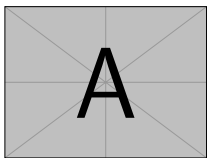
Construcción de mosaicos a partir de imágenes de sonar de barrido lateral

Trabajo Fin De Máster

Por
José Manuel Bernabé Murcia
josemanuel.bernabe@um.es



MÁSTER EN NUEVAS TECNOLOGÍAS

UNIVERSIDAD DE MURCIA

Tutor
Dr. Humberto Martínez Barberá humberto@um.es

Índice

1. Introducción

Bla bla bla

2. Estado del arte

En esta sección trataremos de hacer un repaso de las difierentes técnicas que se hacen uso para la generación de mosaicos a partir de imágenes de sonar de barrido lateral, y también se describirá brevemente que es SLAM y la importancia que juega las imágenes acústicas dentro del mismo.

En los orígenes de los sonar de barrido lateral, las imágenes se imprimían en largas hojas de papel, y la generación de mosaicos de la imágenes recolectadas se realizaban literalmente cortando y pegando [1], como se puede apreciar en la figura 1.

Figura 1: Formando mosaico. [1]

El proceso de crear el mosaico solo se podía hacer una vez hecha la misión, es decir, una vez recopilado y procesado los datos, con el avance y el asentamiento de la tecnología digital, estos procesos se han ido informatizando haciéndolos menos costoso en relación tiempo y coste. Inclusive pudiendo hacerse en tiempo real. [1]

Actualmente, la construcción de mosaicos se compone de líneas de pixeles adyacentes. El proceso de generación de mosaicos es simple, pero realizar las transformaciones, escalas, rotaciones y translaciones, y hacerlo de forma automática hace que aumente su complejidad, para la realización de mosaicos se debe hacer un registro de imágenes [2], en el campo de la visión por computación un registro de imágenes es el procedimiento de superponer dos o más imágenes de la misma escena tomadas en difierentes momentos, desde difierentes puntos de vista y/o por difierentes sensores, alineada geométricamente [2]. En otras palabras, es el proceso de transformar una imagen en difierentes perspectivas dentro de un mismo sistema de coordenadas [3]. La mayoría de los algoritmos usados hoy en día han sido desarrollados antes de la década de los noventa, y dentro de ellos los podemos clasificar en dos métodos principales, métodos no simbólicos y métodos simbólicos. [4]

Es importante destacar que la resolución obtenida de las imágenes del sonar

y los mosaicos, están altamente influenciados por: la corriente, giros bruscos del AUV o ROV, altitud, los ángulos de los haces acústicos e imperfecciones en la localización. Para crear mosaicos con precisión, es decir, que estén bien georeferenciados, más allá de una buena navegación, es muy necesario el disponer de un buen sistema de posicionamiento, lo que es un gran reto incluso a día de hoy para equipos submarinos, ya que debajo del agua los GPS's no funcionan. En este estado del arte no vamos a entrar en los distintos tipos de posicionamiento que podríamos tener, lo que se intenta recalcar que, para hacer una buena cartografía del fondo marino, es necesario un buen sistema de posicionamiento.

2.1. Métodos no simbólicos

Los métodos no simbólicos, son métodos basados en un nivel bajo de información, como puede ser la intensidad de los píxeles. Un método no simbólico implica establecer una relación global entre los píxeles de dos imágenes utilizando solo sus niveles de gris. Los primeros algoritmos desarrollados usando métodos no simbólicos surgen a mediados de los años setenta, utilizando un criterio basado en la suma de las difierencias cuadradas de la intensidad del nivel de gris. Consideran que las imágenes están curvadas por una transformación básica geométrica o corrompidas por ruido gaussiano blanco adicional y usaban o bien optical flow o phase correlation. [4]

Desde la década de los ochenta hasta hoy, ha habido un gran incremento de tipos de sensores, así como la mejora de los ya existentes, por lo que aparecen nueva información a tener en cuenta. En particular, en imágenes por satélite, medicina y submarina. Otro enfoque dentro de los métodos no simbólicos surgió, un enfoque estocástico basándonos en una medida de similitud, que han demostrado comportarse mejor para el registro automático de imágenes sonar [4]. Existen muchas medidas de similitud, las que más destacan son: phase correlation, mutual information y rate correlation. En la tabla 1 se puede ver la gran cantidad de medidas de similitud que existen. Este enfoque estocástico ha demostrado comportarse mejor que el enfoque determinista [4], los métodos deterministas operan de una forma sistemática, generando una coincidencia local por cada pixel (optical flow).

