problema resolviendo de manera iterativa

$$u^{k+1} = \min_u D_{\mathcal{E}}^p(u, u^k) + \lambda \mathcal{H}(u), \quad (2.10)$$

$$= \min_u \mathcal{E}(u) - \langle p^k, u - u^k \rangle + \lambda \mathcal{H}(u), \quad (2.11)$$

como fue sugerido por Bregman en [4].

Por simplicidad, se asume que  $\mathcal{H}$  es diferenciable. En este caso, se tiene que  $0 \in \partial(D_{\mathcal{E}}^p(u, u^k) + \lambda \mathcal{H}(u))$  donde este subdiferencial se evalúa en  $u^{k+1}$ . Como  $p^{k+1} \in \partial \mathcal{E}(u^{k+1})$  en esta ubicación, se tiene que

$$p^{k+1} = p^k - \nabla \mathcal{H}(u^{k+1}).$$

En [20], los autores analizan la convergencia de los esquemas iterativos de Bregman. En particular, se muestra que, bajo supuestos bastante débiles de  $\mathcal{E}$  y  $\mathcal{H}$ , que  $\mathcal{H}(u) \rightarrow 0$  ya que  $k \rightarrow \infty$ .

Son especialmente importantes dos resultados particulares de convergencia de [20], y por tanto, los reafirmamos.

**Teorema 1.** *Asumiendo que  $\mathcal{E}$  y  $\mathcal{H}$  son funciones convexas y que  $\mathcal{H}$  es diferenciable. También se asume que las soluciones a los sub-problemas en (2.10) existen. Se tiene*

- 1) *Decrecimiento monótono en  $\mathcal{H}$  :  $\mathcal{H}(u^{k+1}) \leq \mathcal{H}(u^k)$*

- 2) *Convergencia a un minimizador de  $\mathcal{H}$  :  $\mathcal{H}(u^k) \leq \mathcal{H}(u^*) + J(u^*)/k$*

Además de esos resultados de convergencia, la iteración de Bregman tiene muchas y muy buenas propiedades para la reducción del ruido que han sido discutidos y probados en [20, 16].

### 2.2.1. Optimización con restricciones vía Iteración de Bregman

En esta sección, se presenta un método para resolver una amplia variedad de problemas de optimización con restricciones sin usar continuación. Primero se muestra que la iteración de Bregman se puede usar para resolver el problema de optimización con restricciones (2.7). Se discutirá

una forma simplificada de iteración de Bregman que es equivalente a “añadir el ruido de nuevo” como se hace con la reducción del ruido ROF [20]. Y por último se discutirá las propiedades de convergencia de este método.

Se desea resolver

$$\min_u \mathcal{E}(u) \text{ tal que } Au = b, \quad (2.12)$$

para algún operador lineal  $A$  y un vector  $b$ . Para aplicar la fórmula (2.9) se convierte esto en un problema sin restricciones utilizando una función de penalización cuadrática:

$$\min_u \mathcal{E}(u) + \frac{\lambda}{2} \|Au - b\|_2^2. \quad (2.13)$$

Para valores pequeños de  $\lambda$ , la función de penalización no hace cumplir la restricción con precisión. La solución convencional a este problema es dejar que  $\lambda \rightarrow \infty$ . En lugar de eso, aquí se aplica la Iteración de Bregman (2.10), y se minimiza de forma iterativa:

$$u^{k+1} = \min_u D_{\mathcal{E}}^p(u, u^k) + \frac{\lambda}{2} \|Au - b\|_2^2 \quad (2.14)$$

$$= \min_u \mathcal{E}(u) - \langle p^k, u - u^k \rangle + \frac{\lambda}{2} \|Au - b\|_2^2, \quad (2.15)$$

$$p^{k+1} = p^k - \lambda A^T (Au^{k+1} - b). \quad (2.16)$$

Las iteraciones de Bregman de esta forma fueron consideradas en [28] y en [20]. Aquí se demuestra que cuando  $A$  es lineal, esta iteración aparentemente complicada es equivalente al método simplificado:

$$u^{k+1} = \min_u \mathcal{E}(u) + \frac{\lambda}{2} \|Au - b^k\|_2^2, \quad (2.17)$$

$$b^{k+1} = b^k + b - Au^k. \quad (2.18)$$

En otras palabras, simplemente se añade de nuevo el error de la restricción en la parte derecha. Esto es análogo a “añadir de nuevo el ruido” en el modelo ROF para la reducción de ruido mediante la variación total [20].

Debido a la equivalencia de (2.14-2.16) y (2.17-2.18) y a los resultados de convergencia del teorema 2.1, tenemos que:

$$\lim_{k \rightarrow \infty} Au^k = b, \quad (2.19)$$

donde la convergencia es en el sentido de la norma 2. En otras palabras, para valores grandes de  $k$ , la iteración de  $u^k$  satisface la restricción con un alto grado de precisión.

Ahora se necesita demostrar que una solución  $u^*$ , de  $Au = b$  obtenida a partir de (2.17-2.18) es ciertamente una solución al problema original con restricción (2.12). Cabe destacar que la iteración de Bregman se usó para resolver un problema de optimización con restricciones en [28]. Aquí se amplía este resultado presentando una prueba muy simple.

**Teorema 2.1.** Siendo  $\mathcal{H} : \mathbb{R}^n \rightarrow \mathbb{R}$  convexa. Siendo  $A : \mathbb{R}^n \rightarrow \mathbb{R}^m$  lineal. Se considera el algoritmo (2.17-2.18). Se supone que alguna iteración,  $u^*$ , satisface  $Au^* = b$ . Entonces  $u^*$  es una solución al problema con restricción original (2s.12)

**Prueba:** Sea  $u^*$  y  $b^*$  tales que  $Au^* = b$ , y

$$u^* = \underset{u}{\text{mín}} E(u) + \frac{\lambda}{2} \|Au - b^*\|_2^2. \quad (2.20)$$

Sea  $\hat{u}$  una verdadera solución de (2.12). Entonces  $Au^* = b = A\hat{u}$ , lo que implica que

$$\|Au^* - b^*\|_2^2 = \|A\hat{u} - b^*\|_2^2. \quad (2.21)$$

Ya que  $u^*$  satisface (1.20), se tiene que

$$E(u^*) + \frac{\lambda}{2} \|Au^* - b^*\|_2^2 \leq E(\hat{u}) + \frac{\lambda}{2} \|A\hat{u} - b^*\|_2^2. \quad (2.22)$$

Finalmente, se aprecia que (2.21) y (2.22) juntos implican

$$E(u^*) \leq E(\hat{u}).$$

Como  $\hat{u}$  satisface el problema de optimización original, esta desigualdad se puede enfocar como una igualdad demostrando que  $u^*$  resuelve (2.12).

Esto muestra que, siempre que (2.17-2.18) converjan en el sentido de (2.19), la iteración de  $u^k$  se acercará de manera arbitraria a la solución del problema original con restricción (2.7). Hay que remarcar la generalidad del teorema de arriba. La prueba no utiliza de manera explícita la linealidad de  $A$ , y de hecho no es una condición que el teorema requiera para sostenerse. La aplicación de este método a problemas en los que  $A$  no es lineal será sujeto de estudio en futuras investigaciones.

### 2.2.2. Ventajas de la iteración de Bregman

Esta técnica de la iteración de Bregman tiene muchas ventajas sobre la tradicional función de penalización o los métodos de continuación. En primer lugar, la iteración de Bregman converge muy rápidamente cuando se aplica a ciertos tipos de función objetivo, especialmente para los problemas donde  $\mathcal{E}$  contiene un término de regularización  $l_1$ . Cuando la iteración de Bregman converge muy rápidamente, sólo se necesita resolver un número pequeño de problemas sin restricción.

La segunda (y quizá más importante) ventaja de la iteración de Bregman sobre los métodos de continuación es que el valor de  $\lambda$  en (2.9) permanece constante. Se puede por lo tanto elegir

un valor de  $\lambda$  que minimice el número de condición de los sub-problemas, dando lugar a una convergencia más rápida para los métodos de optimización iterativos como Newton o Gauss-Seidel.

La Iteración de Bregman también evita el problema de la inestabilidad numérica que ocurre cuando  $\lambda \rightarrow \infty$  que aparece cuando se usan los métodos de continuación.

### 2.3. Split Bregman. Una mejor formulación para los problemas regularizados l1

Se va a aplicar la infraestructura de Bregman para resolver el problema general de optimización regularizado l1 (2.1). En la explicación siguiente se asumirá que  $\mathcal{H}(\cdot)$  y  $|\Phi(\cdot)|$  son funciones convexas. También se asumirá que  $|\Phi(\cdot)|$  es diferenciable.

La clave de este método es que se desaparejan las partes l1 y l2 de la energía en (2.1). La formulación de esta separación sigue lo propuesto en [27] donde se aplica una técnica similar para deconvoluciones regularizadas l1. En lugar de considerar (2.1) se considerará el problema

$$\min_{u,d} |d| + \mathcal{H}(u) \text{ tal que } d = \Phi(u). \quad (2.23)$$

Este problema es claramente equivalente a (2.1). Para resolver este problema, primero hay que convertirlo en un problema sin restricciones:

$$\min_{u,d} |d| + \mathcal{H}(u) + \frac{\lambda}{2} \|d - \Phi(u)\|_2^2. \quad (2.24)$$

En esto es en lo que este método difiere de [27]. Si se considera  $E(u, d) = |d| + \mathcal{H}(u)$ , y se define  $A(u, d) = d - \Phi(u)$ , entonces se puede ver que (2.24) es simplemente una aplicación de (2.13). Para reforzar la condición de restricción, ahora este problema se conecta con la formulación de Bregman que aparece anteriormente (2.14-2.16)

$$(u^{k+1}, d^{k+1}) = \min_{u,d} D_{\mathcal{E}}^p(u, u^k, d, d^k) + \frac{\lambda}{2} \|d - \Phi(u)\|_2^2 \quad (2.25)$$

$$= \min_{u,d} \mathcal{E}(u, d) - \langle p_u^k, u - u^k \rangle - \langle p_d^k, d - d^k \rangle + \frac{\lambda}{2} \|d - \Phi(u)\|_2^2, \quad (2.26)$$

$$p_u^{k+1} = p_u^k - \lambda (\nabla \Phi)^T (\Phi u^{k+1} - d^{k+1}), \quad (2.27)$$

$$p_d^{k+1} = p_d^k - \lambda (d^{k+1} - \Phi u^{k+1}). \quad (2.28)$$

Cuando se aplica la simplificación presentada en (2.17-2.18), se obtiene el elegante algoritmo de dos fases

*La iteración de Split Bregman*

$$(u^{k+1}, d^{k+1}) = \underset{u, d}{\text{mín}} |d| + \mathcal{H}(u) + \frac{\lambda}{2} \|d - \Phi(u) - b^k\|_2^2, \quad (2.29)$$

$$b^{k+1} = b^k + (\Phi(u^{k+1}) - d^{k+1}). \quad (2.30)$$

Se ha reducido el problema regularizado l1 (2.1) a una secuencia de problemas de optimización sin restricción y actualizaciones de Bregman. El porqué de que este algoritmo sea tan efectivo puede no ser inmediatamente claro. En el siguiente apartado se verá que este problema es mucho más fácil de computar que la formulación convencional para un problema regularizado l1.

### 2.3.1. Minimización iterativa

Para poder implementar el algoritmo (2.30) hay que ser capaz de resolver el problema

$$(u^{k+1}, d^{k+1}) = \underset{u, d}{\text{mín}} |d| + \mathcal{H}(u) + \frac{\lambda}{2} \|d - \Phi(u) - b^k\|_2^2. \quad (2.31)$$

Debido a la forma en la que se han “separado”, las componentes l1 y l2 de esta función, se puede llevar a cabo una minimización eficiente minimizando iterativamente con respecto a  $u$  y a  $d$  de manera separada. Los dos pasos que se deben llevar a cabo son:

$$\text{Paso 1: } u^{k+1} = \underset{u}{\text{mín}} \mathcal{H}(u) + \frac{\lambda}{2} \|d^k - \Phi(u) - b^k\|_2^2.$$

$$\text{Paso 2: } d^{k+1} = \underset{d}{\text{mín}} |d| + \frac{\lambda}{2} \|d - \Phi(u^{k+1}) - b^k\|_2^2.$$

La velocidad del modelo de separación de Bregman es en gran parte dependiente de cómo de rápido se pueda resolver cada uno de esos dos subproblemas.

Para resolver el Paso 1, es notorio que ya que se tiene  $u$  “desparejada” de la parte l1 del problema, el problema de optimización que se debe resolver para  $u^k$  es ahora diferenciable. Por tanto, se puede usar una gran variedad de técnicas de optimización para resolver este problema. El método particular para resolver este problema de optimización depende de la naturaleza exacta de  $H$ , pero para la mayoría de los problemas comunes se pueden usar los métodos tanto de Gauss-Seidel como de la Transformada de Fourier. Para escasos problemas en los que  $\Phi$  tiene una estructura pequeña, se puede usar un método de pocos pasos de un gradiente conjugado para resolver este problema de manera aproximada.

En el Paso 2 del algoritmo anterior, no hay acoplamiento entre los elementos de  $d$ . Se puede computar explícitamente el valor óptimo de  $d$  utilizando el operador de encogimiento (shrinkage). Se computa simplemente

$$d_j^{k+1} = \text{shrink}(\Phi(u)_j + b_j^k, 1/\lambda),$$

donde

$$\text{shrink}(x, \gamma) = \frac{x}{|x|} * \text{máx}(|x| - \gamma, 0),$$

este encogimiento es extremadamente rápido, y requiere sólo unas pocas operaciones por elemento de  $d^{k+1}$ .

### 2.3.2. Implementación del algoritmo propuesto

Cuando se coloca el esquema de minimización iterativa dentro del proceso descrito en (2.29), se obtiene lo siguiente:

#### Algoritmo generalizado de Split Bregman

```

While  $\|u^k - u^{k-1}\|_2 > tol$ 
  for n=1 to N
     $u^{k+1} = \text{mín}_u \mathcal{H}(u) + \frac{\lambda}{2} \|d^k - \Phi(u) - b^k\|_2^2$ 
     $d^{k+1} = \text{mín}_d |d| + \frac{\lambda}{2} \|d - \Phi(u^{k+1}) - b^k\|_2^2$ 
  end
   $b^{k+1} = b^k + (\Phi(u^{k+1}) - d^{k+1})$ 
end

```

Se ha encontrado que no es deseable resolver el primer subproblema en (2.29) para una convergencia total. Intuitivamente, la razón para esto es que si el error en la solución de este subproblema es pequeña comparada con  $\|b^k - b^*\|_2$ , entonces este extra de precisión será "malgastado" cuando el parámetro de Bregman se actualice. De hecho, se ha demostrado empíricamente que para muchas aplicaciones, la eficiencia óptima se obtiene cuando sólo se ha llevado a cabo una iteración del bucle interior (por ejemplo N=1 en el algoritmo de arriba). Incluso cuando sólo se resuelve para  $u^{k+1}$  aproximadamente (por ejemplo usando unos pocos pasos de un método iterativo), el algoritmo anterior aún converge.

Para entender por qué el algoritmo de Split Bregman es tan robusto a las imprecisiones numéricas, se deben examinar los resultados del Teorema 1.2.2. Utilizando este teorema es fácil demostrar que cualquier punto fijo del algoritmo de Split Bregman es ciertamente un mínimo

del problema con restricción original (2.23), incluso si se usan métodos iterativos inexactos para cada subproblema. Siendo  $(u^*, b^*)$  un punto fijo de (2.29-2.30) que también satisface (2.29). El punto fijo satisface  $b^* = b^* + \Phi u^* - d^*$ , lo que implica que  $d^* = \Phi u^*$ . Estos resultados, junto con (2.29) satisfacen las condiciones del Teorema 1.2.2, que muestra que  $(u^*, b^*)$  es una solución del problema con restricción (2.23).

## 2.4. Aplicaciones

Ahora se ilustrará cómo usar la infraestructura de Split Bregman para tratar distintas aplicaciones.

### 2.4.1. Eliminación del ruido por Variación Total

La reducción del ruido por Variación Total es considerado uno de los mejores modelos pero también uno de los más difíciles de computar. En este apartado se va a mostrar cómo la técnica de Split Bregman puede ser usada para resolver este problema de una forma no sólo simple, también extremadamente eficiente. Es más, este modelo puede resolver el problema de minimización de Variación Total Isotrópica (un modelo superior de Variación Total que no se puede resolver usando el popular método de cortes de grafos). Se tratará esto como un problema bidimensional para demostrar que el método de Split Bregman aplica a problemas con más términos que el término de regularización l1.

