Interfases
E-ISSN: 1993-4912
interfases@ulima.edu.pe
Universidad de Lima
Perú

Taquía Gutiérrez, José Antonio
El procesamiento de imágenes y su potencial aplicación en empresas con estrategia digital
Interfases, núm. 10, 2017, pp. 11-29
Universidad de Lima

Disponible en: https://www.redalyc.org/articulo.oa?id=730180369002

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

Sistema de Información Científica
Red de Revistas Científicas de América Latina, el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Recibido: 18 de agosto del 2017
Aprobado: 15 de octubre del 2017

El procesamiento de imágenes y su potencial aplicación en empresas con estrategia digital

José Antonio Taquía Gutiérrez
Jtaquia @ulima.edu.pe
Universidad de Lima, Lima, Perú

Resumen

Hoy más que nunca, el mercado minorista peruano ha convertido la frase "todo pasa por los ojos" en una herramienta competitiva. El diseño y la optimización del espacio, así como la comercialización visual, son técnicas que impactan en la venta. Nuevos conceptos, tales como *omnicanalidad y experiencia de compra*, son alimentados por el análisis de datos para describir el modo comercial; y nuevas fuentes de información cualitativa, entre ellas la teoría del color, ayudan especialmente a comprender y predecir el impacto de las decisiones futuras en el punto de venta. Este artículo describe la utilidad de las técnicas de procesamiento de imágenes para innovar en el mercado minorista, en un esfuerzo por extraer información útil de piezas publicitarias frecuentemente utilizadas en este sector.

Palabras clave: procesamiento de imágenes / marketing para venta al detalle / aprendizaje automático / analítica predictiva

Abstract

Image processing and its potential application in companies with digital strategy

Peruvian retail market today, more than ever, has turned the phrase "everything goes through the eyes" into a competitive tool. The design and optimization of space, as well as visual merchandising, are techniques that impact the sale new concepts such as omnicanality and buying experience are fed by data analytics in order to describe the commercial mode; and new qualitative sources of information, among them color theory, specially help to understand and predict the impact of future decisions on the point of sale. This paper describes the utility of image processing techniques to innovate the retail market in the effort to extract useful information from advertising pieces frequently used in this sector.

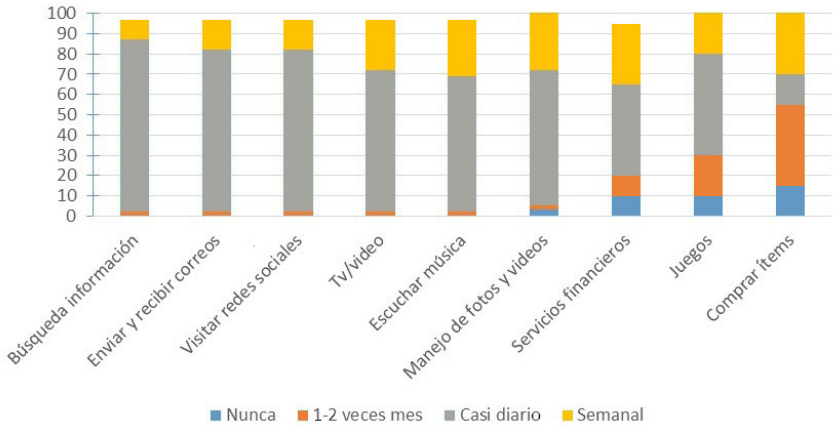
Keywords: image processing / retail marketing / machine learning / predictive analytics

Introducción

El desarrollo de algoritmos para el procesamiento de imágenes ha cobrado mayor impulso en el ámbito científico, en especial en el campo de la inteligencia artificial para aplicaciones relacionadas con visión por computadora. Este conocimiento se nutre de la digitalización de datos en dispositivos para captar videos, fotos, etcétera. Las imágenes, entonces, se convierten en el núcleo de la descripción de la realidad. Asimismo, el uso de piezas publicitarias y su digitalización en el mercado *retail* abren una oportunidad para la aplicación de técnicas de análisis de imágenes en lógicas para identificar oportunidades en los procesos comerciales, operativos y generación de conocimiento.

Si se toma como referencia que el uso de imágenes va en la misma dirección que la digitalización de contenidos, se puede partir hablando del mercado latino, en el que el 70 % de la población tiene acceso a internet y el 95% de la población está cubierta por tecnología 3G (Euromonitor, 2017). En el perfil del consumidor digital, como se observa en la figura 1, la importancia de generar experiencias que activen el ciclo de retorno en el servicio por este canal posee un gran componente visual que debe ser analizado (Kwan, 2009). La administración de dichas imágenes en repositorios, su adecuada indexación, y especialmente su utilidad como fuente de información para toma de decisiones, puede aumentar considerablemente con el aporte de métodos analíticos para extraer del contexto la contribución de esta pieza publicitaria a la venta realizada.