2.1.1. Optical Flow

Optical flow es la distribución de velocidades aparentes de movimiento de patrones de brillo en una imagen, causado por el movimiento relativo entre un observador (un ojo o una cámara) y la escena. En consecuencia, optical flow puede darnos información importante sobre la disposición espacial de

los objetos y la tasa de cambio de estos, influenciados por las condiciones variables del entorno, tales como iluminación, sombras, reflejos y otros efectos luminosos. Las discontinuidades en optical flow puede ayudar a segmentar las imágenes en regiones que corresponden a difierentes objetos, en algunos casos incluso se puede recuperar la forma de ciertos objetos. [5]

El problema de optical flow se ha convertido en uno de los más importantes a resolver en el campo de la visión por computación, dada su gran importancia en campos como la reconstrucción 3D, compresión de vídeo, detección de objetos y navegación robótica.

El desplazamiento de los píxeles no es más que la proyección en una imagen del movimiento tridimensional. Se trata de un problema inverso en donde los valores de algunos parámetros del modelo deben ser obtenidos de los datos observados. Las imágenes son los datos observados y queremos conocer el movimiento que se registra en la escena. Esto provoca que la estimación de optical flow sea un problema mal condicionado ya que puede haber varias soluciones para un mismo desplazamiento, lo que da lugar a cierta ambigüedad [6], por lo que surgen restricciones para evitar esta ambigüedad.

Si representamos una imagen como una aplicación $I : (x, y, t) \rightarrow I(x, y, t)$ donde (x, y) representa la coordenada espacial de la imagen y t el tiempo, se puede ver una secuencia de imágenes como la variación de la intensidad en las coordenadas de la imagen a lo largo del tiempo. El vector desplazamiento se define como la función $h(x, y, t) = (u(x, y, t), v(x, y, t))^T$ y representa el movimiento horizontal y vertical de los píxeles a través de la suencia de imágenes. Para detectar la correspondencias de los píxeles entre dos imágenes se suele suponer que alguna propiedad de la imagen no varía a lo largo del tiempo. Esta suposición la vemos representada en la ecuación 1 donde f es algún tipo de propiedad en la imagen, y t+1 representan dos imágenes de la escena en distintos instantes de tiempo.

$$f(x + u, y + v, t + 1) - f(x, y, t) = 0 \tag{1}$$

La intensidad de los píxeles de una imagen es un valor que indica la cantidad de radiación luminosa reflejada por la superficie de los objetos. Existen muchos modelos para representar los distintos tipos de superficies. Uno de los más simples el modelo lambertiano, que asume un mismo brillo para las difierentes perspectivas, es decir, se mantendrá el mismo valor de intensidad en todas las secuencias de imágenes. Por este motivo, bastaría con sustituir f por I en la ecuación 1 para representar esta invarianza. Si analiza la expresión nos damos cuenta de que no es lineal, esta no linealidad la podemos evitar realizando un desarrollo de Taylor de dicha expresión y desechamos los términos de orden superior. La ecuación resultante es conocida como ecuación de restricción de flujo óptico.

$$I_x u + I_y v + I_t = 0 \tag{2}$$

donde los subíndices indican derivadas parciales. A partir de la expresión 2, no es posible determinar el vector desplazamiento, ya que tenemos dos incógnitas en una sola ecuación.

[6]

Clasificación de los Métodos

En esta sección vamos a realizar un breve recorrido de los algoritmos más importantes que se utilizan en optical flow, y agruparlos basándonos en [7] y [8].