Se empieza abordando el problema anisotrópico

$$\min_u |\nabla_x u| + |\nabla_y u| + \frac{\mu}{2} \|u - f\|_2^2. \quad (2.32)$$

Para aplicar la división de Bregman, primero se sustituye  $\nabla_x u$  por  $d_x$  y  $\nabla_y u$  por  $d_y$ . Esto da paso al problema con restricción.

$$\min_u |d_x| + |d_y| + \frac{\mu}{2} \|u - f\|_2^2, \text{ tal que } d_x = \nabla_x u \text{ y } d_y = \nabla_y u.$$

En esta formulación se refuerza débilmente la restricción, se añaden términos de función de penalización como se hizo en (2.24). Ahora se tiene

$$\min_{d_x, d_y, u} |d_x| + |d_y| + \frac{\mu}{2} \|u - f\|_2^2 + \frac{\lambda}{2} \|d_x - \nabla_x u\|_2^2 + \frac{\lambda}{2} \|d_y - \nabla_y u\|_2^2.$$

Finalmente, se refuerzan estrictamente las restricciones aplicando la iteración de Bregman (2.29) para conseguir

$$\min_{d_x, d_y, u} |d_x| + |d_y| + \frac{\mu}{2} \|u - f\|_2^2 + \frac{\lambda}{2} \|d_x - \nabla_x u - b_x^k\|_2^2 + \frac{\lambda}{2} \|d_y - \nabla_y u - b_y^k\|_2^2$$

que tiene la condición óptima.

$$(\mu I - \lambda \Delta)u^{k+1} = \mu f + \lambda \nabla_x^T (d_x^k - b_x^k) + \lambda \nabla_y^T (d_y^k - b_y^k). \quad (2.33)$$

Con el fin de conseguir una eficiencia óptima, se desea usar un algoritmo iterativo rápido para alcanzar soluciones aproximadas a este sistema. Como el sistema es estrictamente dominante diagonalmente, la elección más natural es el método de Gauss-Seidel. La solución Gauss-Seidel a este problema se puede escribir en cuanto al componente como  $u_{i,j}^{k+1} = G_{i,j}^k$  donde

$$G_{i,j}^k = \frac{\lambda}{\mu + 4\lambda} (u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k + d_{x,i-1,j}^k - d_{x,i,j}^k + d_{y,i,j-1}^k - d_{y,i,j}^k - b_{x,i-1,j}^k + b_{x,i,j}^k - b_{y,i,j-1}^k + b_{y,i,j}^k) + \frac{\mu}{\mu + 4\lambda} f_{i,j}.$$

Usando esta solución, el algoritmo de Split Bregman se escribe

### Eliminación del ruido por Variación Total Anisotrópica con Split Bregman

Inicialización:  $u^0 = f$ , y  $d_x^0 = d_y^0 = b_x^0 = b_y^0 = 0$

While  $\|u^k - u^{k-1}\|_2 > tol$

$$u^{k+1} = G^k$$

$$d_x^{k+1} = shrink(\nabla_x u^{k+1} + b_x^k, 1/\lambda)$$

$$d_y^{k+1} = shrink(\nabla_y u^{k+1} + b_y^k, 1/\lambda)$$

$$b_x^{k+1} = b_x^k + (\nabla_x u^{k+1} - d_x^{k+1})$$

$$b_y^{k+1} = b_y^k + (\nabla_y u^{k+1} - d_y^{k+1})$$

end

se nota que el bucle “for” del algoritmo Generalizado de Split Bregman está ausente aquí. Se ha demostrado que este algoritmo consigue una eficiencia óptima cuando este bucle se ejecuta sólo una vez por iteración, y por lo tanto, se ha eliminado para una mayor claridad. Puede ser necesario incluir este bucle en aplicaciones donde se necesite una alta precisión en los resultados.

El modelo de Variación Total Isotrópica también se puede minimizar usando la técnica de Split Bregman. En este caso se quiere resolver

$$\min_u \sum_i \sqrt{(\nabla_x u)_i^2 + (\nabla_y u)_i^2} + \frac{\mu}{2} \|u - f\|_2^2.$$

Justo como se hizo para el problema anisotrópico, se separarán las componentes l1 y l2 del problema estableciendo  $d_x \approx \nabla_x u$  y  $d_y \approx \nabla_y u$ . La formulación de Split Bregman del problema se convierte entonces en:

$$\min_{u, d_x, d_y} \|(d_x, d_y)\|_2 + \frac{\mu}{2} \|u - f\|_2^2 + \frac{\lambda}{2} \|d_x - \nabla_x u - b_x\|_2^2 + \frac{\lambda}{2} \|d_y - \nabla_y u - b_y\|_2^2,$$

donde

$$\|(d_x, d_y)\|_2 = \sum_{i,j} \sqrt{d_{x,i,j}^2 + d_{y,i,j}^2}. \quad (2.34)$$

Hay que remarcar que las variables  $d_x$  y  $d_y$  no se desparejan como se hizo en el caso anisotrópico. Esto cambia la manera en la que se deben tratar las variables. Para aplicar el proceso de minimización iterativa a este problema, se debe resolver este subproblema

$$(d_x^{k+1}, d_y^{k+1}) = \underset{d_x, d_y}{\text{mín}} \|(d_x, d_y)\|_2 + \frac{\lambda}{2} \|d_x - \nabla_x u - b_x\|_2^2 + \frac{\lambda}{2} \|d_y - \nabla_y u - b_y\|_2^2.$$

A pesar del hecho de que  $d_x$  y  $d_y$  no se desparejan como se hizo en el caso anisotrópico, se puede resolver explícitamente el problema de minimización para  $(d_x^{k+1}, d_y^{k+1})$  utilizando una fórmula generalizada de achicamiento [27]:

$$d_x^{k+1} = \text{máx}(s^k - 1/\lambda, 0) \frac{\nabla_x u^k + b_x^k}{s^k}, \quad (2.35)$$

$$d_y^{k+1} = \text{máx}(s^k - 1/\lambda, 0) \frac{\nabla_y u^k + b_y^k}{s^k}, \quad (2.36)$$

donde

$$s^k = \sqrt{|\nabla_x u^k + b_x^k|^2 + |\nabla_y u^k + b_y^k|^2}. \quad (2.37)$$

Si se aplica la iteración de Bregman a este problema, se obtiene el algoritmo de minimización para la función de Variación Total Isotrópica:

### Eliminación del ruido por Variación Total Isotrópica con Split Bregman

Inicialización:  $u^0 = f$ , y  $d_x^0 = d_y^0 = b_x^0 = b_y^0 = 0$

While  $\|u^k - u^{k-1}\|_2 > \text{tol}$

$$u^{k+1} = G^k$$

$$d_x^{k+1} = \text{máx}(s^k - 1/\lambda, 0) \frac{\nabla_x u^k + b_x^k}{s^k}$$

$$d_y^{k+1} = \text{máx}(s^k - 1/\lambda, 0) \frac{\nabla_y u^k + b_y^k}{s^k}$$

$$b_x^{k+1} = b_x^k + (\nabla_x u^{k+1} - d_x^{k+1})$$

$$b_y^{k+1} = b_y^k + (\nabla_y u^{k+1} - d_y^{k+1})$$

end

donde  $s^k$  es definida arriba en (2.35).

### 2.4.2. Compressed Sensing rápida para la reconstrucción de imágenes

La detección comprimida o Compressed Sensing (CS) es un área emergente de Imagen Médica, y muchos predicen que será algún día una herramienta ordinaria para los radiólogos. Con-

secuenteemente, los algoritmos rápidos para este problema son extremadamente deseables.

La formulación exacta del problema de optimización CS depende de alguna manera de la aplicación que se esté considerando. Para un propósito demostrativo, habrá que centrarse en la aplicación de CS para imágenes de Resonancia Magnética Dispersa (MRI). Se elige esta aplicación no sólo porque es la mejor de CS en este campo, sino también porque las dificultades de este problema permiten demostrar la versatilidad del método de Split Bregman.

La forma general para los problemas de reconstrucción Sparse MRI se presenta y se discute en [17], [16] y [14]. En esta ocasión se elige escribir el problema de reconstrucción de MRI dispersa de la siguiente forma

$$\min_u J(u) \text{ tal que } \|RFu - f^k\|_2^2 < \sigma^2, \quad (2.38)$$

donde  $F$  representa la matriz de transformada de Fourier,  $f$  representa el espacio- $k$  de datos observados, y  $\sigma$  representa la varianza del ruido de la señal. La matriz  $R$  representa una matriz "selectora de filas", que constituye un subconjunto de las filas de una matriz identidad. Además,  $J(u)$  representa un término regularizado l1 cualquiera.

En [28] se muestra que, utilizando la técnica de Iteración de Bregman, el problema (2.36) se podría reducir a una secuencia de problemas sin restricción de la forma

$$u^{k+1} = \min_u J(u) + \frac{\mu}{2} \|RFu - f^k\|_2^2, \quad (2.39)$$

$$f^{k+1} = f^k + f - RFu^{k+1}. \quad (2.40)$$

Este es el problema sin restricción que se desea resolver utilizando la técnica de Split Bregman.

Para ser concretos sobre el término de regularización que se está usando, se elegirá una forma específica para  $J(\cdot)$ . Muchos autores han observado que las mejores reconstrucciones de imágenes ocurren cuando se utilizan híbridos de Variación Total y regularizadores de Besov. Siguiendo a estos autores, se elige  $J(u) = \|u\|_{BV} + \|u\|_{B_{1,1}} = |\nabla u| + |Wu|$ , donde  $W$  representa la transformada discreta ondícula ortogonal de Haar. A pesar de que puede parecer que los regularizadores de BV y Haar son muy parecidos, se ha demostrado que la inclusión de múltiples regularizadores ayuda a asegurar una reconstrucción precisa de imágenes lisas. Ya que la inclusión de estas transformadas "sobrecpletas" tiene pequeños efectos en la velocidad de convergencia del algoritmo de Split Bregman, y ambas transformadas se pueden evaluar rápidamente, hay un pequeño coste adicional al incluir dichos regularizadores.

Cabe destacar que ahora  $u$  puede tomar valores complejos, por lo que se debe ser preciso en la notación. A continuación se tiene  $|v| = \|v\|_1 = \sum_i \sqrt{v_i^H v_i}$  donde  $v_i^H$  denota la transposición Hermitiana del vector  $v_i$ . Utilizando esta definición de la norma l1, se puede definir  $\|v\|_{BV} =$

$$|\nabla v| = \sum_i \sqrt{|\nabla_x v|^2 + |\nabla_y v|^2}.$$

Para aplicar el método de Split Bregman a este problema, primero se hace el cambio de variable  $w \leftarrow Wu$ ,  $d_x \leftarrow \nabla_x u$ , y  $d_y \leftarrow \nabla_y u$ . La formulación dividida del problema se convierte en

$$\begin{aligned} \min_{u, d_x, d_y, w} & \|(d_x, d_y)\|_2 + |w| + \frac{\mu}{2} \|RFu - f\|_2^2 + \frac{\lambda}{2} \|d_x - \nabla_x u - b_x\|_2^2 \\ & + \frac{\lambda}{2} \|d_y - \nabla_y u - b_y\|_2^2 + \frac{\gamma}{2} \|w - Wu - b_w\|_2^2. \end{aligned}$$

Donde se ha usado la notación corta

$$\|(d_x, d_y)\|_2 = \sum_{i,j} \sqrt{|d_{x,i,j}|^2 + |d_{y,i,j}|^2},$$

que es el análogo complejo de (2.34). Cabe destacar que  $d_x$ ,  $d_y$ ,  $b_x$  y  $b_y$  son ahora valores complejos, y se han obtenido aplicando el operador diferencia a las partes real y compleja de  $u$  por separado.

Se descompone esta minimización en subproblemas utilizando el proceso de minimización iterativa expuesta anteriormente. Se debe usar la fórmula generalizada de encogimiento (2.35) para resolver los valores óptimos de  $d_x$  y  $d_y$ , y la fórmula estándar de encogimiento para resolver el valor óptimo de  $w$ . Para encontrar el valor óptimo de  $u$ , se debe resolver el subproblema de optimización

$$u^{k+1} = \min_u \frac{\mu}{2} \|Fu - f\|_2^2 + \frac{\lambda}{2} \|d_x^k - \nabla_x u - b_x^k\|_2^2 + \frac{\lambda}{2} \|d_y^k - \nabla_y u - b_y^k\|_2^2 + \frac{\gamma}{2} \|w^k - Wu - b_w^k\|_2^2$$

Ya que este subproblema es diferenciable, las condiciones óptimas para  $u^{k+1}$  son fácilmente derivadas. Diferenciando con respecto a  $u$  y estableciendo el resultado igual a cero, se obtiene la regla actualizada

$$(\mu F^T R^T + \lambda \nabla_x^T \nabla_x + \lambda \nabla_y^T \nabla_y + \gamma W^T W) u^{k+1} = rhs^k,$$

donde

$$rhs^k = \mu F^T R f - \lambda \nabla_x^T (d_x^k - b_x) + \lambda \nabla_y^T (d_y^k - b_y) + \gamma W^T (w - b_w),$$

representa el lado derecho en la ecuación de arriba.

Ahora se aprovechan las identidades  $\nabla^T \nabla = -\Delta$ ,  $W^T W = I$ , y  $F^T = F^{-1}$  para obtener

$$(\mu F^T R^T F - \lambda \Delta + \gamma I) u^{k+1} = rhs^k.$$

Por lo tanto, el sistema que debe ser invertido para resolver  $u^{k+1}$  es circular. Así, se puede escribir el sistema como  $F^{-1} K F$ , donde  $K$  es el operador diagonal

$$K = (\mu R^T R - \lambda F \Delta F^{-1} + \gamma I).$$

Debido a la estructura circular del sistema, se puede resolver el valor óptimo de  $u^{k+1}$  usando sólo dos transformadas de Fourier.

Cuando se ponen todos esos elementos juntos, se obtiene el siguiente algoritmo:

**Algoritmo de optimización CS sin restricciones**

Inicialización:  $u^0 = F^{-1}f$  y  $d_x^0 = d_y^0 = w^0 = b_x^0 = b_y^0 = b_w^0 = 0$   
 While  $\|u^k - u^{k-1}\|_2 > tol$   
 $u^{k+1} = F^{-1}K^{-1}Frhs^k$   
 $d_x^{k+1} = \text{máx}(s^k - 1/\lambda, 0) \frac{\nabla_x u^k + b_x^k}{s^k}$   
 $d_y^{k+1} = \text{máx}(s^k - 1/\lambda, 0) \frac{\nabla_y u^k + b_y^k}{s^k}$   
 $w^{k+1} = \text{shrink}(Wu^{k+1} + b_w^k, 1/\gamma)$   
 $b_x^{k+1} = b_x^k + (\nabla_x u^{k+1} - d_x^{k+1})$   
 $b_y^{k+1} = b_y^k + (\nabla_y u^{k+1} - d_y^{k+1})$   
 $b_w^{k+1} = b_w^k + (Wu^{k+1} - w^{k+1})$   
 end

donde  $s^k$  se define en (2.35).

Cabe destacar que este algoritmo sólo resuelve el problema CS sin restricción (2.37). Para resolver el problema con restricción (2.36), se debe reemplazar  $f$  por  $f^k$  en el algoritmo de arriba. Después de resolver de manera aproximada cada problema sin restricción, se debe aplicar la regla de actualización de Bregman  $f^{k+1} = f^k + f - RFu^{k+1}$ . Cuando se integra el algoritmo sin restricción dentro de esta actualización externa de Bregman, queda:

**Algoritmo de optimización CS con restricciones**

Inicialización:  $u^0 = F^{-1}f$  y  $d_x^0 = d_y^0 = w^0 = b_x^0 = b_y^0 = b_w^0 = 0$   
 While  $\|RFu^k - f\|_2^2 > \sigma^2$   
 For  $i=1$  to  $N$   
 $u^{k+1} = F^{-1}K^{-1}Frhs^k$   
 $d_x^{k+1} = \text{máx}(s^k - 1/\lambda, 0) \frac{\nabla_x u^k + b_x^k}{s^k}$   
 $d_y^{k+1} = \text{máx}(s^k - 1/\lambda, 0) \frac{\nabla_y u^k + b_y^k}{s^k}$   
 $w^{k+1} = \text{shrink}(Wu^{k+1} + b_w^k, 1/\gamma)$   
 $b_x^{k+1} = b_x^k + (\nabla_x u^{k+1} - d_x^{k+1})$   
 $b_y^{k+1} = b_y^k + (\nabla_y u^{k+1} - d_y^{k+1})$   
 $b_w^{k+1} = b_w^k + (Wu^{k+1} - w^{k+1})$   
 end  
 $f^{k+1} = f^k + f - RFu^{k+1}$   
 end

La velocidad de este algoritmo dependerá en gran medida de cuántas veces se necesite ejecutar el bucle "While". Cuando los valores de los parámetros se escogen de una manera apropiada, se ha demostrado que el bucle externo de este algoritmo sólo necesita ejecutarse un número pequeño de veces. Además, para aplicaciones de imagen no es necesario resolver cada sub-problema sin restricción con precisión numérica completa. Como consecuencia, este algoritmo es muy rápido cuando se utilizan los valores apropiados para los parámetros.