Figura 1. Perfil del consumidor digital – Latinoamérica 2016



Fuente: Euromonitor (2017)

A toda dirección comercial le sería beneficioso contar con la información de las imágenes utilizadas para un proceso comercial, ya que ello puede servir para entender el comportamiento de los clientes. El problema inicial es cómo una organización puede manejar tantas

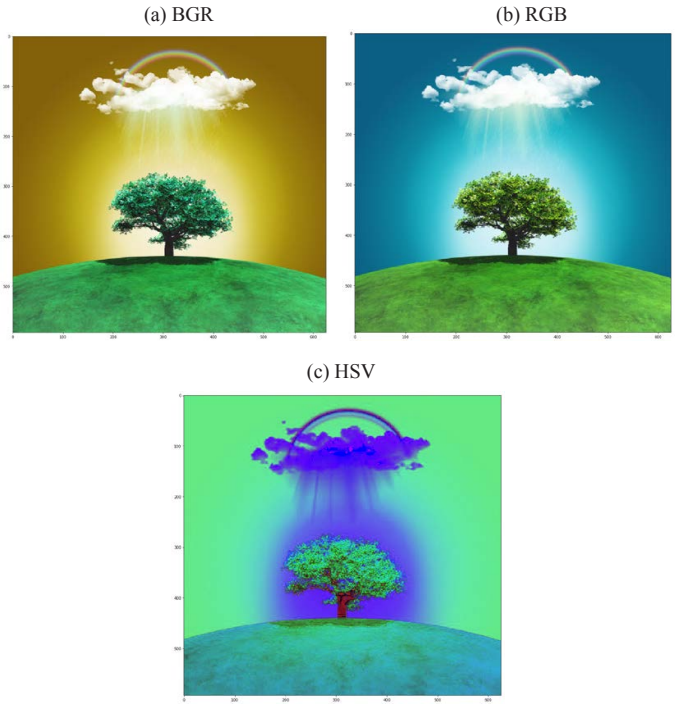
imágenes que se encuentran almacenadas en sus sistemas y que no necesariamente interactúan con la *metadata* de su negocio. Este artículo busca responder a esta interrogante.

2. Marco teórico

2.1 Espacios de color

Entre los espacios de color más utilizados para el procesamiento de imágenes están el rojo, el verde y el azul (RGB). Según la Comisión Internacional de Iluminación, en su norma técnica CIE 1931-RGB, el color se conforma por tres dimensiones monocromáticas perceptibles al campo visual. Sin embargo, estos no son los únicos espacios de color; existen, entre otros, los espacios HSV (Hue-Saturation-Value) y los espacios L*a*b, más orientados a describir la percepción del color ante un cambio en alguna dimensión de la imagen. La elección de un espacio de color está en función del uso que se desea para la imagen, en caso de que fuera necesario hacer un análisis en relación con la superficie donde se utilizará. En este tipo de aplicaciones se emplea la combinación CMyK (Cyan, Magenta y Key) que tiene mucha utilidad para la comparación de patrones de impresión. Para el campo de procesamiento y análisis de pixeles en una imagen se utiliza en especial el espacio de color RGB. En la figura 2 se muestran diversas aplicaciones de espacio de color para una misma imagen.

Figura 2. Imagen en espacio de color diferente



Elaboración propia

2.2 Pixel

Cada imagen consiste en un conjunto de pixeles. Los pixeles son la unidad básica de análisis en una imagen. No hay un elemento de menor dimensión y más fino en una imagen que el pixel. Se suele relacionarlo con "color" o "intensidad". Si pensamos en una imagen como una cuadrícula, cada cuadrado en la cuadrícula contiene un solo pixel. Por ejemplo, una imagen con una resolución de 500 x 300 significa que dicha imagen está conformada por una matriz de pixeles con 500 filas y 300 columnas.

La mayoría de los pixeles se representan de dos maneras: escala de grises y color. En una imagen en escala de grises, cada pixel tiene un valor entre 0 y 255, donde cero corresponde al color negro y 255 al color blanco. Los valores entre 0 y 255 representan variaciones de los tonos de gris, donde los valores cercanos a 0 son más oscuros y los cercanos a 255 son más claros. Los pixeles de color se representan normalmente en espacios de color; el más utilizado de ellos es el denominado RGB, compuesto por un valor para el componente rojo, uno para el verde y uno para el azul. Cada uno de estos tres colores está representado por un número entero en el rango 0 a 255. Dado que el valor de pixel solo necesita estar en el rango [0, 255] se usa normalmente un entero sin signo de 8 bits para representar cada intensidad de color. Como referencia, a continuación se muestran ejemplos de colores y su vector RGB:

Tabla 1. Codificación

Color	Vector RGB
Negro	(0, 0, 0)
Rojo	(255, 0, 0)
Azul	(0, 0, 255)
Marrón	(128, 0, 0)
Amarillo	(255, 255, 0)
Blanco	(255, 255, 255)
Verde	(0, 255, 0)
Fucsia	(255, 0, 255)
Morado	(128, 0, 128)
Azul marino	(0, 0, 128)

Elaboración propia

Al formar un vector RGB (rojo, verde, azul), este vector representa al color; así, para construir un color blanco, un vector tendría cada uno de los colores rojo, verde y azul, quedando: (255, 255, 255) y luego, para un color negro, se tendría el vector (0, 0, 0).

De esa manera cada pixel queda definido en toda la grilla de la imagen; esta grilla se convierte en un sistema coordenado, empezando siempre en la esquina superior izquierda, con la coordenada del punto (0, 0). Partiendo de este punto, los valores de los ejes *x e y* configuran las regiones y las diversas operaciones numéricas que se pueden aplicar a la imagen.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1)\ldots & f(0,n-1) \\ f(1,0) & f(1,1)\ldots & f(1,n-1) \\ f(m-1,0) & f(m-1,0)\ldots & f(m-1,n-1) \end{bmatrix}$$

2.3 Etapas en el procesamiento de imágenes

Las etapas de procesamiento parten de una secuencia de captura de la imagen y el preprocesamiento de esta. Si el análisis se orienta a una procesamiento de aprendizaje de imágenes continuas (como el caso de un video), para métodos de aprendizaje de inteligencia artificial las técnicas de enfoque y nitidez de la captura cobran mayor relevancia (Kaur, 2016).