Existen muchos métodos para calcular optical flow, y por ello varios investigadores han aportado sus difierentes clasificaciones, en este trabajo los agruparemos en los tres grupos principales y que más han aportado al campo de optical flow: métodos difierenciales, métodos basados en la frecuencia y métodos basados en la correlación [7].

Métodos difierenciales

Los métodos difierenciales calculan el desplazamiento de los píxeles a partir de las derivadas espaciales (o espaciotemporales) de las intensidades de la imagen. Una limitación que tiene este tipo de métodos es que obliga a que las derivadas sean computables en el dominio de la imagen.

Dentro de los métodos basados en derivadas, podemos encontrar algoritmos basándose en la primera derivada ([5]), y otros en la segunda derivada. Beauchemin-Barron ([7]) estableció la siguiente subcategoría para los métodos difierenciales: locales, globales, de superficie, de contornos y multirestricciones. Los métodos locales y globales se basan en la ecuación de restricción del flujo óptico (2), como no es posible estimar el flujo óptico únicamente con esa ecuación, es necesario una restricción adicional.

Dentro de los métodos globales, el más representativo es el de Horn y Schunck toman que toman como segunda restricción una restricción de suavidad.

Restricción de constancia de brillo, Horn y Schunck

El cambio de brillo entre dos cuadros consecutivos del video es cero o aproximadamente cero, tomando como referencia la ecuación 2. Para el cálculo de

las derivadas parciales espaciales, así como para la derivada temporal se utiliza el método de difierencias finitas. Existen muchas fórmulas para realizar este cálculo, Horn y Schunck propusierón las siguientes ecuaciones 3, 4, 5. Donde E representa el brillo en un punto de la imagen

$$\begin{aligned} E_x \approx \frac{1}{4} \{ & E_{i,j+1,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i+1,j,k} \\ & + E_{i,j+1,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i+1,j,k+1} \}, \end{aligned} \tag{3}$$

$$\begin{aligned} E_y \approx \frac{1}{4} \{ & E_{i+1,j,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i,j+1,k} \\ & + E_{i+1,j,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i,j+1,k+1} \}, \end{aligned} \tag{4}$$

$$\begin{aligned} E_t \approx \frac{1}{4} \{ & E_{i,j,k+1} - E_{i,j,k} + E_{i+1,j,k+1} - E_{i+1,j,k} \\ & + E_{i,j+1,k+1} - E_{i,j+1,k} + E_{i+1,j+1,k+1} - E_{i,j+1,k} \} \end{aligned} \tag{5}$$

El objetivo es referenciar el mismo punto en la imagen al mismo tiempo, por lo que es importante que las estimaciones de E_x, E_y y E_t, sean consistentes. Utilizan un conjunto que les da una estimación de E_x, E_y, E_t en un punto en el centro de un cubo formado por ocho mediciones. La relación en el espacio y el tiempo entre estas mediciones se muestra en la figura 2 sacada del artículo [5]. Cada una de las estimaciones es el promedio de cuatro primeras difierencias tomadas sobre las mediciones adyacentes en el cubo. Este método consiste en aproximar la derivada considerando el peso de un conjunto discreto de mediciones de brillo disponibles alrededor del pixel E(x,y,t).

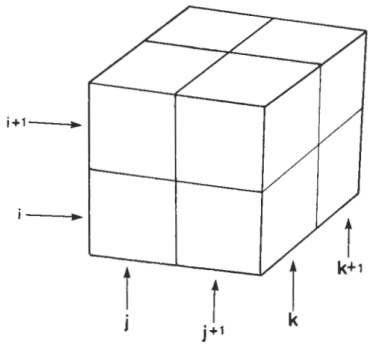


Figura 2: Aquí el índice de columna j corresponde a la dirección x en la imagen, el índice de fila i a la dirección y. mientras k se encuentra en la dirección del tiempo.