### 3.1. Pruebas con ruido Gaussiano

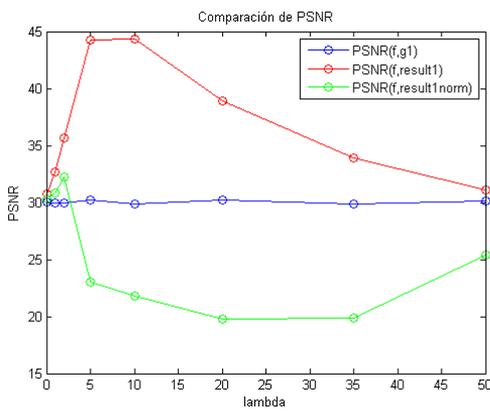
Con el fin de comprender mejor el funcionamiento del código que se va a utilizar para calcular la Variación Total a las imágenes de retinografías, primero se va a probar dicho código con imágenes más sencillas a las que se les ha introducido un ruido Gaussiano. A estas imágenes ruidosas se les aplica la Variación Total y, al hacerlo, se observa que su intensidad se ve reducida con respecto a la imagen con ruido original, por ello el siguiente paso a la aplicación de la Variación Total será hacer la normalización de la imagen resultante para que ésta recupere la intensidad inicial. En este algoritmo existen tres parámetros principales ( $\mu$ ,  $\lambda$  y varianza) a los que se les va a ir variando su valor en las distintas pruebas. La imagen inicial desde la que se parte es la se ve en la Figura 3.1.

El primer valor que se va tener en cuenta es el de  $\lambda$ , que en el algoritmo utilizado se considera el parámetro de regularización, es decir, el peso que tiene la Variación Total sobre la imagen final. Fijando  $varianza = 0,001$  y  $\mu = 1$ , se va a hacer que este parámetro tome los siguientes valores,  $\lambda = [0.1, 2, 5, 10, 35, 50]$  para así obtener respecto a él los valores de PSNR, SSIM y Ratio de correlación en las imágenes con ruido, con Variación Total y, finalmente, normalizadas.

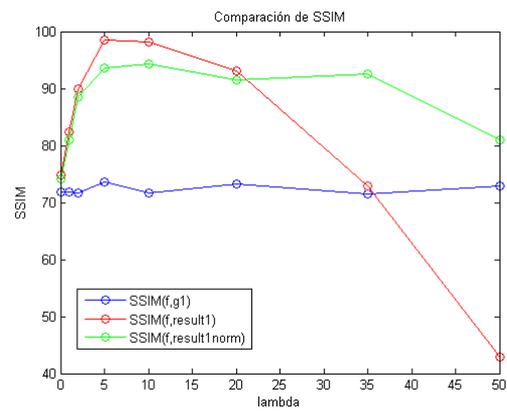
En la Figura 3.2 se obtienen las gráficas para PSNR, SSIM y Ratio de Correlación.



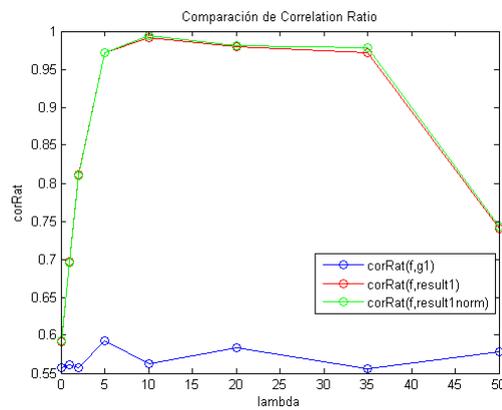
Figura 3.1: Imagen sintética utilizada para las pruebas.



(a)

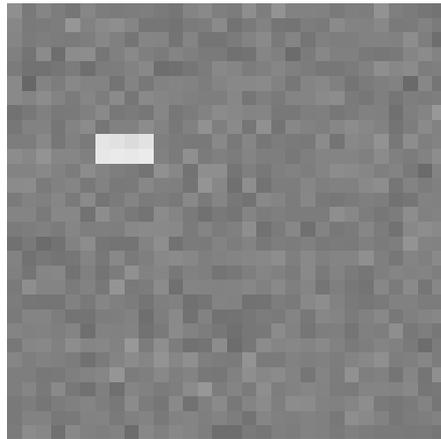


(b)



(c)

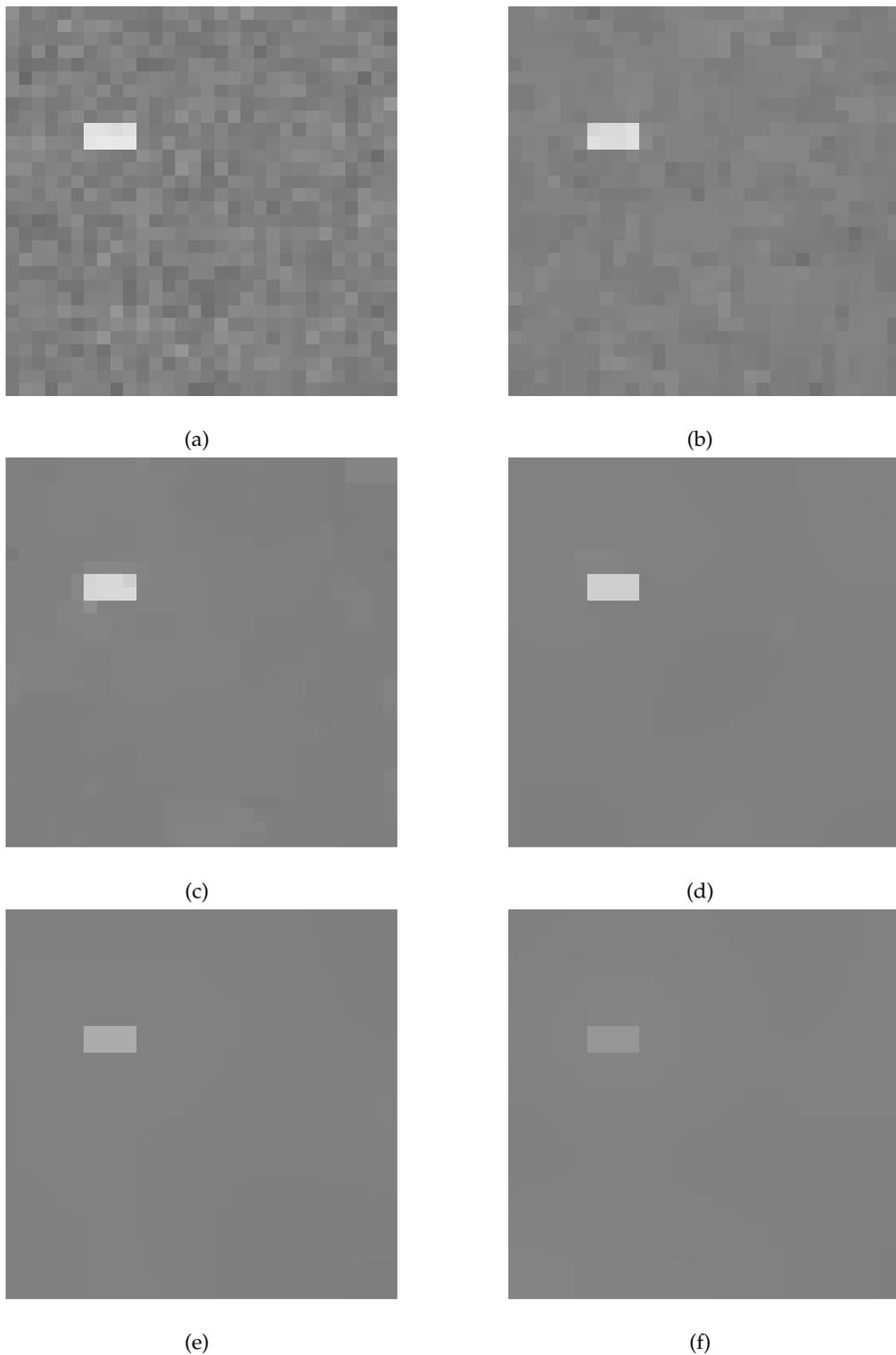
Figura 3.2: Variación de los valores según  $\lambda$  respecto a la imagen original. (a) PSNR, (b) SSIM y (c) Ratio de correlación.



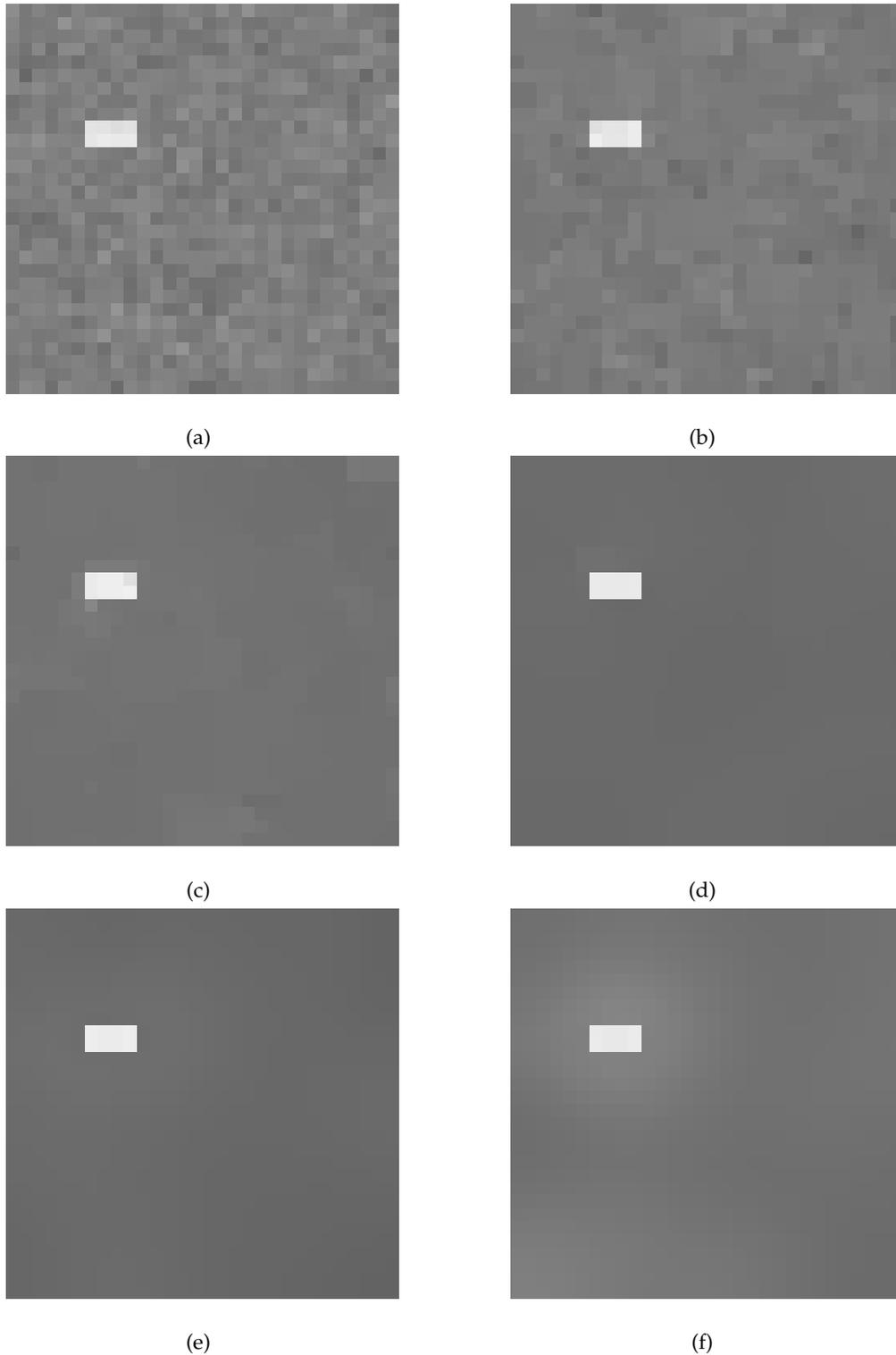
**Figura 3.3:** Imagen con ruido según los valores de  $\lambda$ .

Además, si se observan las distintas imágenes que se han obtenido como resultado de introducir ruido Gaussiano a la original, es destacable que la imagen con ruido es la misma en los seis distintos casos planteados, la de la Figura 3.3. Las imágenes a las que se les aplica Variación Total y las normalizadas sí que sufren cambios.

En la Figura 3.4 se ven las imágenes con ruido a las que se les ha aplicado la variación total y las normalizadas se pueden observar en la Figura 3.5.



**Figura 3.4:** Resultados con Variación Total según el valor de  $\lambda$  (a)  $\lambda = 0,1$  y 14 iteraciones , (b)  $\lambda = 2$  y 11 iteraciones, (c)  $\lambda = 5$  y 11 iteraciones, (d)  $\lambda = 10$  y 11 iteraciones, (e)  $\lambda = 35$  y 10 iteraciones, (f)  $\lambda = 50$  y 11 iteraciones.



**Figura 3.5:** Resultados normalizados con Variación Total según el valor de  $\lambda$  (a)  $\lambda = 0,1$  y 14 iteraciones, (b)  $\lambda = 2$  y 11 iteraciones, (c)  $\lambda = 5$  y 11 iteraciones, (d)  $\lambda = 10$  y 11 iteraciones, (e)  $\lambda = 35$  y 10 iteraciones, (f)  $\lambda = 50$  y 11 iteraciones.

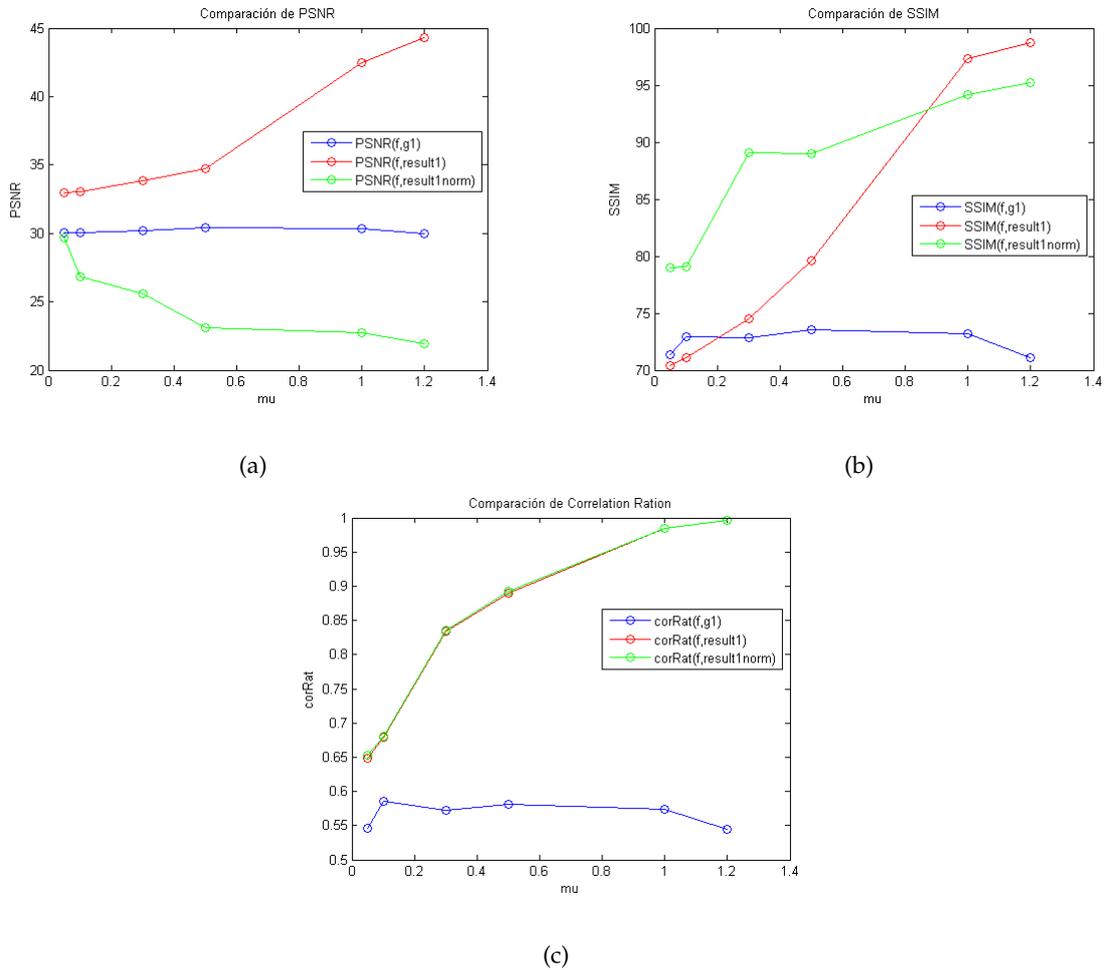
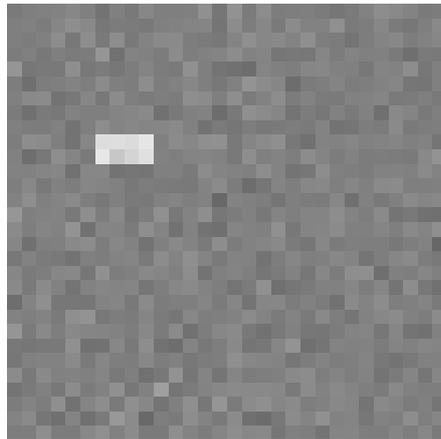


Figura 3.6: Variación de los valores según  $\mu$  respecto a la imagen original. (a) PSNR, (b) SSIM y (c) Ratio de correlación.