En cualquier caso, se requiere de un preprocesamiento y el objetivo de esta etapa es reducir el "ruido" modificando la escala de una imagen, y acotando con valores límites, superior o inferior, la intensidad de cada pixel en su respectiva región. Por lo general se aplica difuminación y binarización, pero en el caso de que se necesite identificar características propias de la imagen se requiere que el preprocesamiento esté orientado a encontrar y extraer listas de detectores o características, que son regiones de pixeles con valores que cumplen una función umbral y forman parte de un algoritmo que puede variar si se busca similitud alta (homografía geométrica) o similitud cercana, como son los detectores para hacer "match robustos" (Montoya, Cortes y Chaves, 2014).

Para normalizar el contraste y los efectos de brillo, un método muy común es sustraer la media de las intensidades de las imágenes y dividirlas por su respectiva desviación estándar. Cuando se trata de imágenes a color, suele hacerse un cambio binario para convertir la imagen a la escala de grises y evitar mayor procesamiento con tres canales de colores básicos que tiene toda imagen en color (rojo, verde y azul). Esta binarización siempre es un preprocesamiento de la imagen para convertirla en una representación de pixeles. En función al objetivo del procesamiento —sea para transformar la imagen, clasificarla o utilizarla para reconocer otra— el analista deberá identificar si conviene modificar el espacio de color, el cambio a grises o cambiar los parámetros de optimización de vectorización de la imagen haciendo prueba y error e iterando esta lógica hasta obtener los resultados esperados.

Si el análisis fuera el de comparar dos imágenes, se requerirán etapas de *cropping* o segmentación. Entre las técnicas mas utilizadas de preprocesamiento tenemos la binarización y el difuminado, que se detallan a continuación.

2.3.1 Binarización

Antes de procesar una imagen, habitualmente se buscan regiones dentro de ella que puedan ser consideradas como un pico, o comportamientos en la secuencia de pixeles, para ser clasificados como integrantes o no de un mismo patrón y región. Una de estas técnicas de preparación es la binarización, que ocurre al transformar la imagen a una escala de grises donde cada pixel recibe un valor entre 0 y 255.

Figura 3. Binarización a escala de grises

Elaboración propia

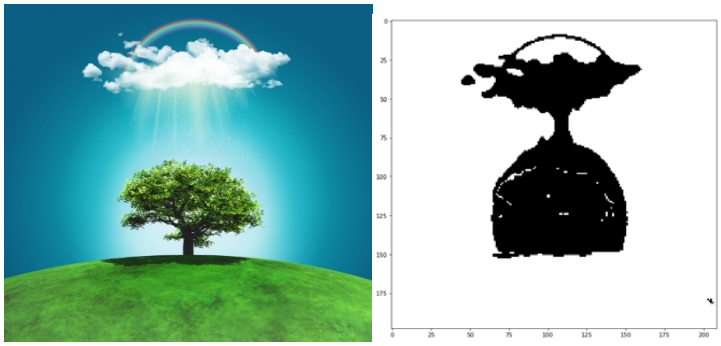
Figura 4. Imagen convertida a escala de grises

230	226	217	183	121	170	115	91	154	230	228	222	222	226	229	229	228	228	227	227	226
226	223	227	183	161	118	109	87	202	226	226	227	224	227	229	229	227	226	227	229	229
218	210	206	139	150	92	143	74	196	226	231	226	226	227	230	230	226	224	227	230	228
174	181	203	169	117	178	120	82	119	224	234	226	228	223	230	225	233	228	226	232	219
164	189	183	168	168	171	158	95	193	233	231	226	225	216	223	238	226	222	239	225	176
173	191	212	199	141	142	111	211	225	228	232	223	208	149	192	225	236	231	222	160	139
155	167	224	176	103	99	85	218	230	227	224	211	124	117	214	235	227	224	141	150	206
205	212	227	218	104	192	114	206	208	197	233	215	153	160	216	214	219	165	164	106	181
218	232	236	219	141	175	179	167	130	120	181	217	172	201	222	177	162	135	154	111	126
229	231	227	197	197	176	237	215	164	126	107	135	180	208	212	167	131	135	214	189	123
232	223	231	223	168	219	205	203	224	164	117	117	140	186	176	188	159	186	200	185	119
184	207	222	220	206	217	208	115	135	134	105	149	164	119	199	195	200	134	163	146	131
108	214	214	169	211	190	185	152	105	110	109	117	97	99	183	180	112	104	118	134	135
120	135	175	148	157	131	170	202	134	165	187	172	163	167	162	150	125	97	87	51	116
127	140	124	140	153	142	99	114	100	165	126	146	124	113	144	177	99	98	111	70	85
135	116	129	121	104	154	167	140	111	114	110	146	107	70	99	93	80	101	63	78	112
107	156	105	107	98	114	99	89	130	119	133	120	105	125	163	138	107	65	80	91	145
127	93	86	123	103	93	68	104	112	135	154	121	81	139	49	167	91	150	90	64	138
92	67	42	42	87	84	69	27	28	39	119	164	171	182	78	42	73	102	71	94	71

Elaboración propia, basada en una imagen descargada de Freepik.

Al utilizar esta técnica se crea una representación binaria de la imagen, buscando énfasis en áreas específicas. Una ventaja del uso de esta técnica es la clasificación de cada pixel considerando un umbral (*threshold*) para asignar el valor de cero o 255, en el caso de que el valor del pixel sea menor o mayor a dicho umbral. En la figura 5 se busca binarizar la nube y el árbol.