Una vez obtenidas las aproximaciones de las derivadas, y al poner una de las velocidad en función de otra en la ecuación 2, se observa que lo que se obtiene es la ecuación de una recta dentro de la cual se encuentra la solución para la determinación de optical flow. Se muestra en la Figura 3 la recta correspondiente a la ecuación $v = \frac{E_x}{E_y}u - \frac{E_t}{E_y}$ [9]

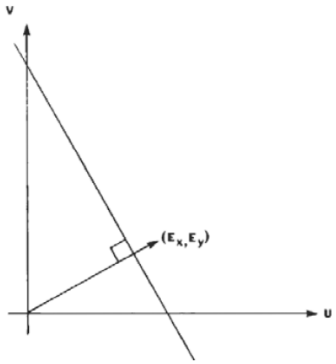


Figura 3: Línea de restricción del flujo óptico comprendida en el plano formado por las velocidades u y v. El vector perpendicular a la línea de restricción muestra el gradiente de brillo E_x, E_y. [5]

Como se observa en la Figura 3, a pesar de disponer de la ecuación que describe la restricción de constancia de brillo, el flujo óptico puede encontrarse a lo largo de la recta descrita en el plano u-v.

Restricción de suavidad

La restricción de suavidad utilizada por Horn y Schunck asume, que los pixeles que conforman los objetos de tamaño finito en la imagen (patrones de brillo) tienden a someterse a movimientos rígidos como un todo, por lo que casi nunca se encuentran pixeles con movimientos independientes de sus vecinos cercanos. Esto genera un campo de velocidades de los patrones de brillo que varía suavemente en casi toda la imagen, debido a determinadas excepciones como en el caso de presencia de texturas. Esta restricción implica la minimización de las derivadas parciales espaciales de las componentes de la velocidad. Para poder realizar esta minimización se puede plantear la suma de los Laplacianos en los ejes x e y, como plantea Horn en su artículo de 1981, con las ecuaciones 6 y 7. En estas ecuaciones se muestra una equivalencia donde se consideran promedios locales de las velocidades ($\bar{u}$ y $\bar{v}$) así como un factor proporcional k. Para lograr la minimización de los Laplacianos de las componentes de la velocidad, estos deben ser igualados a 0. [9]

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\left(\bar{u}_{i,j,k} - u_{i,j,k}\right) \tag{6}$$

$$\nabla^2 v = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \left(\bar{v}_{i,j,k} - v_{i,j,k} \right) \tag{7}$$

Al igual que se utilizaron difierencias finitas para la aproximación de las derivadas parciales, el cálculo del Laplaciano utiliza el mismo método de aproximación, tomando las ponderaciones de un cuadrado de valores de los vecinos cercanos y sustrayéndolo del valor central, como se muestra en la Figura 4.

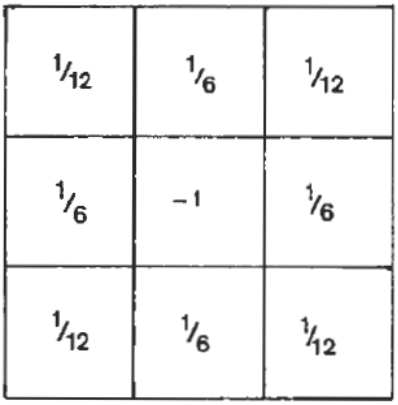


Figura 4: Cuadrado de ponderaciones con los pesos correspondientes a cada vecino para el cálculo del Laplaciano en las componentes de u y v del flujo óptico. [5]

El cálculo de los promedios locales de las componentes de la velocidad se realiza utilizando las ecuaciones 8 y 9, donde se realiza la ponderación planteada en la Figura 4. La restricción de suavidad es una suposición bastante buena para el cálculo de optical flow excepto cuando existen objetos en la imagen que ocultan a otros, provocando de esta manera una discontinuidad en el flujo [9].

$$\begin{aligned} \bar{u}_{i,j,k} = & \frac{1}{6} \left\{ u_{i-1,j,k} + u_{i,j+1,k} + u_{i+1,j,k} + u_{i,j-1,k} \right\} \\ & + \frac{1}{12} \left\{ u_{i-1,j-1,k} + u_{i-1,j+1,k} + u_{i+1,j+1,k} + u_{i+1,j-1,k} \right\} \end{aligned} \tag{8}$$

$$\begin{aligned} \bar{v}_{i,j,k} = & \frac{1}{6} \left\{ v_{i-1,j,k} + v_{i,j+1,k} + v_{i+1,j,k} + v_{i,j-1,k} \right\} \\ & + \frac{1}{12} \left\{ v_{i-1,j-1,k} + v_{i-1,j+1,k} + v_{i+1,j+1,k} + v_{i+1,j-1,k} \right\} \end{aligned} \tag{9}$$