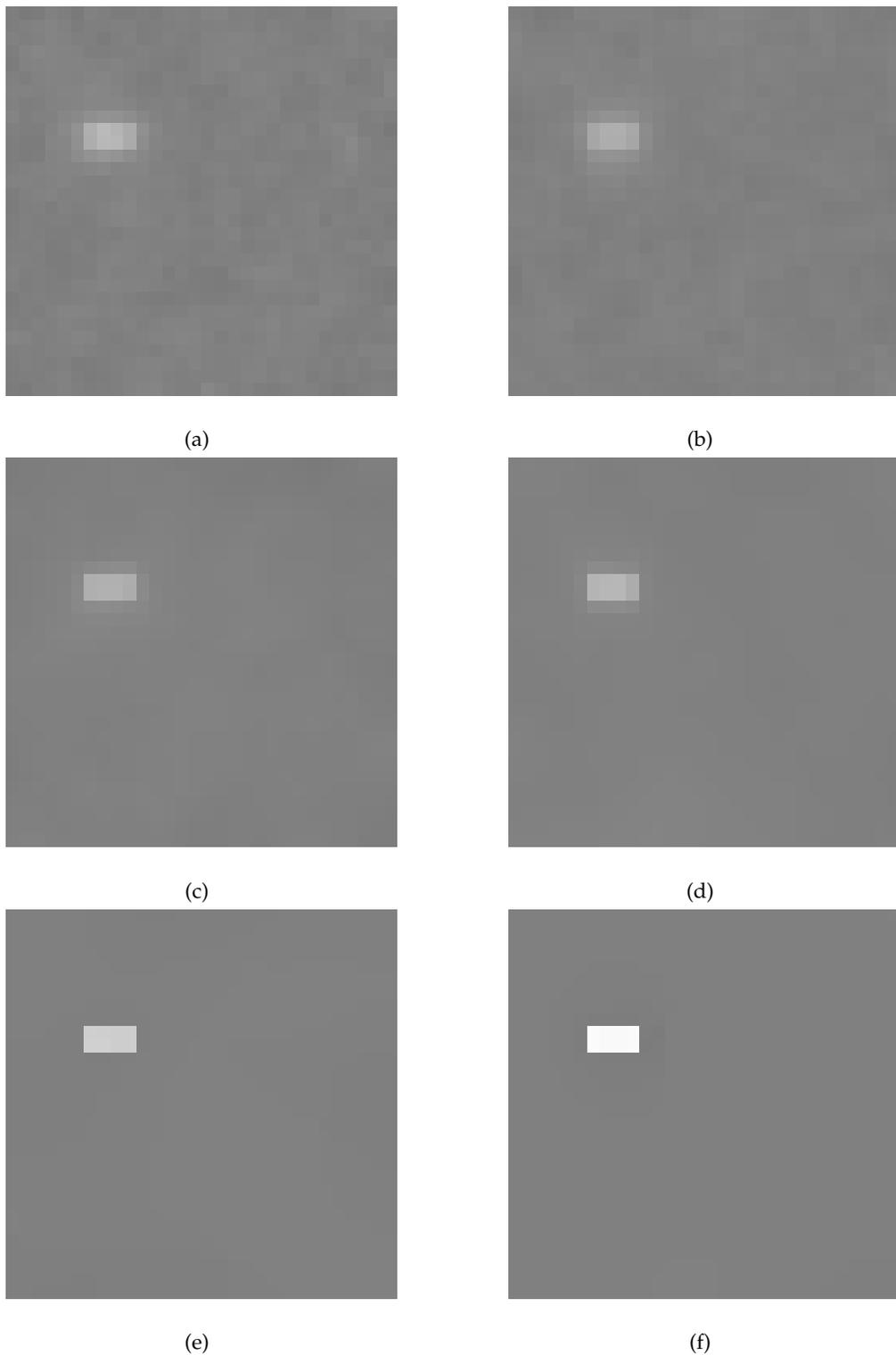
El siguiente valor que se va a considerar es el de  $\mu$  que representa el parámetro de fidelidad a los datos de la función, es decir, cuánto se parecen la imagen original y la imagen después de aplicarle la Variación Total. Se fijan  $\lambda = 10$  y  $varianza = 0,001$  Los valores que va a tomar son  $\mu = [0.05, 0.1, 0.3, 0.5, 1, 1.2]$  y se obtienen las gráficas anteriores pero según este nuevo parámetro. En la Figura 3.6 se obtienen las gráficas de PSNR, SSIM y Ratio de Correlación en función de  $\mu$

En este caso, de nuevo todas las imágenes a las que se le aplica el ruido van a ser iguales. El resultado será el mismo que cuando se variaba el parámetro  $\lambda$  y se muestra en la Figura 3.7

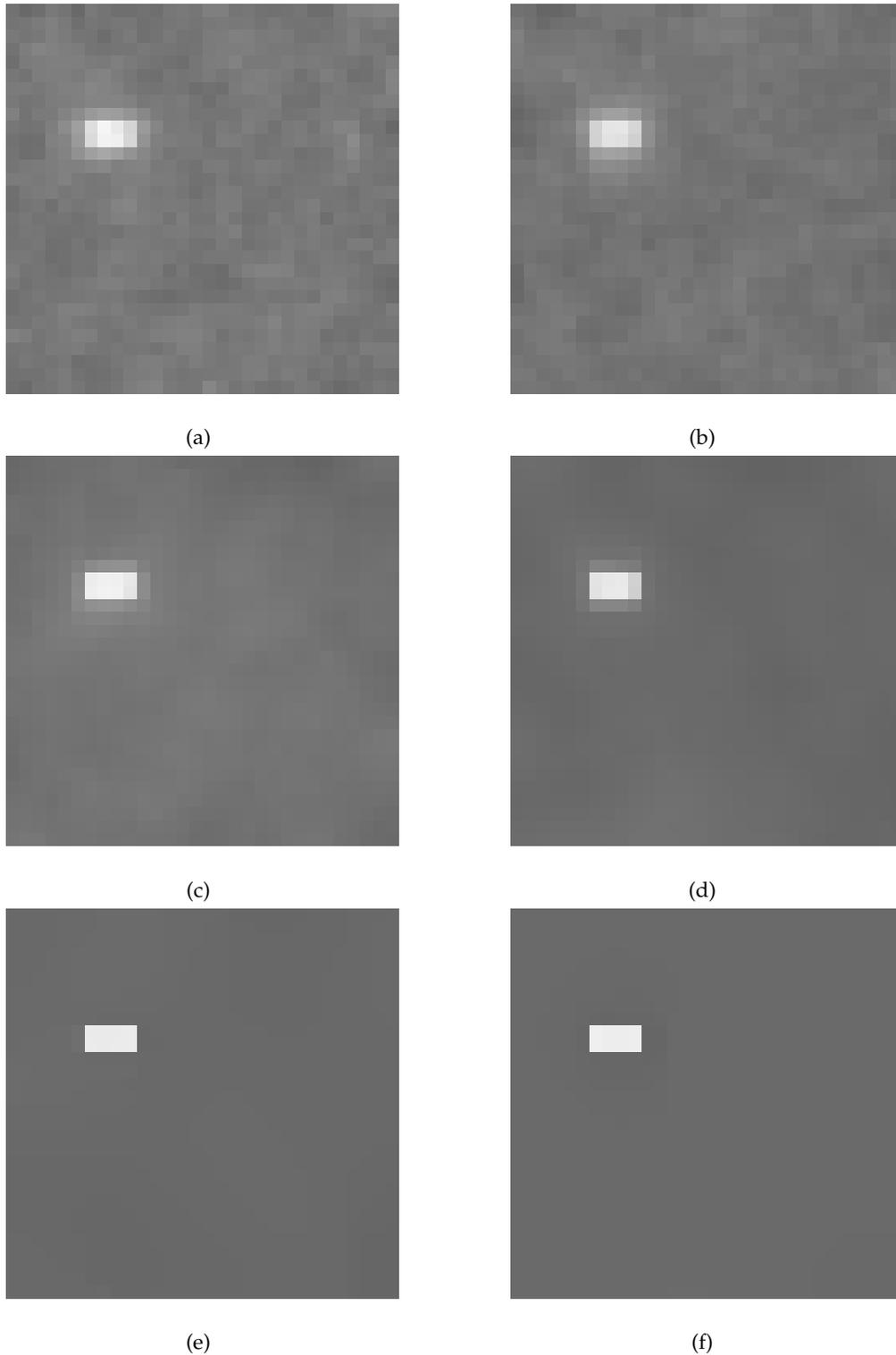
Las distintas imágenes tras aplicar la Variación Total se aprecian ahora en la Figura 3.8 y las normalizadas en la Figura 3.9



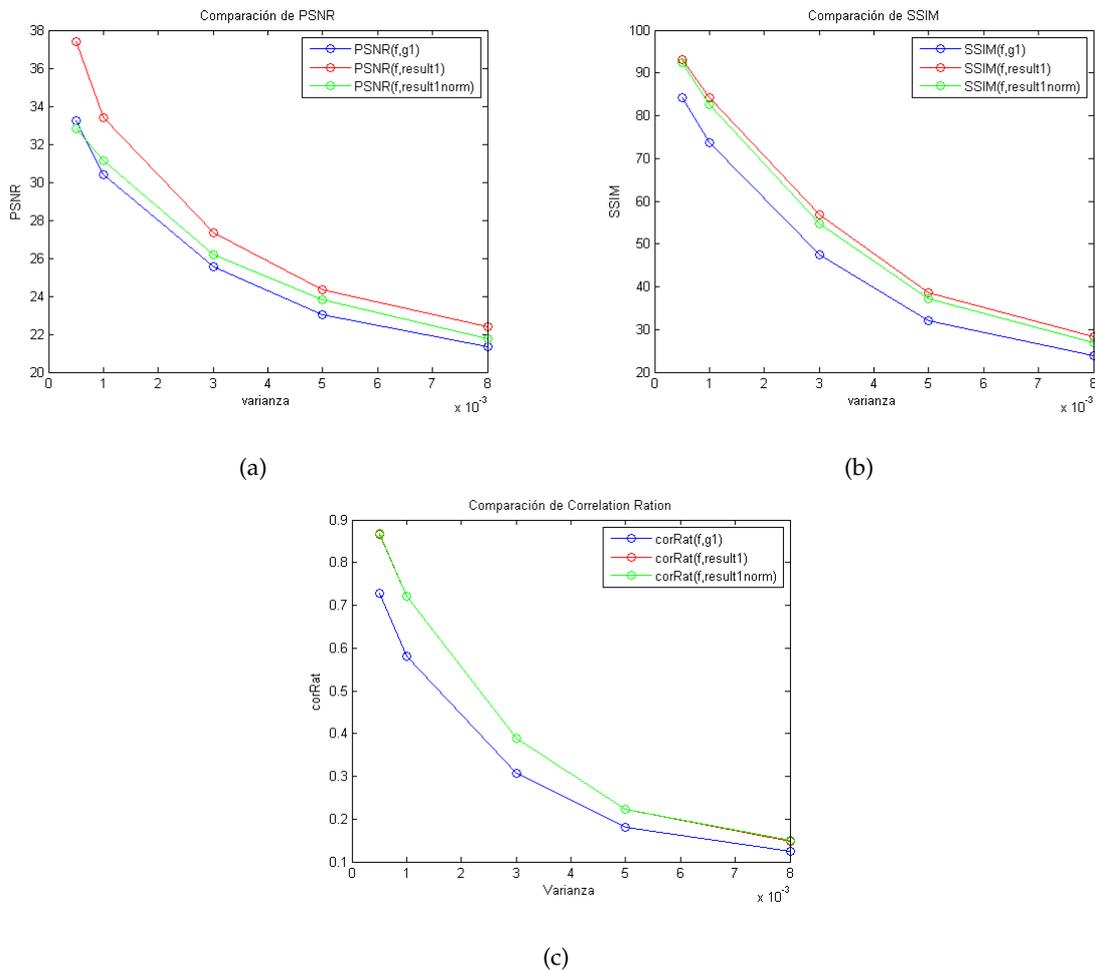
**Figura 3.7:** Imagen con ruido según el valor de  $\mu$ .



**Figura 3.8:** Imágenes con Variación Total según el valor de  $\mu$ . (a)  $\mu = 0,05$  y 3 iteraciones , (b)  $\mu = 0,1$  y 8 iteraciones, (c)  $\mu = 0,3$  y 9 iteraciones, (d)  $\mu = 0,5$  y 8 iteraciones, (e)  $\mu = 1$  y 10 iteraciones, (f)  $\mu = 1,2$  y 14 iteraciones.



**Figura 3.9:** Imágenes normalizadas con Variación Total según el valor de  $\lambda$ . (a)  $\mu = 0,05$  y 3 iteraciones, (b)  $\mu = 0,1$  y 8 iteraciones, (c)  $\mu = 0,3$  y 9 iteraciones, (d)  $\mu = 0,5$  y 8 iteraciones, (e)  $\mu = 1$  y 10 iteraciones, (f)  $\mu = 1,2$  y 14 iteraciones.

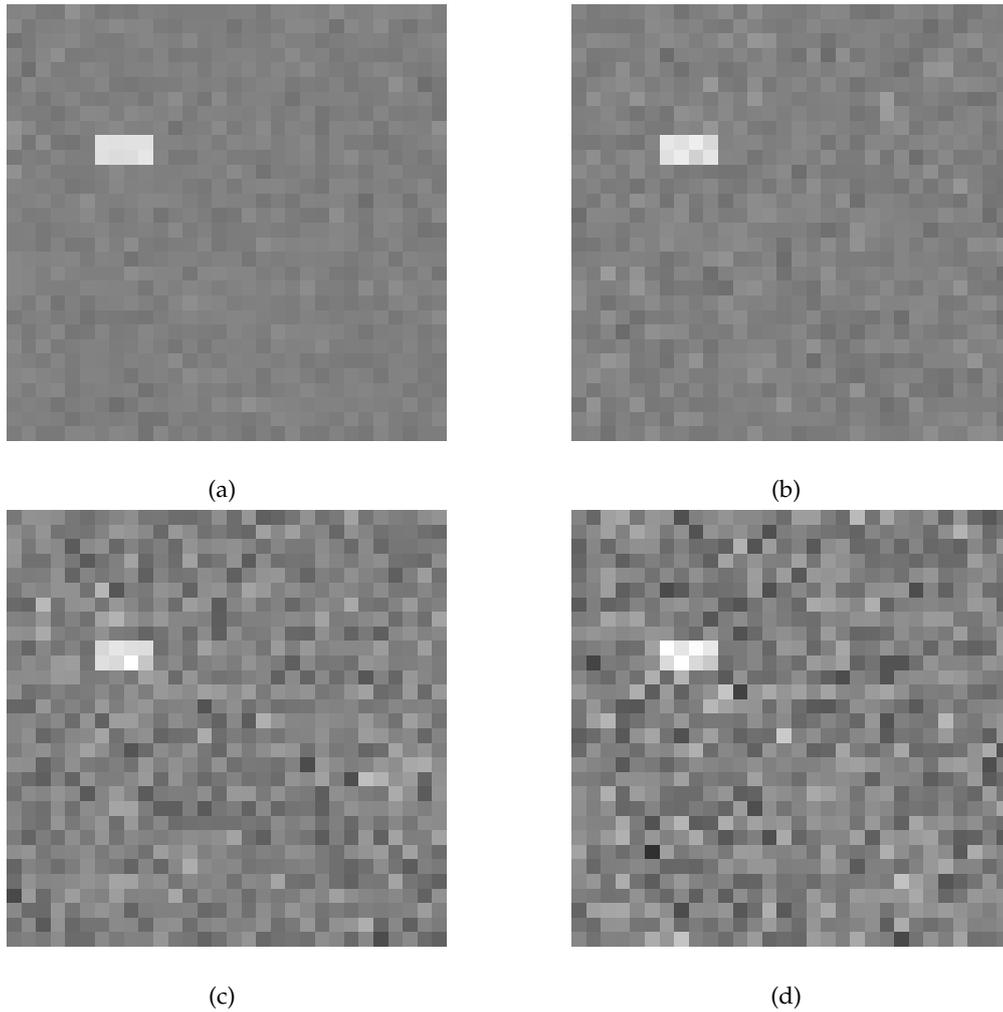


**Figura 3.10:** Variación de los valores según la varianza respecto a la imagen original con  $\mu = 1$ . (a) PSNR, (b) SSIM y (c) Ratio de correlación.