Figura 5. Binarización con umbral



Elaboración propia

2.3.2 Difuminado

Una de las técnicas de procesamiento es la de *blurring* o difuminado, la cual consiste en promediar el valor de un pixel con los valores de intensidad de los pixeles circundantes. Esta mezcla de intensidades genera una imagen borrosa. A diferencia de una foto que se desea que luzca lo más nítida posible, la técnica de difuminado ayuda en el procesamiento de imágenes en tareas como acotamiento de regiones en una imagen y detección de bordes. Entre los diversos métodos de *blurring* se puede mencionar el método gaussiano, que utiliza la media de distancia entre los pixeles circundantes a un pixel central, haciendo que aquellos más cercanos a este centro tengan mayor ponderación.

Figura 6. Imagen difuminada *(Gaussian Blur*)

Elaboración propia

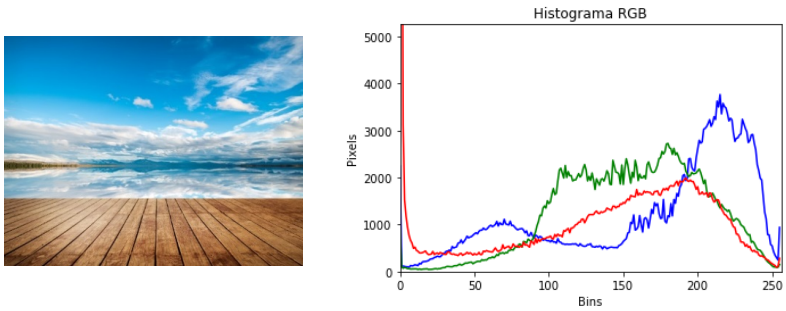
2.4 Histogramas de gradientes

Es bastante común que una imagen de entrada posea demasiada información adicional que no es necesaria para la clasificación y posterior análisis. Por lo tanto, el primer paso en la clasificación de imágenes es simplificar la imagen extrayendo la información importante contenida en esta y dejando de lado el resto. Este paso se llama *extracción de características*. En los enfoques tradicionales de visión por computador, el diseño de estas características es crucial para el rendimiento del algoritmo. Algunas técnicas utilizadas en la visión por ordenador son los detectores de características de Haar (Wang, 2014), Histograma de Gradientes Orientados (HOG), Transformación de Caracteres Invariantes (SIFT), Función Robusta Acelerada (SURF), entre otras.

Un histograma representa la distribución de las intensidades de los pixeles (color o escala de grises) en una imagen. Se puede visualizar en un histograma el nivel de la intensidad (valor de pixel) del color. En un espacio de color RGB en los valores de pixel estarán en el rango de 0 a 255. Al trazar el histograma, el eje X con 256 bins, entonces estamos contando efectivamente el número de veces que cada pixel produce este valor. En contraste, si se utilizan solo dos segmentos (espaciados equitativamente), entonces se estaría contando el número de veces que un pixel está en el rango [0, 128] o [128, 255]. Simplemente, al examinar el histograma de una imagen se obtiene una comprensión general con respecto al contraste, brillo y distribución de intensidad.

Por ejemplo, al examinar el histograma de color en las figuras 7 y 8 se observa un pico agudo en el histograma azul alrededor de bin 210. Esto indica un valor azul más claro debido a la imagen del cielo.

Figura 7. Histograma de color - Imagen A

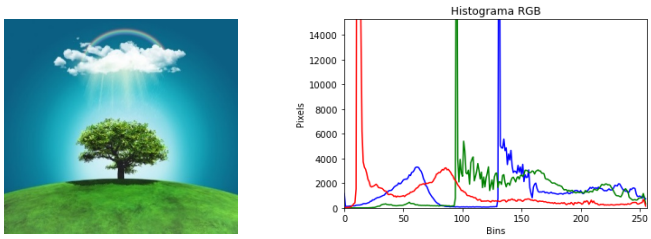


Nota. Imagen obtenida de Freepik

Elaboración propia

Por otro lado, en la figura 8 (histograma respectivo) se observan picos en los tres colores. El pico en el color rojo obedece a la sombra que genera el árbol; y los picos en los colores verde y azul, a que hay una uniformidad en la presencia de una misma intensidad de color.

Figura 8. Histograma de color - Imagen B



Nota. Imagen obtenida de Freepik

Elaboración propia

Los algoritmos de procesamiento buscan extraer características al convertir una imagen de tamaño fijo en un vector. Esta transformación sirve para detectar patrones. El descriptor de la característica denominado histograma de gradientes (HOG) se calcula para una región de 64 × 128 pixeles de una imagen, moviendo bloques de pixeles en pasos de 8 en dirección X e Y, devolviendo un vector de tamaño 3.780, esto es, 7 pasos en la dirección horizontal y 15 pasos en la dirección vertical (7 x 15 = 105 pasos), en cada paso procesa 36 valores obtenidos de la normalización del histograma de gradiente (9 bines para 4 bloques contiguos), lo cual hace un total de 105 x 36 = 3.780. Obsérvese que la dimensión original de esta región de imagen en color era 64 x 128 x 3 = 24.576, que es reducido a un vector de 3780 elementos por el descriptor HOG.