Por otro lado, los métodos locales utilizan la información en una vecindad alrededor de un píxel para estimar su movimiento. El método más representativo de esta familia es el de Lucas-Kanade ([10]), cuyo desarrollo es el mismo que el de Horn, pero difieren en la segunda restricción. Lucas y Kanade, plantean un método que calcula el desplazamiento a partir de la minimización de la ecuación del flujo óptico alrededor de una ventana centrado en un píxel (ecuación 10) donde W(x, y) es la ventana centrada en el píxel (x, y) y N es la vecindad.

$$\sum_{(x,y)\epsilon N} W^2\left(x,y\right)\left(E_x u + E_y v + E_t\right)^2 \tag{10}$$

El mayor inconveniente de este tipo de métodos (locales) es que sólo es posible detectar el movimiento en aquellas zonas donde exista variaciones en la imagen. En zonas homogéneas, donde puede haber movimiento éste no es detectable. Por ello, los campos de desplazamiento no son densos.

Métodos basados en la Frecuencia

Los métodos basados en la frecuencia o también llamados, métodos basados en energía espacio-temporal, son métodos que utilizan la transformada de Fourier para calcular el flujo ópticos a través del dominio de la frecuencia, adaptando la ecuación que describe cualquier invarianza (2) en el dominio de la imagen, al dominio de la frecuencia obtienemos la ecuación 11

$$\hat{I}\left(k,f\right) - \hat{I}_{\mathrm{o}}(k)\delta\left(\omega^T k + f\right) = \mathrm{o} \tag{11}$$

donde $\hat{I}_0(k)$ es la transformada de Fourier de I(x,y,0), δ es la delta de Dirac y k, f es la frecuencia espacio-temporal. Según esta ecuación, cualquier patrón que se mueva de una imagen a otra por una simple traslación, se manifiesta en el dominio de Fourier como un cambio de fase. Este tipo de métodos suele resultar muy útil para la detección del movimiento de objetos que son difíciles de capturar, como es el caso de puntos aleatorios. [6]

Métodos basados en la correlación

Los métodos basados en la correlación realizan la búsqueda de correspondencias utilizando ventanas o patrones alrededor de cada píxel. La idea que subyace a estos métodos es que es mucho más fácil encontrar las correspondencias entre los píxeles a través de la comparación de regiones entre las imágenes por maximización de alguna medida de similaridad. Una ventaja

que tienen estos métodos es que al usar mayor información la búsqueda de las correspondencias es más efectiva.

$$C(\omega) = \int_{\Omega} f(x+\omega)\, g(x)\, dx \tag{12}$$

donde ω es el desplazamiento del píxel x y Ω es el dominio de la imagen. En el caso discreto la medida de la correlación en un punto que se suele tomar es el siguiente:

$$C(\omega) = \frac{\sum_{\delta\omega=-(a,b)}^{(a,b)} \left(f(x+\delta\omega) - \overline{f(x)}\right)\left(g(x+\omega+\delta\omega) - \overline{g(x+\omega)}\right)}{(2a+1)(2n+1)\,\sigma f(x)(x+\omega)} \tag{13}$$

donde (a, b) representa las dimensiones de la ventana de correlación, $\overline{f(x)}$ la media de la imagen f en esa ventana y $\sigma_f(x)$ la desviación estandar.