Finalmente se trabaja con el parámetro varianza, que es el que indica la cantidad de ruido Gaussiano que introducimos en las imágenes. Para ello se establece el valor de  $\mu = 1$  y se toman hacen las pruebas para dos valores distintos de  $\lambda$ , 1 y 10, para así poder apreciar mejor el hecho de que a mayor ruido, mayor tendrá que ser el valor de  $\mu$  para mejorar los resultados finales. Para observar estos cambios, la varianza va a tomar los siguientes valores  $varianza = [0.0005, 0.001, 0.005, 0.008]$ .

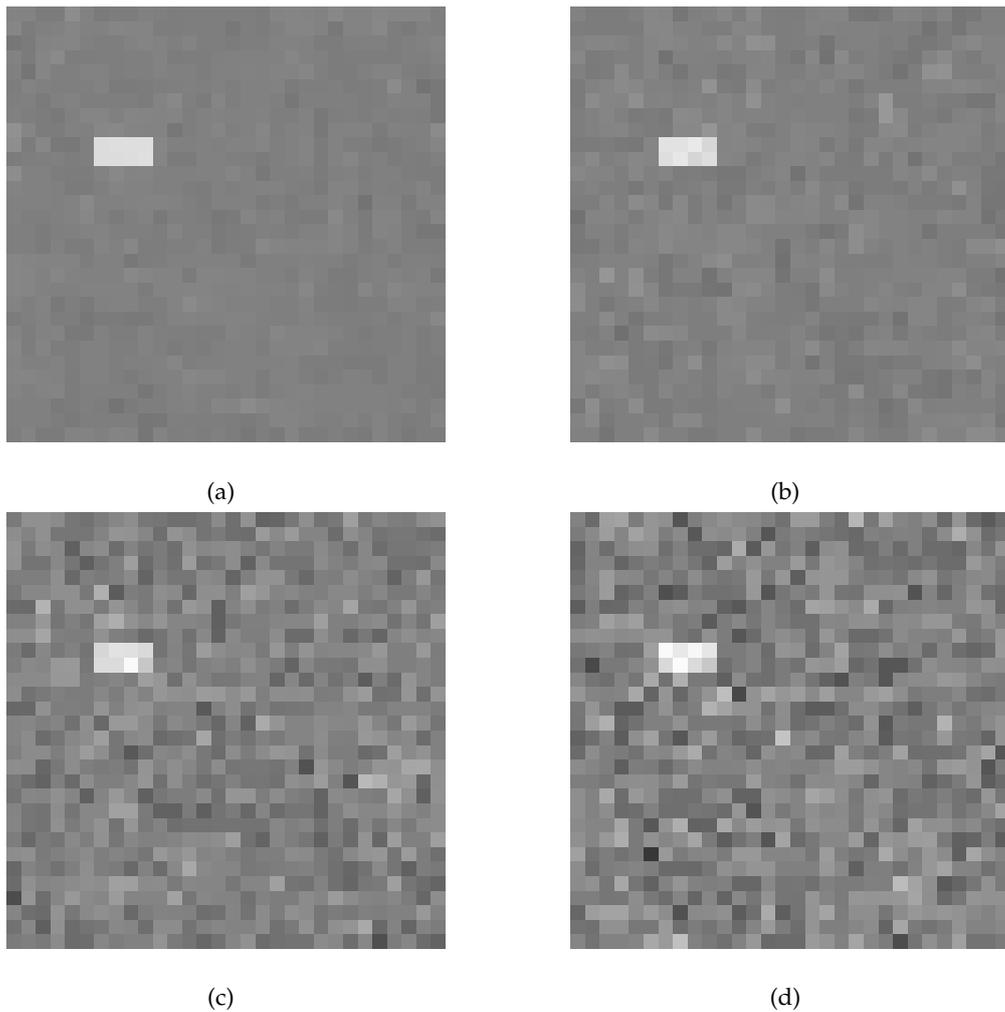
Se empieza estableciendo el valor  $\lambda = 1$  y de nuevo, primero se obtienen las gráficas que relacionan PSNR, SSIM y el Ratio de Correlación con la varianza en la Figura 3.10

A continuación se representan las imágenes con ruido en la Figura 3.11, que esta vez sí que variarán ya que al modificar el valor de la varianza se modifica el ruido que se añade a cada imagen.

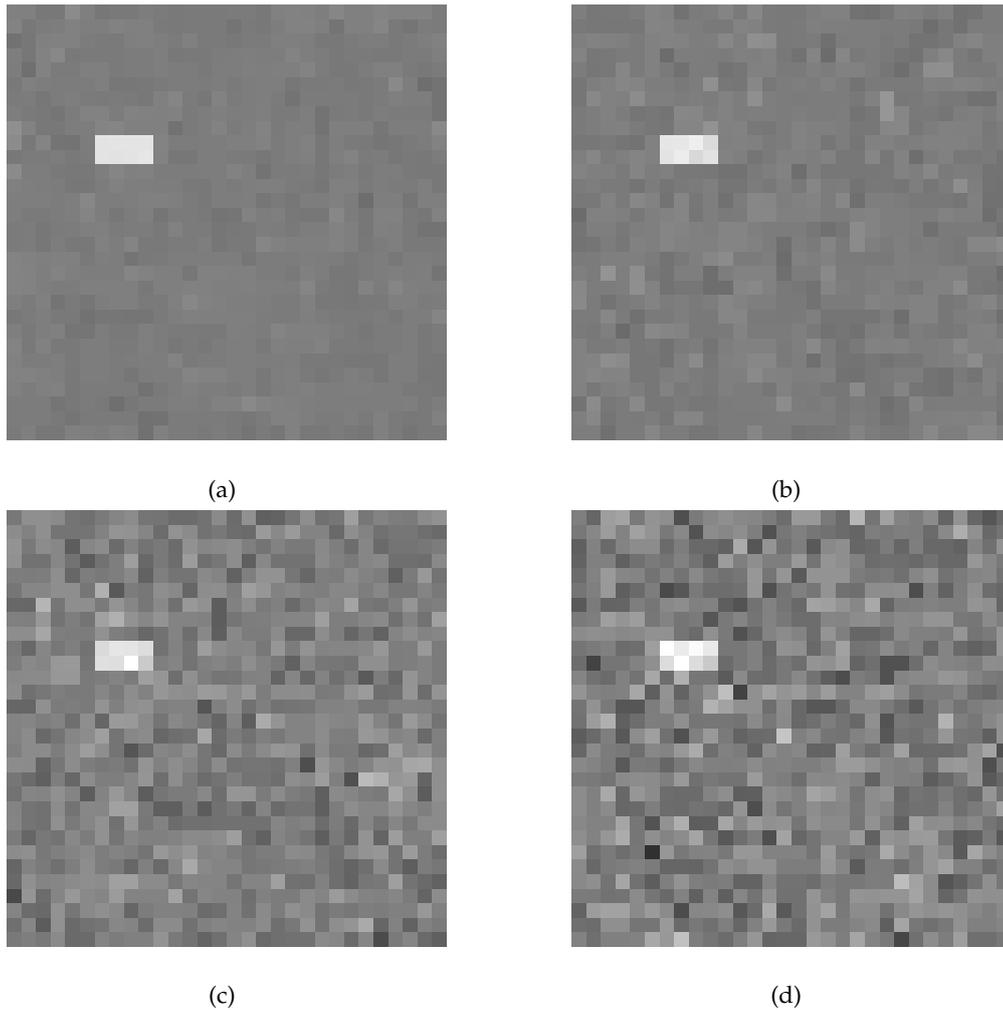


**Figura 3.11:** Imágenes con ruido según el valor de la varianza para  $\lambda = 1$ . (a)varianza=0.0005 , (b)varianza=0.001, (c)varianza=0.005, (d)varianza=0.008.

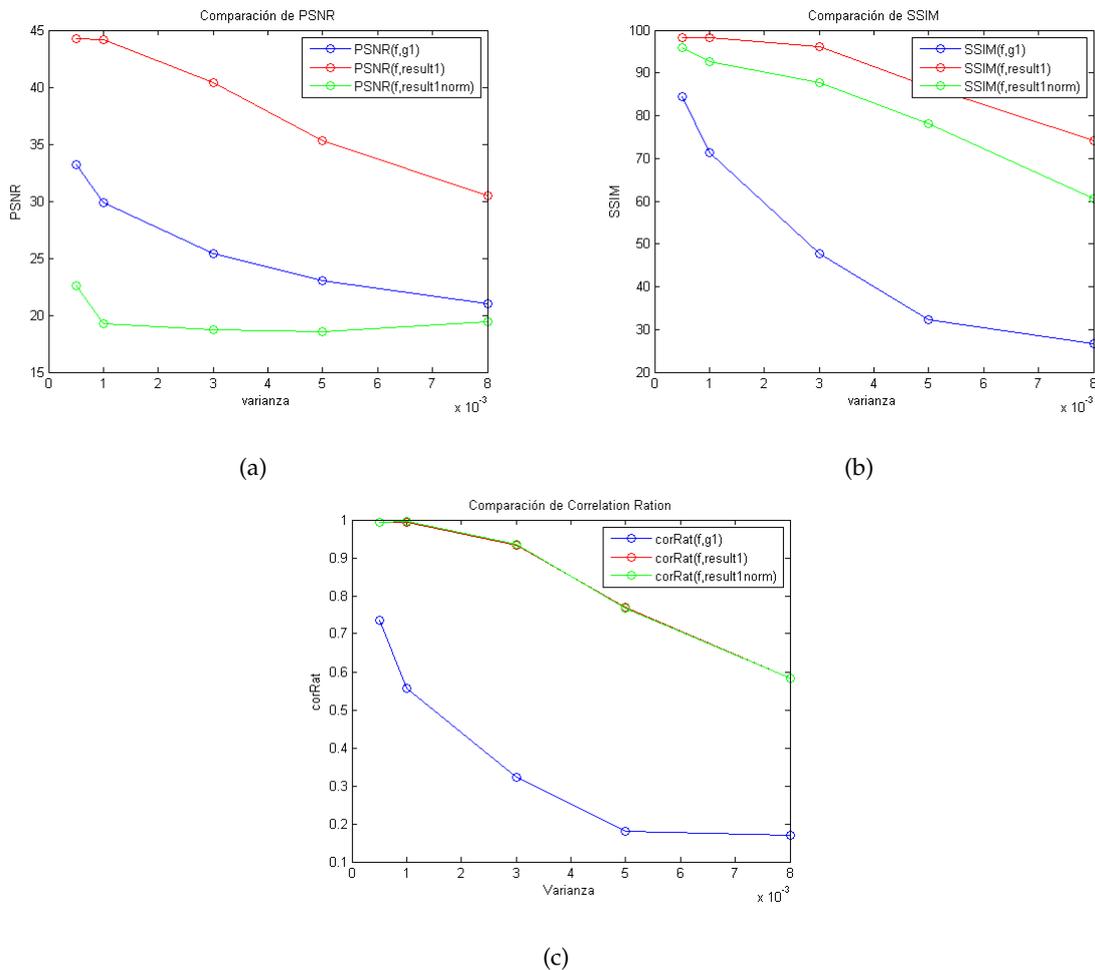
En la Figura 3.12 se muestran las imágenes a las que se les ha aplicado Variación Total y las normalizadas se muestran en la Figura 3.13



**Figura 3.12:** Imágenes con Variación Total según el valor de la varianza para  $\lambda = 1$ . (a)varianza=0.0005 y 12 iteraciones , (b)varianza=0.001 y 13 iteraciones, (c)varianza=0.005 y 18 iteraciones, (d)varianza=0.008 19 iteraciones.



**Figura 3.13:** Imágenes normalizadas con Variación Total según el valor de la varianza para  $\lambda = 1$ . (a)varianza=0.0005 y 12 iteraciones , (b)varianza=0.001 y 13 iteraciones, (c)varianza=0.005 y 18 iteraciones, (d)varianza=0.008 19 iteraciones.

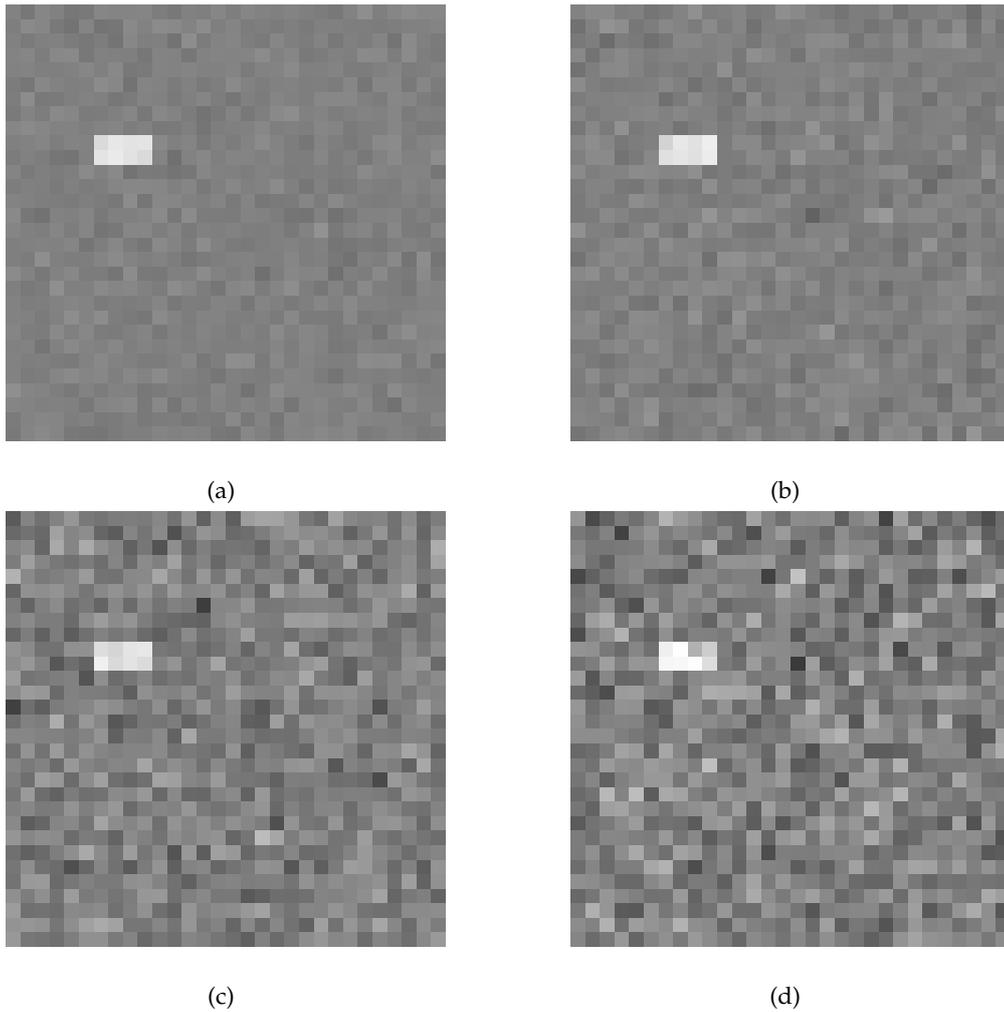


**Figura 3.14:** Variación de los valores según la varianza respecto a la imagen original con  $\lambda = 10$ . (a) PSNR, (b) SSIM y (c) Ratio de correlación.