El HOG se basa en la idea de que la apariencia del objeto local puede describirse mediante la distribución (histograma) de las direcciones de los bordes (gradientes orientados). Al aplicar filtros para clasificar los pixeles se calcula para cada bloque en la secuencia de coordenada *x* e *y* la gradiente g_x e g_y de la imagen original. La gradiente es un vector con las derivadas parciales:

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{df}{dx} \\ \frac{df}{dy} \end{bmatrix}$$

La derivada de una imagen puede ser aproximada por diferencias finitas. Para evitar distorsionar la imagen, se aplica un filtro 3 x 1:

$$\begin{bmatrix} -1 \\ 0 \\ +1 \end{bmatrix}$$

La dirección de la gradiente y la magnitud se obtiene de las fórmulas

$$\theta = tan^{-1}\left[\frac{g_y}{g_x}\right], \theta \in [0^{\circ}, 180^{\circ}] \text{ y } \sqrt{{g_y}^2 + {g_x}^2}$$

En cada pixel de una celda 8 × 8 se conoce el gradiente (magnitud y dirección), y por lo tanto se tiene 64 magnitudes y 64 direcciones, es decir, 128 números. El histograma se construye asignando estos 128 números en un histograma de 9 bines (es decir, 9 números). Los bines del histograma corresponden a las direcciones de los gradientes con

valores de 0, 20, 40 ... 160 grados. Si la dirección del gradiente en un pixel es exactamente 0, 20, 40 ... o 160 grados, el histograma asigna este puntaje al respectivo bin. Un pixel en el que la dirección del gradiente no es exactamente 0, 20, 40 ... 160 grados divide su voto entre los dos bines más cercanos basados en la distancia desde el bin inicial.

El HOG es un descriptor de características, que fue desarrollado para aplicaciones en detección de personas en combinación con algoritmos de máquinas de soporte vectorial (SVM), (Dalal y Triggs, 2005). Un clasificador que utiliza HOG puede ser empleado para la detección de objetos; entre los *software* que utilizan este detector podemos mencionar el OpenCV y el Octave for Matlab, ambas herramientas con estándar abierto. Existen varios clasificadores, y uno de los más conocidos es el clasificador de cascada, muy usado para la detección de rostros.

3. Marco referencial

El reconocimiento de imágenes como campo científico se considera parte de los estudios relacionados con la visión por computadora, que a su vez forma parte de un campo mayor como es la inteligencia artificial. Su principal objetivo es identificar contornos, formas y objetos específicos que se realizan en tareas como identificación y clasificación de las imágenes y localización de áreas en una imagen (Hidalgo y Sánchez, 2015).

Los sistemas de reconocimiento de imágenes usualmente contienen funcionalidades de presentación, búsqueda y navegación a través de ficheros de imágenes; un sistema de este tipo utiliza las características de las imágenes, como el color, la textura y la ubicación espacial de los objetos. Estas características pueden ser globales o locales; se dice que son globales cuando se refiere a información del total de la imagen, y son locales cuando se refieren a una zona específica de ella. Entre las principales herramientas se puede mencionar a QBIC[1] y Project Visiir[2], entre otras.

3.1 Métodos de reconocimiento de regiones en imágenes

Considerando los métodos que utilicen, estas herramientas se pueden dividir en dos categorías (Shrivastava y Tyagi, 2016):

i. Sistemas que designan regiones de interés de manera automática (SDR).

ii. Sistemas con regiones de interés designadas por el usuario (UDR).

1 Query by image content. IBM, image solutions.

2 Visual Seek for Interactive Image Retrieval.

Ambos tipos tienen maneras de clasificar y recuperar regiones parciales de un sistema de base de datos. Los sistemas SDR fraccionan de manera automática las regiones de interés considerando algoritmos que le indican aquellas regiones que detectan picos; y los sistemas UDR brindan opciones para que el usuario asigne la región que debe ser resaltada en la imagen como región de interés. El éxito de los sistemas SDR se basa en la eficacia de las diversas técnicas para clasificar las regiones dentro de una imagen. Sin embargo, la clasificación de imágenes puede producir ruido inesperado, dando como resultado una asignación poco fiable. Por razón de estas distorsiones, los sistemas SDR usualmente tienen limitaciones y aplicaciones muy específicas. Por el contrario, la tecnología UDR parece ser más prometedora, al permitir al usuario resaltar las regiones de interés. Esto lo hace dividiendo la imagen en bloques que extraen las características importantes de dicha región y comparándola con la que el usuario ha definido que es de interés. Si la región de interés está en una ubicación específica de la imagen, las métricas como ubicación son más precisas en comparación con la búsqueda de esta región de interés en cualquier zona que no necesariamente coincida con la zona inicial.

3.2 Aplicaciones del reconocimiento de imágenes

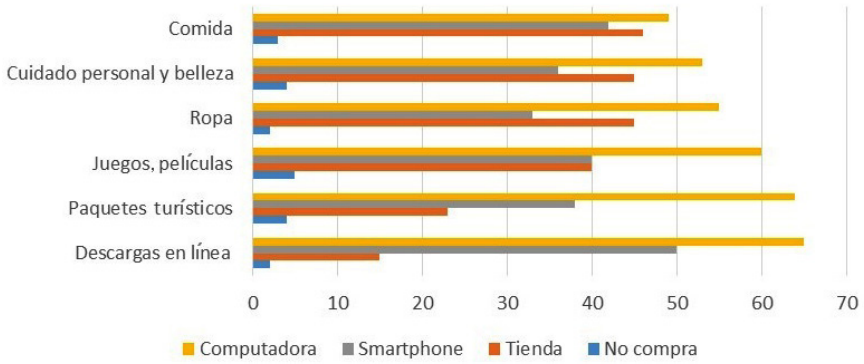
El análisis de campo de imágenes ha abierto la posibilidad de estudiar la influencia de la imagen en un diseño comercial, debido a que genera una serie de interacciones en toda la mente del consumidor. Estudios recientes sobre los efectos captados visualmente en relación con la exposición de usuarios a dispositivos que captan imágenes, permiten analizar los efectos del recuerdo y el agrado que brindan los estímulos (Zielke, 2010). Diversos estudios indican que existe un patrón de actividad visual en las personas y estas diferencias influyen en este impacto esperado para el fin por el que se presenta una imagen (Cuesta, Niño y Rodríguez, 2017).