[6]

Hemos hecho un breve repaso de optical flow, de sus principales métodos y los dos más usados. Actualmente, optical flow es muy usado en odometría visual, este método es relativamente eficiente con imágenes nítidas, pero en imágenes con poca nitidez, como pasa con las imágenes submarinas donde hay mucho ruido, este método no da buenos resultados, por lo tanto, en la práctica no se hace uso de el, ya que es muy sensible a las variaciones luz y a la oclusión. [4]

2.1.2. Phase Correlation

Phase Correlation (PC) se calcula con la transformada rápida de Fourier en dos imágenes, y usar la difierencia entre la fase de los componentes espectrales, para evaluar el desplazamiento espacial entre ambas imágenes. Un cambio de fase en el dominio del espacio, correspondería a un desplazamiento. Las difierencias de fase están sujetas a una transformación inversa que determina una superficie de correlación, compuesta de picos cuyas posiciones corresponden a la norma de movimientos entre imágenes. Esto se puede describir de la siguiente manera: dejando $f_1(x, y)$ y $f_2(x, y)$ ser dos imágenes que se difierencian únicamente por un desplazamiento (x_0, y_0)

$$f_2(x, y) = f_1(x - x_0, y - y_0) \tag{14}$$

$$F_2(\varepsilon, \eta) = e^{-j2\pi(\varepsilon_{x_0} + \eta_{y_0})} * F1(\varepsilon, \eta) \tag{15}$$

La ecuación 14 está relacionada con su correspondiente transformada de Fourier 15. El espectro de potencia cruzada de dos imágenes f y f'con transformada de Fourier F y F' se define en la ecuación 16

$$\frac{F(\varepsilon,\eta)F'*(\varepsilon,\eta)}{|F(\varepsilon,\eta)F'(\varepsilon,\eta)|} = e^{-j2\pi(\varepsilon_{x_0}+\eta_{y_0})} \tag{16}$$

donde F* es el conjugado de F. El teorema de desplazamiento garantiza que la fase del espectro de potencia cruzada es equivalente a la difierencia de fase entre las dos imágenes. Al tomar la transformada inversa de Fourier de la representación en el dominio de la frecuencia, obtendremos una función que es un impulso, o llamada correlación superficial; es decir, es aproximadamente cero en todas partes, excepto donde el desplazamiento es adecuado para el registro óptimo. Luego aparecen picos en estos lugares, debemos de eligir cual de estos picos es el que más apropiado para representar el movimiento real. Luego se aplica una coincidencia de bloque orientada (block matching) en las imágenes de nivel de gris correspondientes (normalizadas), que compara cada conjunto candidato, centrado en uno de los picos de superficie de correlación en la primera imagen, con su correspondiente en la segunda imagen de nivel de gris. De hecho, un vector candidato se define desde el centro de la imagen referenciada a cada uno de los picos de la superficie de correlación. El máximo de correlación en los niveles de gris normalizados revela el vector de movimiento válido. En consecuencia, el uso de una estrategia de desplazamiento y correlación nos permite discriminar los vectores válidos de los falsos, entre los vectores candidatos. En resumen, el proceso de registro se realiza en dos etapas que son un análisis espectral, seguido de un método de coincidencia de bloque (block matching de correlación). La principal ventaja de phase correlation es su excelente robustez contra ruido aleatorio. [11]

2.1.3. Mutual Information

Mutual Information se ha hecho popular en difierentes campos, como en el registro de imágenes médicas o radares, particularmente debido a su gran indifierencia a los cambios de iluminación. Mutual Information es una medida de dependencia estadística entre dos variables aleatorias A y B, y está dada por:

$$I(A,B) = H(A) + H(B) - H(A,B) \tag{17}$$

Donde $H(A)$ es la entropía de una variable aleatoria, la cual esta definida como $H(A) = \int_{-\infty}^{+\infty} p(A)ln(p(A))dA$ y la entropía conjunta es $H(A,B) =$

$-\int_{-\infty}^{+\infty} p(A,B)ln(p(A,B))dAdB$ La entropía de una variable aleatoria X mide la cantidad de 'información' proporcionada por una observación de X. A partir de esta definición, se observa que cuanto menos probable es un evento, más información recibimos cuando ocurre.