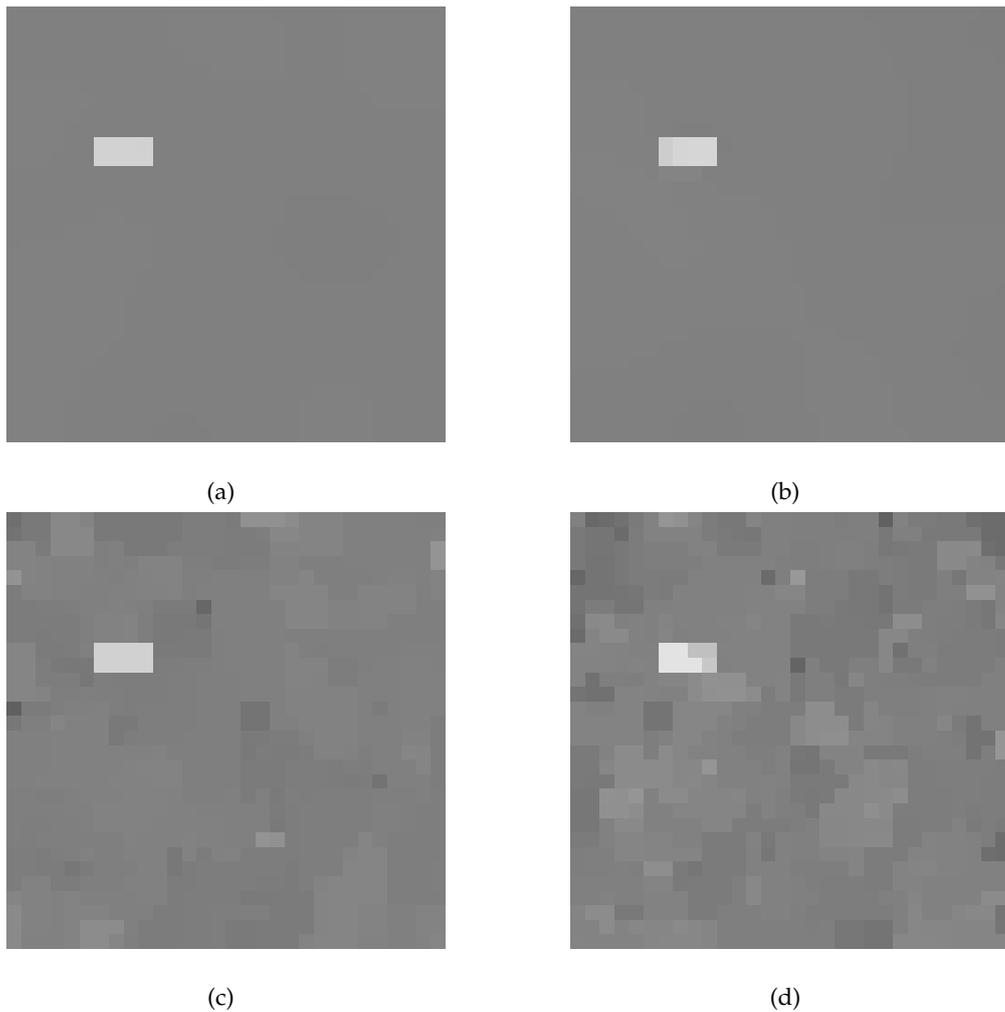
Por último se establece  $\lambda = 10$  y se realiza el estudio con los valores de la varianza establecidos previamente. Se obtienen las gráficas que relacionan PSNR, SSIM y el Ratio de Correlación con la varianza en la Figura 3.14:

En la Figura 3.15 se representan las imágenes con ruido, que de nuevo variarán al modificar el valor de la varianza.

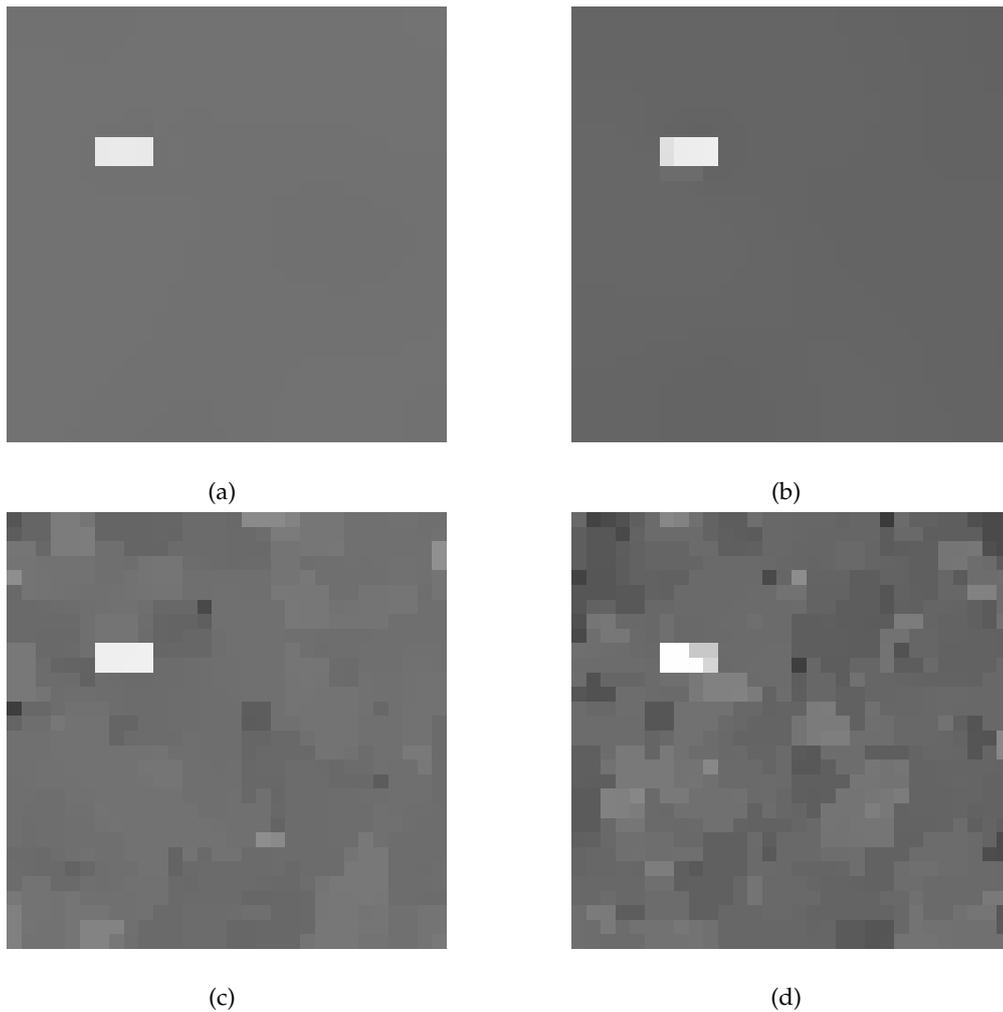
Se representan las imágenes a las que se les ha aplicado Variación Total en la Figura 3.16 y en la Figura 3.17 se muestran las imágenes normalizadas



**Figura 3.15:** Imágenes con ruido según el valor de la varianza para  $\lambda = 10$ . (a)varianza=0.0005 , (b)varianza=0.001, (c)varianza=0.005, (d)varianza=0.008.



**Figura 3.16:** Imágenes con Variación Total según el valor de la varianza para  $\lambda = 10$ . (a)varianza=0.0005 y 11 iteraciones , (b)varianza=0.001 y 11 iteraciones, (c)varianza=0.005 y 12 iteraciones, (d)varianza=0.008 y 13 iteraciones.



**Figura 3.17:** Imágenes normalizadas con Variación Total según el valor de la varianza para  $\lambda = 10$ . (a)varianza=0.0005 y 11 iteraciones , (b)varianza=0.001 y 11 iteraciones, (c)varianza=0.005 y 12 iteraciones, (d)varianza=0.008 y 13 iteraciones.

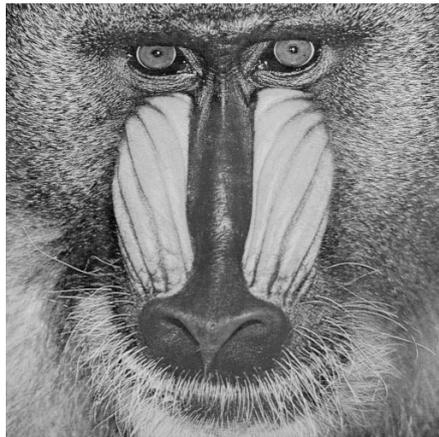


Figura 3.18: Imagen original utilizada para las pruebas.

## 3.2. Efecto en las texturas

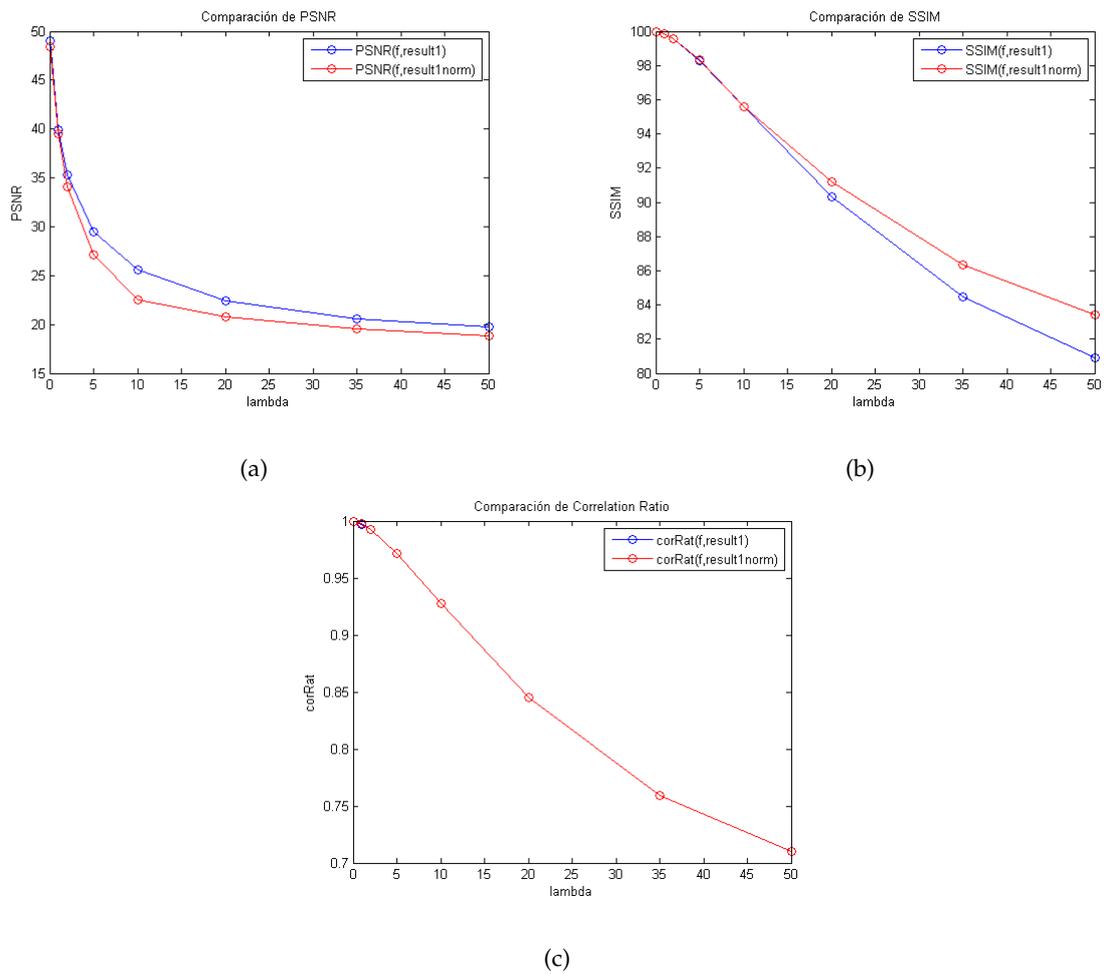
La técnica de la Variación Total elimina los detalles no deseados conservando los detalles importantes como, por ejemplo, los bordes. En ocasiones esto puede suponer que se pierda la percepción de las distintas texturas en una determinada imagen. Para estudiar este hecho se va a aplicar la Variación Total Anisotrópica a la imagen de la Figura 3.18 variando los parámetros de regularización y fidelización.

Se empieza variando el parámetro de regularización  $\lambda$  para ver el efecto que esto tiene en las texturas de la imagen. Fijando el valor del parámetro de fidelización  $\mu = 1$ , le damos a  $\lambda$  los valores  $\lambda = [0.1, 2, 5, 10, 35, 50]$  y obtenemos los resultados para la PSNR, SSIM y Ratio de Correlación en la Figura 3.19.

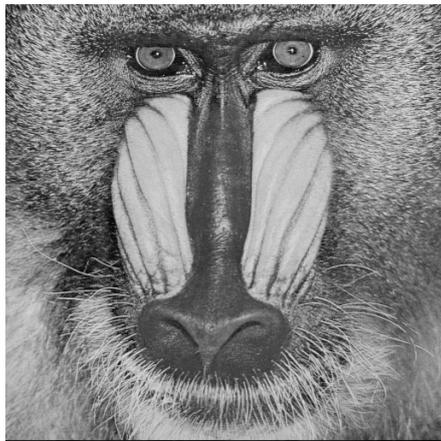
Se aprecia que a medida que aumenta el valor del parámetro de regularización empeoran los tres parámetros si los medimos respecto a la imagen original.

Si vemos las imágenes a las que se les ha aplicado la Variación Total en la Figura 3.20, se observa que a medida que aumenta el valor de  $\lambda$  la imagen se suaviza haciendo perder las texturas. Además cuanto más se suaviza la imagen se va a necesitar un mayor número de iteraciones en el proceso.

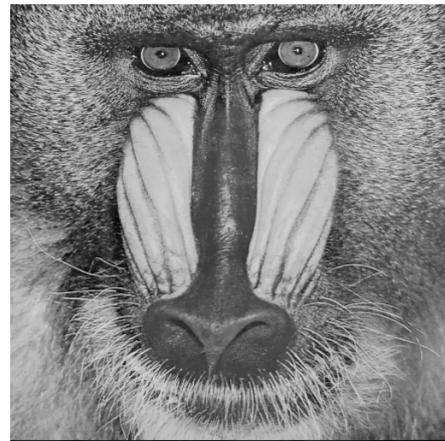
Lo mismo se aprecia al normalizar las imágenes a las que se les ha aplicado Variación Total en la Figura 3.21.



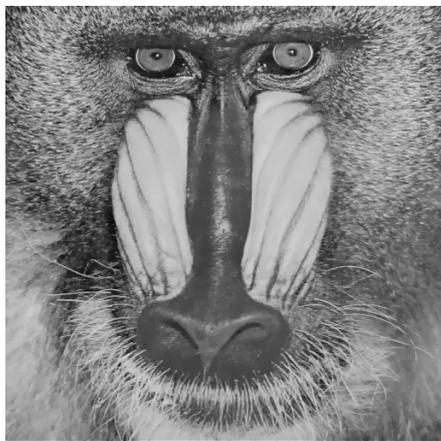
**Figura 3.19:** Variación de los valores según  $\lambda$  respecto a la imagen original (a) PSNR, (b) SSIM y (c) Ratio de correlación.