De igual forma, el estado del arte en el reconocimiento de imágenes muestra avances notables en aplicaciones para todo el sector de la industria y los servicios. El seguimiento de la vista como herramienta para identificar regiones visuales va de la mano con el diseño de los dispositivos donde estas se plasman (Seeman, 2013). El espacio digital en pixeles genera un ambiente natural para aplicar diversas técnicas de clasificación; y como resultado de ello, corregir funciones de error detectadas para obtener una mejor resolución. Esto hace que técnicas como la optimización global, optimización local y funciones de pérdida, que ayudan a mejorar el procesamiento de imágenes, y las de secuencia como es el caso de los videos, se estén convirtiendo en herramientas para contextualizar escenarios. Estas tecnologías se están aplicando en ámbitos como la medicina, la biología y la industria en general.

3.3 Aplicaciones factibles para el mercado peruano de venta *retail*

Para contextualizar el potencial uso de las técnicas de procesamiento de imágenes se describirá el mercado *retail* peruano, que se compone de venta en tiendas por departamentos y venta directa. El canal de venta directa, según la Cámara Peruana de Venta Directa (Cámara Peruana de Venta Directa – Capevedi (mayo del 2017), reúne a 18 empresas representativas del sector que gestionan siete categorías comerciales (artículos del hogar, bienestar corporal, ropa y calzado, *bijouterie*, cosméticos, lencería y nutricionales), y suman un total de aproximadamente 496 000 consultoras en todo el país. Del total de esta venta se espera que en el 2017 las categorías de cosméticos y *bijouterie* representen el 70 % de la facturación global del rubro. Para el 2021, la tendencia en venta directaes de aproximadamente cinco billones de soles. Este importante segmento del canal *retail* se compone de venta por catálogo y en los últimos años viene desarrollando el esfuerzo de digitalizar el proceso comercial, como se observa en la tendencia regional mostrada en la figura 9.

Figura 9. Preferencias en el canal de compra - Latinoamérica 2016



Fuente: Euromonitor (2017)

De acuerdo con un estudio realizado por Álvarez (mayo del 2016), los consumidores peruanos, en la venta detallista por tiendas, compran por impulso y buscan alternativas que se encuentren cerca a su hogar o a su trabajo. En la actualidad, el *boom* de las bodegas se encuentra en una dinámica distinta debido a la aparición de tiendas pequeñas que ofrecen menos prestaciones de espacio pero mayor movimiento de margen por volumen, al brindar una mayor cantidad de tiempo de atención al día. Como referencia, es interesante analizar el crecimiento de la marca Tambo y la aparición de formatos similares en poco tiempo.

En cada uno de estos segmentos de comercio *retail*, la importancia dada a lo visual y el estímulo de la intención de compra de los clientes con publicidad digital requiere de un soporte analítico que integre el componente de procesamiento y análisis de las imágenes utilizadas.

4. Aplicación para identificar regiones en imágenes

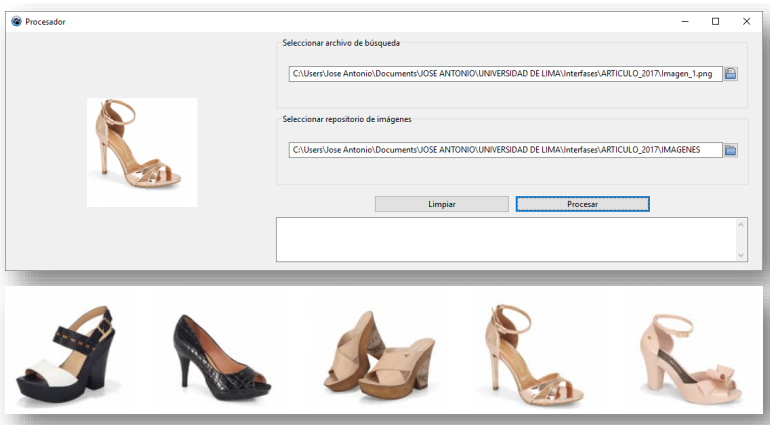
Para verificar la factibilidad de utilizar estas herramientas en la búsqueda de imágenes similares contenidas en otras, se utilizó Open CV[3]. Esta es una librería de código abierto para visión de computadora, que se integra al ambiente de desarrollo en lenguaje Python de la suite Anaconda Framework; inicialmente desarrollada en lenguaje C, y posteriormente reescrita para lenguaje Python[4]. Es bastante utilizada en el campo de la analítica de predicción, aplicaciones de aprendizaje de máquina (*machine learning*) y aplicaciones consideradas de aprendizaje no supervisado denominadas *deep learning*.

Una mención especial es el uso de Open CV, considerada una plataforma líder en el mercado en temas relacionados con la visión de computadora, y se integra —al ser de fuente abierta— con diversos sistemas operativos, como Ubuntu, MacOS y Windows.