Uno de los inconvenientes de mutual information es que no tiene en cuenta la relación espacial entre píxeles; de hecho, los histogramas no proporcionan información espacial sobre los píxeles de la imagen. Consecuentemente, muchos autores trabajaron en el tema para mejorar la información mutua, Russakoff sugirió tener en cuenta una vecindad de píxeles en el cálculo de MI, Marti propuso usar matrices de coeficiente de gris en lugar de histogramas. [11]

2.1.4. Correlation Ratio

El Correlation Ratio mide la dependencia funcional entre dos variables X e Y. Toma valores entre 0 (sin dependencia funcional) y 1 (dependencia puramente determinista). Se define por:

$$\eta\left(X|Y\right) = 1 - \frac{Var\left[\left(Y - \left(Y|X\right)\right)\right]}{Var(Y)} \tag{18}$$

$\eta(X|Y)$ es invariante a los cambios multiplicativos en Y, es decir, $\forall K, \eta(kY|X) = \eta(Y|X)$. A difierencia de la información mutua, la relación de correlación es asimétrica y depende de qué variable se use para predecir la otra, de hecho, las variables no juegan el mismo papel en la relación funcional; eso significa $\eta(Y|X) \neq \eta(X|Y)$. Algunos autores proponen usar un ratio de correlation simétrico para encontrar una forma de evitar este inconveniente al evaluar la varianza condicional normalizada como la suma de dos razones de correlación.

$$\eta_{symetric} = \eta\left(X|Y\right) + \eta\left(Y|X\right) \tag{19}$$

En el artículo [4], podemos encontrar la realización de mosaicos utilizando una medida de similaridad SM, en concreto 35 medidas, de las cuales intenta buscar identificar que medida es la más adecuada para la generación de mosaicos, se generan mediante un algoritmo de block matching, para realización correcta del mosaico, lleva a cabo una transformación rígida global, y una elastica local. Los resultados experimentales en el trabajo de [4] mostraron que un mosaico con buen efecto solo se podía obtiener cuando había una ligera variación en la rotación y en la escala entre las dos imágenes coincidentes. Sin embargo, según las intensidades de píxeles o sus distribuciones

estadísticas, el rendimiento de los métodos no simbólicos se verá influenciado por el ruido complejo del entorno oceánico y las ganancias variacionales. Además, el estudio concluye que de todas las medidas las que mejor resultado dieron fue CR (Correlation Ratio) y MI (Mutual Information), capaces de complementarse una a la otra. [3]

2.2. Métodos simbólicos

Los métodos simbólicos se basan principalmente en la extracción y coincidencia de características. Las características incluyen: formas, bordes, detección de esquinas, etc. Los algoritmos que tienen propiedades de invariancias en rotación, escala, afinidad y perspectiva, han demostrado tener resultados muy positivos [3]. Los algoritmos más conocidos que nos encontramos para estas aplicaciones son: SIFT, SURF y ORB.

Estos algoritmos también han sido usados para navegación, por ejemplo, en el artículo [8], hace uso de SIFT para el registro de imágenes, con el fin de mejorar los errores de navegación de los sistemas inerciales, ya que el registro de imagen proporciona un feedback capaz de usarse para compensar dichos errores. En este caso, no genera un mosaico, pero la conclusión que podemos sacar es que se abren vías para el desarrollo de un sistema complementario capaz de corregir los errores provocados por los sistemas inerciales a partir de imágenes de ultra sonidos.

En algunos casos como en el artículo [8] o [12] hacen uso del algoritmo RANSAC, debido a que se puede hacer uso de difierentes herusitcas para la coincidencia entre imágenes, y usando una heurística como nearest-neighbour tiende a dar muchos falsos positivos, por lo que para poder lidiar con esos outliers y estimar correctamente la transformación del modelo aplican el algoritmo RANSAC.

2.2.1. SIFT

SIFT (Scale-Invariant Feature Transform), es ampliamente usado en visión por computación y fue desarrollado por David Lower [13]. SIFT es capaz de extraer puntos de interés (keypoints) de una imagen aunque esta esté rotada, escalada, sometida a cambios de iluminación, ruido y pequeños cambios de pose o perspectiva. Esto se consigue a partir de características locales, que se almacenan en los descriptores, los descriptores tratan de describir localmente zonas importantes de la imagen con determinadas variables, entre ellas el gradiente.