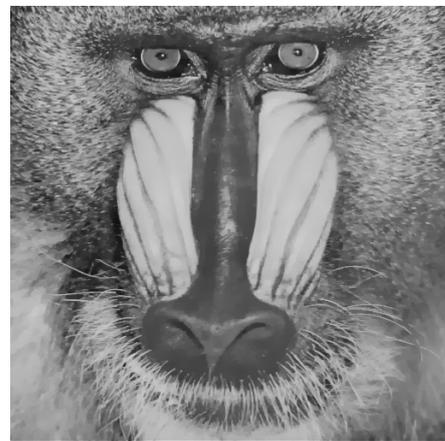
(a)



(b)



(c)



(d)

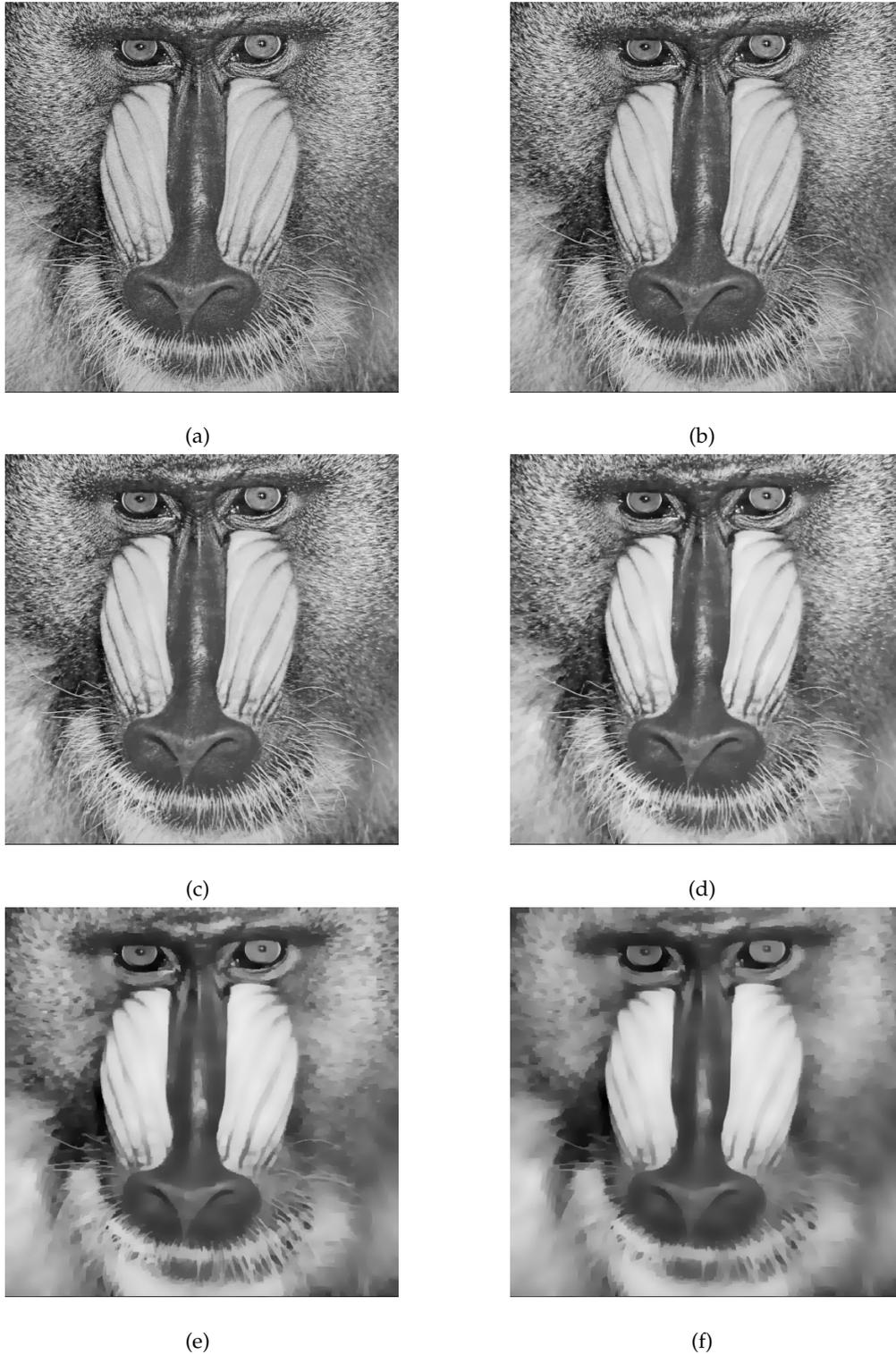


(e)

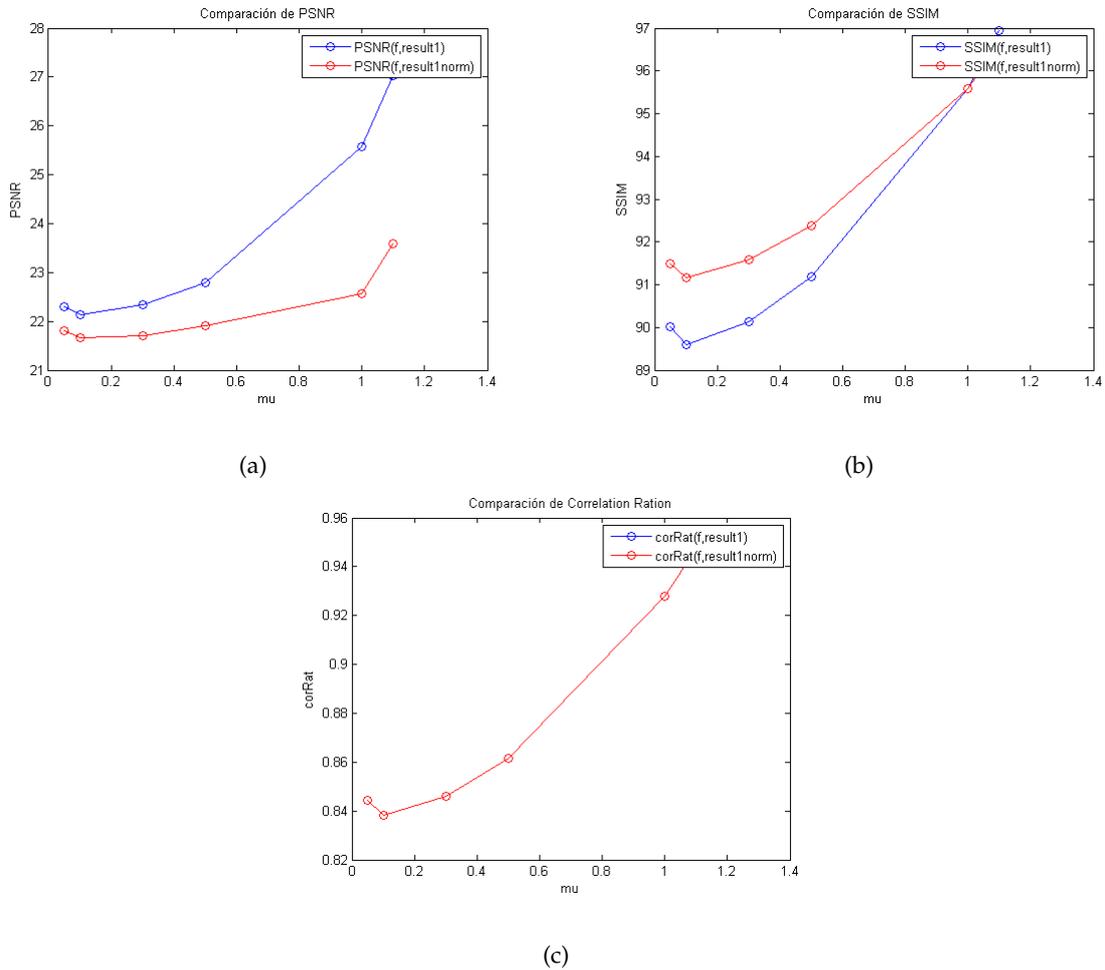


(f)

**Figura 3.20:** Imágenes con Variación Total según el valor de  $\lambda$ . (a)  $\lambda = 0,1$  y 17 iteraciones , (b)  $\lambda = 2$  y 16 iteraciones, (c)  $\lambda = 5$  y 15 iteraciones, (d)  $\lambda = 10$  y 14 iteraciones, (e)  $\lambda = 35$  y 20 iteraciones, (f)  $\lambda = 50$  y 24 iteraciones.



**Figura 3.21:** Imágenes normalizadas con Variación Total según el valor de  $\lambda$ . (a)  $\lambda = 0,1$  y 17 iteraciones, (b)  $\lambda = 2$  y 16 iteraciones, (c)  $\lambda = 5$  y 15 iteraciones, (d)  $\lambda = 10$  y 14 iteraciones, (e)  $\lambda = 35$  y 20 iteraciones, (f)  $\lambda = 50$  y 24 iteraciones.

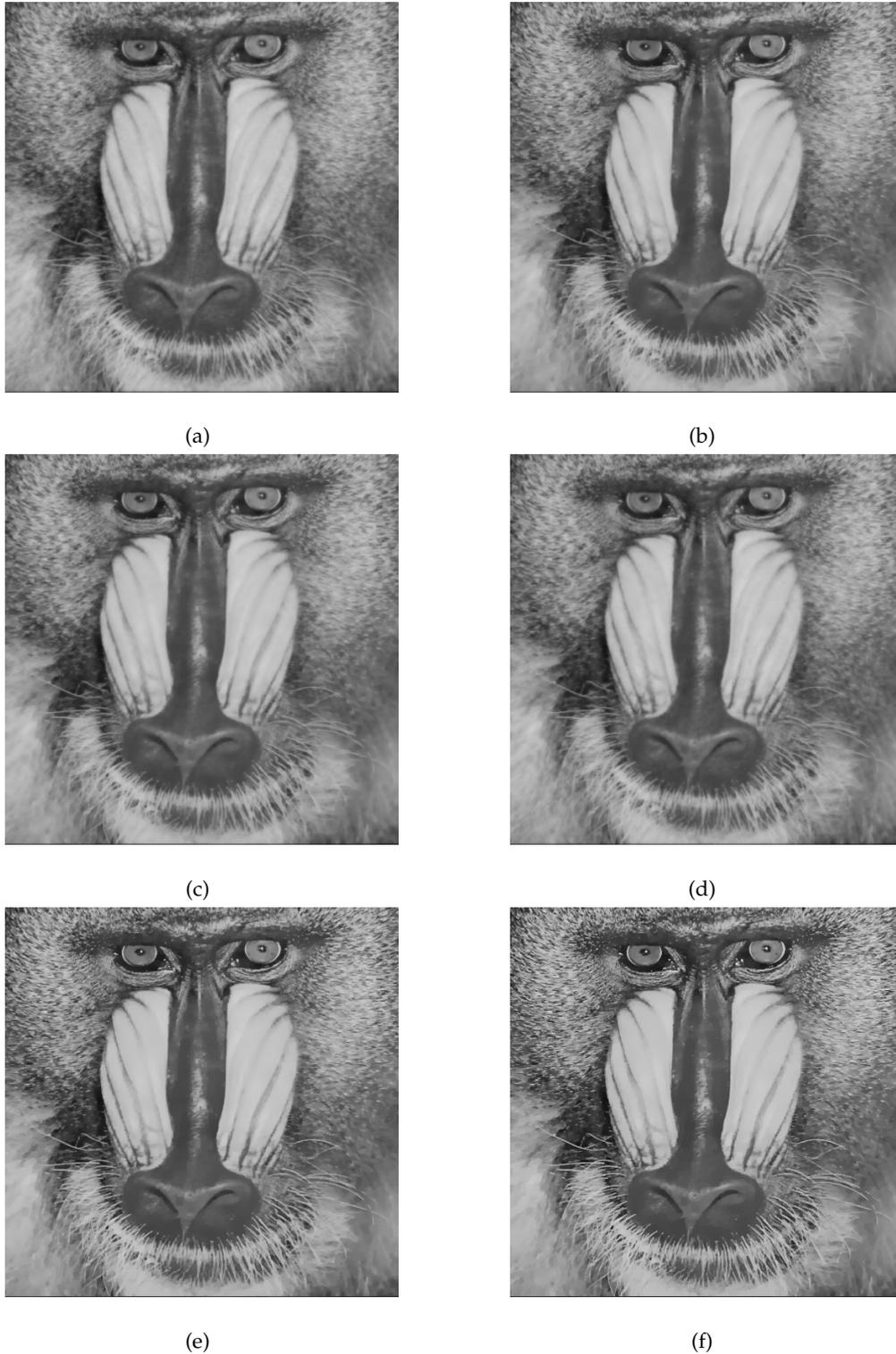


**Figura 3.22:** Variación de los valores según  $\mu$  respecto a la imagen original (a) PSNR, (b) SSIM y (c) Ratio de correlación.

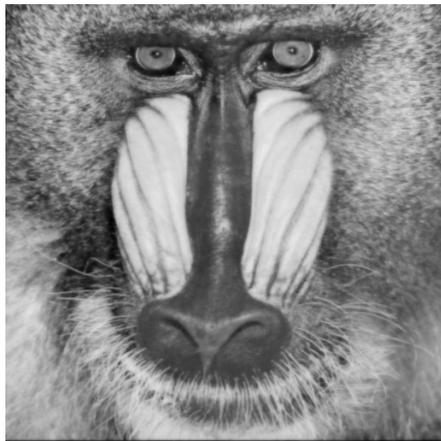
En el caso de que se varíe el parámetro de fidelización  $\mu$  manteniendo  $\lambda = 10$  y dándole los valores  $\mu = [0.05, 0.1, 0.3, 0.5, 1, 1.2]$ , se obtienen los resultados de PSNR, SSIM y Ratio de Correlación en la Figura 3.22, donde se observa que los tres parámetros mejoran al aumentar el valor de  $\mu$ .

En la Figura 3.23 se muestra el efecto de la Variación Total según el valor del parámetro de fidelización y es apreciable que las texturas se pueden ver mejor a medida que aumenta el valor de  $\mu$  ya que cuanto mayor sea este parámetro más se parecerá la imagen final a la original, con sus correspondientes texturas. Además a mayor  $\mu$ , mayor es el número de iteraciones necesarias para llevar a cabo la aplicación de la Variación Total.

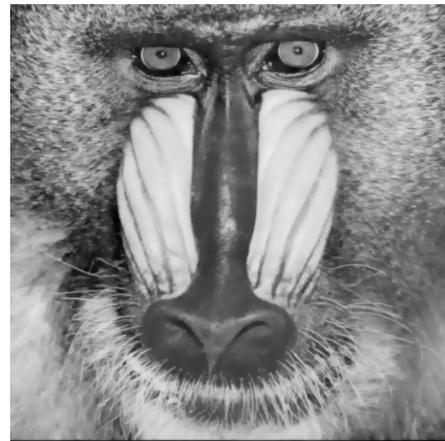
Esto mismo ocurre al normalizar las imágenes anteriores, como se aprecia en la Figura 3.24.



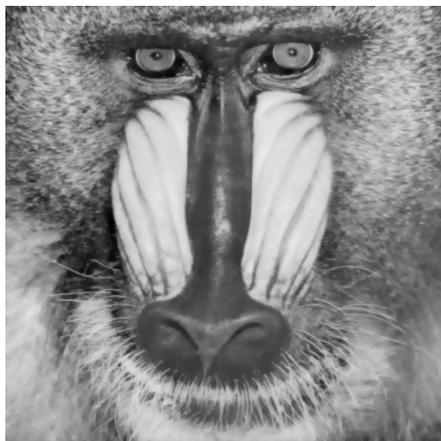
**Figura 3.23:** Imágenes con Variación Total según el valor de  $\mu$ . (a)  $\mu = 0,051$  y 15 iteraciones, (b)  $\mu = 0,5$  y 10 iteraciones, (c)  $\mu = 0,3$  y 10 iteraciones, (d)  $\mu = 0,1$  y 17 iteraciones, (e)  $\mu = 1$  y 14 iteraciones, (f)  $\mu = 1,1$  y 19 iteraciones.



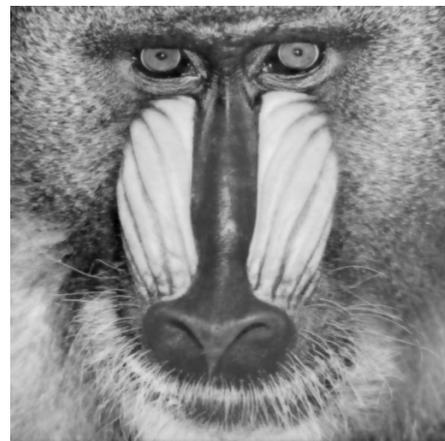
(a)



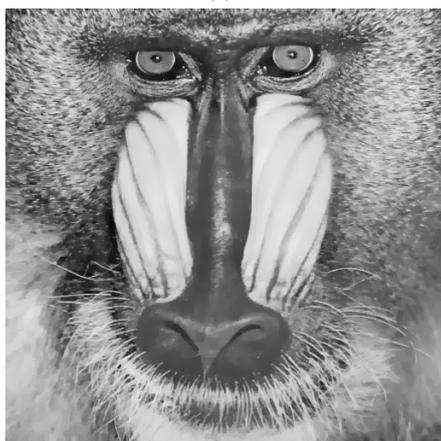
(b)



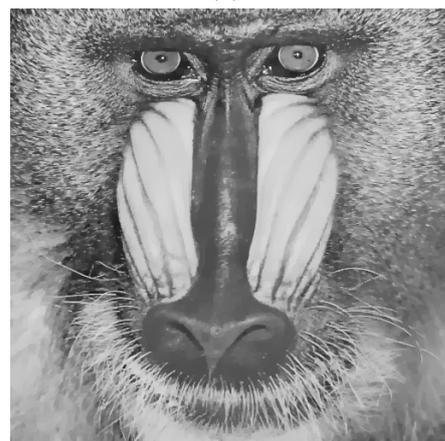
(c)



(d)



(e)



(f)

**Figura 3.24:** Imágenes normalizadas con Variación Total según el valor de  $\mu$ . (a)  $\mu = 0,051$  y 15 iteraciones, (b)  $\mu = 0,5$  y 10 iteraciones, (c)  $\mu = 0,3$  y 10 iteraciones, (d)  $\mu = 0,1$  y 17 iteraciones, (e)  $\mu = 1$  y 14 iteraciones, (f)  $\mu = 1,1$  y 19 iteraciones.

---

### Pruebas en imágenes retinográficas

---

A lo largo de este capítulo se van a realizar las pruebas anteriores a imágenes retinográficas, que son el objetivo final de estudio de este trabajo.

Con un parámetro de fidelización  $\mu = 1$  y un parámetro de regularización  $\lambda = 10$ , valores que, como se observó en apartados anteriores, son adecuados para suavizar el ruido de las imágenes, se procesan retinografías de ojos distintos procedentes de una base de datos en la que las imágenes ya aparecen suficientemente claras pese al ruido presente.