Para el desarrollo de este artículo se diseñó una aplicación que permita buscar una imagen que obtenga el mayor puntaje en la comparación de esta con una serie de imágenes colocadas en un repositorio que se convertirá en el repositorio de búsqueda, como es el caso de la base de imágenes por analizar en la presente aplicación.

Se preparó una interfaz para el ingreso de la ruta de la imagen por comparar y el directorio que contiene diversas imágenes, del cual debería identificar si existe alguna que contenga a la imagen en búsqueda. Se utilizó el IDE *open source* Lazarus[5] para el ingreso de parámetros, iterando todos los archivos y generando una lista con aquellos resultados que tuvieron mejor puntaje en la cantidad de puntos comunes. La interfaz se muestra en la figura 10.

Figura 10. Ingreso de parámetros para la búsqueda de imagen



Fuente: Catálogo Litzy (www.litzy.pe)

3 www.opencv.org
4 www.python.org
5 www.lazarus-ide.org

La sintaxis de preparación de la imagen consiste en cambiar el espacio de color a grises y la reducción del tamaño de la imagen, puesto que no se requiere un análisis de áreas o superficies sino, básicamente, detectar picos de valores luego de aplicar algoritmos de similitud. Para este proyecto se utilizan las librerías Numpy[6] para convertir las imágenes en matrices y arreglos. Numpy es una librería escrita en lenguaje Python de amplio uso en el ámbito de la ciencia de datos. Asimismo, se utiliza la librería Matplotlib[7] para mostrar el resultado y la integración con los vectores Numpy.

Existen diversos algoritmos para encontrar los puntos de similitud entre dos imágenes, entre los más difundidos se encuentran los siguientes:

a) Harris Corner Detection.

b) Scale Invariant Feature Transform (SIFT).

c) Speed up robust feature (SURF).

d) Oriented FAST and Rotated BRIEF (ORB).

En la aplicación desarrollada se trabajó con el algoritmo ORB (Oriented FASTand Rotated BRIEF), una alternativa de *software* libre a los algoritmos SIFT y SURF que son licenciados. ORB es básicamente una fusión de un detector FAST para localizar zonas pico y un descriptor BRIEF (Binary Robust Independent Elementary Features), escogiendo los *n* mejores picos de comparación luego de aplicar el método Harris Corner Detection, es decir, utiliza una medida de orientación, calculada entre la esquina de una región en la imagen y el centroide de intensidad de dicha región. Este método asume que la orientación hacia una esquina en una región es el desplazamiento desde el centroide, y este vector puede ser usado para determinar la gradiente dominante que debe ser invariante ante una rotación de esta para obtener un descriptor robusto (Rublee, Rabaud, Konolige y Bradski, 2011).

$$m_{pq} = \sum_{x,y} x^p \; y^q I(x,y)$$

$$C = \left(\frac{m_{10}}{m_{00}} , \quad \frac{m_{01}}{m_{00}} \right)$$

$$\theta = atan2(m_{01} , m_{10})$$

6 http://www.numpy.org/

7 https://matplotlib.org/

Como mencionan los creadores del ORB, las ventajas que tiene este algoritmo en comparación con otros similares son:

a) La adición de un mejor componente de orientación.

b) Un eficiente procesamiento computacional.

c) Mejoras en el desempeño del algoritmo de medición de distancia entre secuencias.

A continuación se presenta la sintaxis y una descripción (# al inicio del código) en Python para obtener estos puntos:

```
# Se activan las instancias de manejo de arrays y procesamiento de pixeles.
import numpy as np
import cv2
from matplotlib import pyplot as plt

#Crea el objeto ORB
orb = cv2.ORB_create()

#Orb.DetectAndCompute : Kp1: es la cantidad de invariantes encontrados
#des1: Es el la matriz de la imagen1
kp1, des1 = orb.detectAndCompute(img1,None)
```

Figura 11. Array descriptor de la imagen fuente

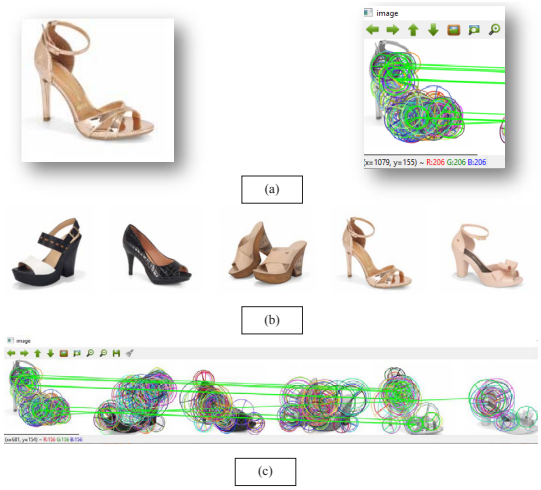
	0	1	2	3	4
0	37	169	136	135	252
1	66	71	122	220	106
2	245	211	69	239	142
3	18	76	239	253	110
4	26	79	66	158	210
5	2	198	176	155	85
6	255	75	228	70	95
7	19	92	122	146	246
8	125	5	205	196	76
9	222	188	191	114	131
10	209	21	29	252	10
11	201	196	21	175	36

Fuente: Spyder - Python

```
kp2, des2 = orb.detectAndCompute(img2,None)

#La distancia de Hamming es una de varias métricas de cadena utilizada para medir la
distancia entre dos secuencias de vectores.

bf = cv2.BFMatcher(cv2.NORM_HAMMING, crossCheck=True)

#BFMatcher: Brute force matcher. Utiliza todos los descriptores en el primer set
buscando #equivalentes en el segundo set de descriptores de la imagen destino. Es el
método más #robusto de los que detectan descriptores.

matches = bf.match(des1,des2)
 matches= sorted(matches, key = lambda x:x.distance)

# Dibuja la cantidad de líneas que hacen match entre las 2 imágenes considerando al
menos #20 enlaces.

img3 = cv2.drawMatches(img1,kp1,img2,kp2,matches[:20],None, flags=4)

#Genera el gráfico de resultado utilizando Matplotlib

plt.imshow(img3)
plt.show()
```

El resultado de la búsqueda realizada se aprecia en la figura 12.