En la Figura 4.1 se pueden observar las imágenes originales de la base de datos DRIVE<sup>1</sup> [23] y las imágenes tras aplicarles variación total en las que se observa la clara mejora al verse reducido el ruido.

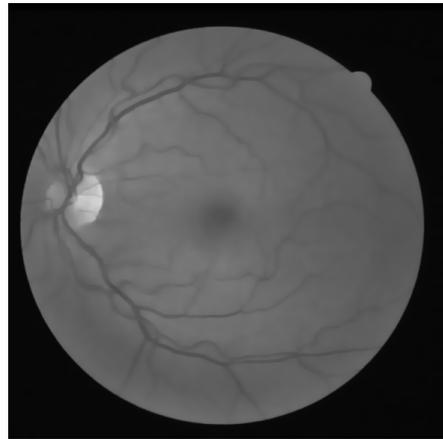
Las imágenes que se se observan en la Figura 4.2, cedidas por la doctora Inmaculada Sellés, del Servicio de Oftalmología del Hospital Reina Sofía, muestran una fotografía del ojo derecho y el ojo izquierdo de un paciente.

---

<sup>1</sup><http://www.isi.uu.nl/Research/Databases/DRIVE/>



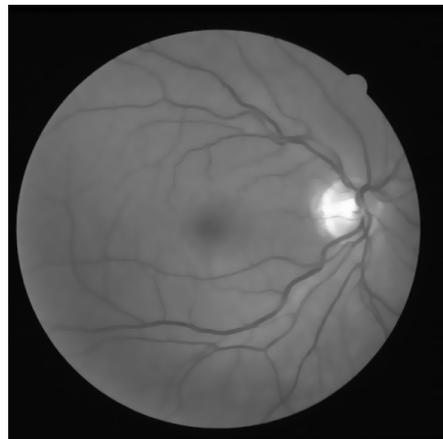
(a)



(b)



(c)



(d)

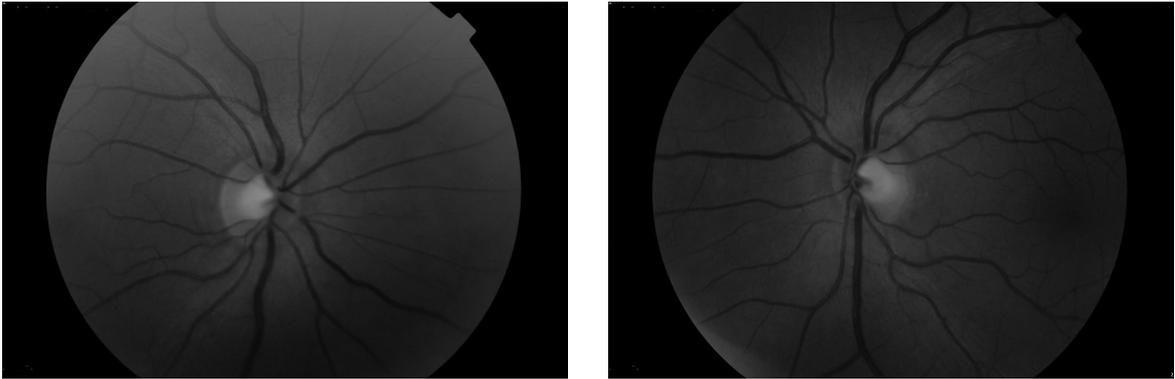


(e)



(f)

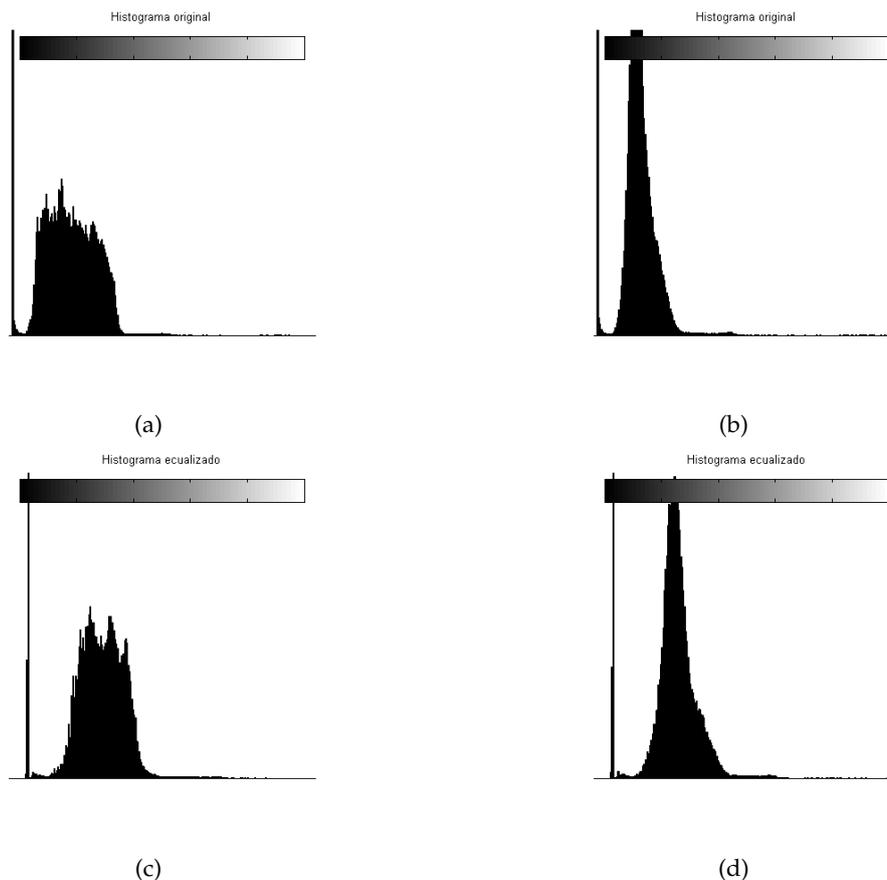
**Figura 4.1:** Imágenes retinográficas. (a), (c) y (e) Imágenes originales, (b), (d) y (f) Imágenes con variación total.



(a)

(b)

**Figura 4.2:** Imágenes originales de retinografías. (a) Ojo derecho , (b) Ojo izquierdo.



**Figura 4.3:** Representación de histogramas. (a) Histograma original del ojo derecho , (b) Histograma original del ojo izquierdo, (c) Ecualización de histograma del ojo derecho, (d) Ecualización de histograma del ojo izquierdo.

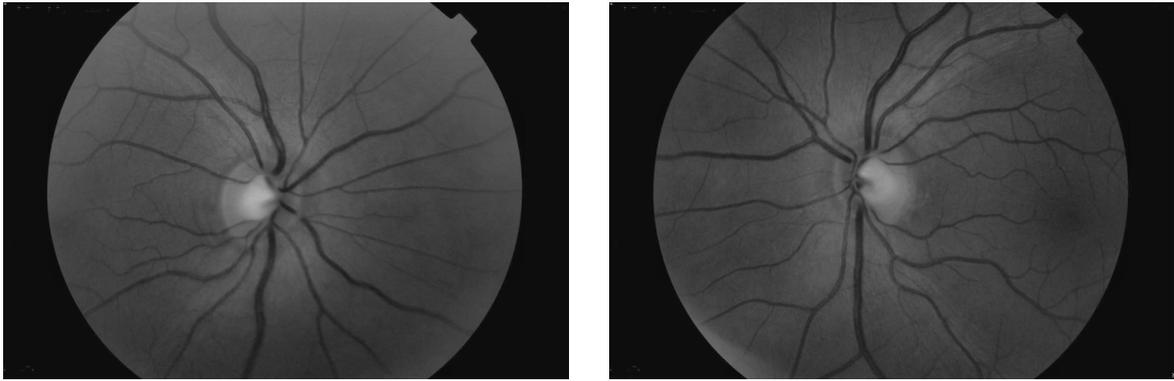
La oscuridad que presentan estas imágenes hace complicada la percepción de detalles que podrían ser de especial relevancia a la hora de un diagnóstico. Por ello, antes de aplicarles variación total se las va a tratar con una ecualización de histograma.

Este proceso mejora el contraste de la imagen repartiendo de forma más o menos uniforme los valores del histograma, como se puede apreciar en las imágenes de la Figura 4.3.

En este caso, ya que únicamente se necesitaba iluminar un poco las imágenes para facilitar la apreciación de las mismas, se observa que los histogramas de las retinografías han sufrido variaciones mínimas.

Las imágenes retinográficas finales tras realizarles la ecualización de histograma son las que se muestran en la Figura 4.4

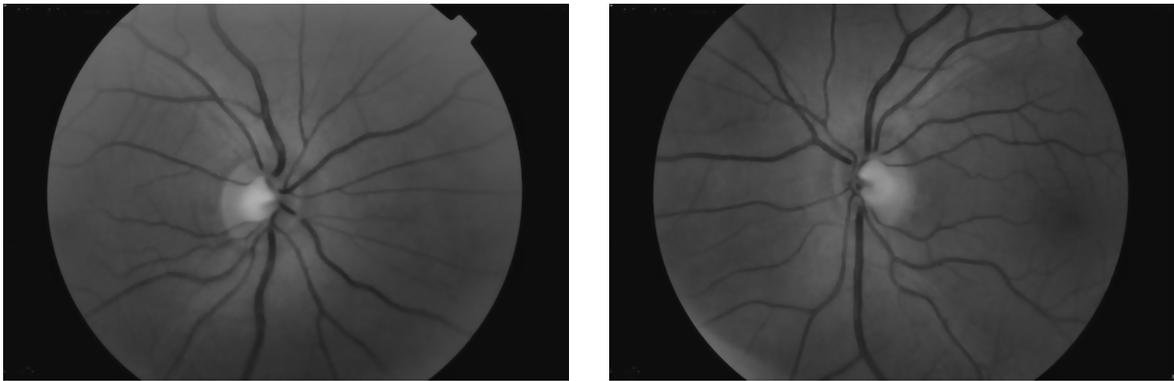
Una vez preparadas las imágenes, el siguiente paso que se dará será aplicarles variación



(a)

(b)

**Figura 4.4:** Imágenes de retinografías iluminadas. (a) Ojo derecho , (b) Ojo izquierdo.



(a)

(b)

**Figura 4.5:** Imágenes retinográficas a las que se les ha aplicado variación total. (a) Ojo derecho , (b) Ojo izquierdo.

total. El resultado de este proceso se muestra en la Figura 4.5 y para obtener estos resultados se ha utilizado un valor del parámetro de regularización  $\lambda = 10$  y un valor del parámetro de fidelización  $\mu = 1$ .

Como se puede observar en las imágenes con variación total, el ruido existente en la retinografía ha disminuido notablemente hasta permitir la fácil apreciación de detalles relevantes como podrían ser las venas o la papila óptica.



---

### Conclusiones

---

Para acabar este trabajo, cabe destacar que en la realización del mismo todas las pruebas se han llevado a cabo para una disminución del ruido con variación total anisotrópica. En teoría, el caso isotrópico debería devolver resultados más precisos que el anisotrópico, sin embargo debido al tipo de imágenes que se utilizan y al ruido que hay presente en ellas, las diferencias entre los resultados para los dos casos son imperceptibles para el ojo humano. Además las gráficas que son devueltas para los cálculos de PSNR, SSIM y ratio de correlación según los distintos parámetros de la variación total también son iguales. Por tanto, se escoge el modelo anisotrópico porque optimiza el tiempo de procesado de las imágenes.

Inicialmente, en este trabajo se pretendía haber reducido ese tiempo de procesado creando un nuevo código para llevar a cabo la disminución del ruido por variación total. El código que se ha utilizado, basado en las publicaciones [12] y [10] procesa la imagen de una manera unidimensional, con transformaciones de la matriz que compone la imagen. La idea original fue conseguir programar en Matlab el algoritmo que se proporciona en [12] para que la imagen fuese procesada en dos dimensiones, y de este modo, reducir el tiempo de ejecución. Lamentablemente, debido a diversos problemas que han ido surgiendo tanto de tiempo como de programación, esto no ha sido posible.

Respecto a la detección comprimida que se presenta a lo largo de la memoria, cabe decir que no ha sido utilizada para las pruebas pero que podría ser una futura línea de estudio por sus

aplicaciones para la reconstrucción de imagen médica.

En cuanto al tema principal tratado, el procesado de retinografías mediante regularización por variación total, se encuentra que las imágenes procesadas suponen una importante mejora respecto a las originales. A falta de una opinión médica experta, se cree que la reducción de ruido que presentan, haría más fácil la detección y el diagnóstico de enfermedades relacionadas con la retina.

---

## Bibliografía

---

- [1] Richard Baraniuk. Compressive sensing. *IEEE Signal Processing Mag*, pages 118–120, 2007.
- [2] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.
- [3] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11):1222–1239, November 2001.
- [4] L. M. Bregman. The Relaxation Method of Finding the Common Point of Convex Sets and Its Application to the Solution of Problems in Convex Programming. *USSR Computational Mathematics and Mathematical Physics*, 7:200–217, 1967.
- [5] E. J. Candes and Romberg. Signal recovery from random projections. *Proceesings of SPIE Computational Imaging III*, 2005.
- [6] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theor.*, 52(2):489–509, February 2006.
- [7] Tony F. Chan, Gene H. Golub, and Pep Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM J. Sci. Comput.*, 20(6):1964–1977, May 1999.
- [8] Jérôme Darbon and Marc Sigelle. A fast and exact algorithm for total variation minimization. pages 351–359, 2005.

- [9] David L. Donoho. Compressed sensing. *IEEE Trans. Inform. Theory*, 52:1289–1306, 2006.
- [10] Micchelli et al. Proximity algorithms for image models: denoising inverse problems. 2011.
- [11] Jian feng Cai, Stanley Osher, and Zuowei Shen. Fast linearized bregman iteration for compressed sensing. pages 08–37, 2008.
- [12] Tom Goldstein and Stanley Osher. The split bregman method for l1-regularized problems. *SIAM J. Img. Sci.*, 2(2):323–343, April 2009.
- [13] Elaine T. Hale, Wotao Yin, and Yin Zhang. A Fixed-Point Continuation Method for l1 - Regularized Minimization with Applications to Compressed Sensing. Technical report, Rice University, July 2007.
- [14] Koh-K. Lustig M. Boyd S. Kim, S. and D. Gerinvesky. A method for large-scale l1-regularized least squares problems with applications in signal processing and statistics. *TEch. Report, Dept. of Electrical Engineering*, 2007.
- [15] Y Li and F Santosa. An affine scaling algorithm for minimizing total variation in image enhancement. *TEch Repost 12/94, Center for theory and simulation in science and engineering*, 1994.
- [16] ti-Chiun Chang Lin He and Stanley Osher. Mr image reconstruction from soarse radial samples by using iterative refinement procedures. *Proceedings of the 13th annual meeting of ISMRM*, 2006.
- [17] M. Lustig, D. Donoho, and J. M. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195, 2007.
- [18] M. Lustig, J. H. Lee, D. L. Donoho, and J. M. Pauly. Faster Imaging with Randomly Perturbed, Undersampled Spirals and L1 Reconstruction. In *ISMRM*. ISMRM, 2005.
- [19] J. Nocedal and S. J. Wright. Numerical optimization. 2006.
- [20] Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin. An iterative regularization method for total variation-based image restoration. *Simul*, 4:460–489, 2005.
- [21] Stanley Osher, Yu Mao, Bin Dong, and Wotao Yin. Fast linearized bregman iteration for compressive sensing and sparse denoising. *Commun. Math. Sci.*, 8(1):93–111, 03 2010.

- [22] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. D*, 60(1-4):259–268, November 1992.
- [23] J. Staal, M.D. Abramoff, M. Niemeijer, M.A. Viergever, and B. van Ginneken. Ridge-based vessel segmentation in color images of the retina. *Medical Imaging, IEEE Transactions on*, 23(4):501–509, 2004.
- [24] J. Trzasko, A. Manduca, and E. Borisch. Sparse mri reconstruction via multiscale l0-continuation. pages 176–180, Aug 2007.
- [25] C. R. Vogel. A multigrid method for total variation-based image denoising. 1995.
- [26] C. R. Vogel and M. E. Oman. Iterative methods for total variation denoising. *SIAM J. Sci. Comput.*, 17(1):227–238, January 1996.
- [27] Yilun Wang, Wotao Yin, and Yin Zhang. A fast algorithm for image deblurring with total variation regularization. 2007.
- [28] Wotao Yin, Stanley Osher, Donald Goldfarb, and Jerome Darbon. Bregman iterative algorithms for  $\ell_1$ -minimization with applications to compressed sensing. *SIAM J. Imaging Sci*, pages 143–168, 2008.