Figura 12. Resultado de la búsqueda de imagen



Elaboración propia

En la figura 12 se puede observar que la imagen (a) ingresada para ser buscada aparece reconocida en la segunda imagen (b), luego de iterar en un directorio con una serie de imágenes. El resultado está identificado en la imagen (c). Esto permite localizar la región en la imagen, resultado de evaluar los puntos comunes.

5. Conclusiones

- Es factible aplicar métodos de procesamiento de imágenes para construir un buscador de regiones de imagen en un directorio con múltiples archivos.

- En relación con los algoritmos que se pueden utilizar para detectar imágenes, la técnica ORB muestra resultados correctos en pruebas realizadas para distintos sets de imágenes, y es una ventaja adicional que tenga categoría de *software* libre. A la vez, integrándose en lenguajes como Python, puede abrir más posibilidades de aplicaciones en la gestión comercial.

- En diversas pruebas realizadas para distintos datasets de imágenes, un elemento adicional es la configuración de la extensión de la imagen, siendo PNG (Portable Network Graphics) el formato de compresión sin pérdida de nitidez, muy adecuado por su calidad y menor peso de almacenamiento en comparación con otros formatos como tiff, jpg, etcétera.

- El aplicativo desarrollado para este artículo es un punto de inicio para otras funcionalidades que los algoritmos de procesamiento de imágenes permiten, como identificación de objetos en la imagen, clasificación según etiquetas de información, reconocimiento de imágenes similares y texturas en ellas.

- El potencial uso de la información que se obtiene al determinar el área utilizada de una imagen es amplio. Puede servir como una variable independiente que explique el comportamiento de una venta comercial, también permitiría que una región de una imagen reciba una etiqueta de caracteres, convirtiendo cualquier repositorio de imágenes en una fuente de datos útiles para contextualizar y predecir con técnicas analíticas escenarios comerciales futuros.

Referencias

Álvarez, J. (Mayo del 2016). Perfil del ama de casa en Lima Metropolitana. Informe Gerencial de Marketing. Base de datos Marketing Data Plus. Apoyo Opinión y Mercado S. A.

Cámara Peruana de Venta Directa – Capedevi (mayo del 2017). Venta Directa al cierre del primer trimestre 2017. *Boletín Mensual*, *53*. Recuperado de http://www.capevedi.com/Public/boletines.html

Cuesta, U., Niño, J. y Rodríguez, J. (2017). The Cognitive Processing of an Educational App with Electroencephalogram and "Eye Tracking". *Comunicar*, *25*(52), 41-50. DOI: 10.3916/C52-2017-04

Dalal, N., y Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. En *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (Vol. 1, 886-893). IEEE. DOI: 10.1109/CVPR.2005.177

Euromonitor International. (Agosto del 2017). Digital Consumer Profiles: How Latin Americans will shop and spend digitally. Análisis del consumidor. Base de datos Euromonitor International.

Hidalgo, I., y Sánchez, R. (2015). Reconocimiento de caracteres mediante imágenes en contadores de gas en entornos reales (Trabajo de fin de grado). Universidad Complutense de Madrid, España.

Kaur, S. (2016). An automatic number plate recognition system under image processing. *International Journal of Intelligent Systems and Applications*, *8*(3), 14-25. DOI: 10.5815/ijisa.2016.03.02

Kwan, S. P. (2009, June 25). Methods and apparatus for improved image processing to provide retroactive image focusing and improved depth of field in retail imaging systems. Google Patents. Recuperado de https://www.google.ch/patents/US20090160975

Montoya, C., Cortés, J., y Chaves, J. (2014). Sistema automático de reconocimiento de frutas basado en visión por computador. *Ingeniare. Revista Chilena de Ingeniería*, *22*(4), 504-516. DOI: 10.4067/S0718-33052014000400006

Rublee, E., Rabaud, V., Konolige, K., y Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. En *2011 International Conference on Computer Vision* (2564-2571). IEEE. DOI: 10.1109/ICCV.2011.6126544

Seeman, M. (2013). Image processing for improved perception and interaction. *Information Sciences and Technologies Bulletin of the ACM Slovakia*, *5*(3), 8-12.

Shrivastava, N., y Tyagi, V. (2016). An integrated approach for image retrieval using local binary pattern. *Multimedia Tools and Applications*, *75*(11), 6569-6583. DOI: 10.1007/s11042-015-2589-2

Wang, Y. (2014). An analysis of the Viola-Jones face detection algorithm. *Image Processing On Line*, *4*, 128-148. DOI: 10.5201/ipol.2014.104

Zielke, S. (2010). How price image dimensions influence shopping intentions for different store formats. *European Journal of Marketing*, *44*(6), 748-770. DOI: 10.1108/03090561011032702