El algoritmo está estructurado en cuatro fases:

- Scale-space extrema detection
- Keypoint localization
- Orientation assignment
- Keypoint descriptor

Scale-space extrema detection

En esta fase, se busca un primer conjunto de puntos de interés, en las etapas posteriores, estos puntos de interés o keypoints, se irán descartando por no cumplir ciertos requisitos. En esta etapa, la busqueda se realiza sobre todas las localizaciones y todas las escalas de la imagen. Para detectar localizaciones invariantes a cambios de escala, se utiliza la función conocida como scale-space:$L(x, y, \sigma)$, utilizaremos una función gausiana para obtiener dicha función a partir de nuestra imagen original ($I(x, y)$), en la ecuación 20 podemos ver como quedaría la expresión, donde el operador * indica la convoución entre la imagen y la gausiana G.

$$L = (x, y, \sigma) * I\,(x, y) \tag{20}$$

Para calcular todo el espacio $L(x, y, \sigma)$ hay que construir una pirámide gausiana, convolucionando con difierentes filtros $G(x, y, \sigma)$ variando el parámetro σ. Definiremos dos terminos que se hacen uso en esta fase:

- Octava: Conjunto de imágenes del espacio L con el mismo tamaño que difieren en el filtrado σ con el que han sido obtenidas.
- Escala: Conjunto de imágenes del espacio L filtradas con el mismo parámetro σ pero con difierentes tamaños.

Existe una condición que debe cumplirse en este proceso, $\sigma_i = k^{i-1} = 2^{1/n^{\text{o}}escalas-2}$.

Una vez completada toda la pirámide, para detectar puntos de interés estables, se utiliza la función Difference of Gaussian (DoG) ya que es una función que hace una aproximación cercana a escala-normalizada laplaciana gaussiana. DoG se calcula restando imágenes vecinas de una misma octava, ecuación 21.

$$\begin{aligned} D\,(x, y, \sigma) &= (G\,(x, y, k\sigma) - G\,(x, y, \sigma)) * I\,(x, y) \\ &= L\,(x, y, k\sigma) - L\,(x, y, \sigma\,) \end{aligned} \tag{21}$$

Lindeberg demostró que la normalización de los laplacianos con el factor σ^2 es necesaria para la invariancia de escala real. Mikolajczyk (2002) descubrió

que los máximos y mínimos de $\sigma^2\nabla^2$ produce unas características de imagen más estables en comparación con con otras funciones i.e. gradient, Hessian, or Harris corner function.

A partir de los cálculos anteriores, se calculan los máximos y mínimos locales del espacio $D(x, y, \sigma)$ (local extrema detection).

[13]

Keypoint localization

La segunda fase del método SIFT se centra en almacenar toda la información disponible de cada keypoint. Es decir, para cada punto de interés encontrado se guardará a qué escala y octava de la pirámide pertenece, y su posición [fila, columna] dentro de la imagen correspondiente. Esta información permite rechazar puntos que tienen poco contraste (y por lo tanto son sensibles al ruido) o que están mal localizados a lo largo de un borde

Para descartar los puntos con un contraste bajo se utiliza la expresión de Taylor de la función $D(x, y, \sigma)$, hasta el término cuadrático. Si definimos un punto p de los seleccionados como keypoint en el apartado anterior, tal que $p = (x, y, \sigma)^T$, se obtiene la expresión 22

$$D\left(p\right) = D + \frac{\delta D^T}{\delta p}p + \frac{1}{2}p^T\frac{\delta^2 D}{\delta p^2}p \tag{22}$$

El extremo $\hat{p}$ es determinado por la derivada de su función respecto a p y estableciéndola a cero. obtienemos:

$$\hat{p} = -\left(\frac{\delta^2 D^{-1}}{\delta p^2} \cdot \frac{\delta D}{\delta p}\right) \tag{23}$$

La función $D(\hat{p})$ es muy útil para descartar puntos de bajo contraste, esta puede ser obtenida sustituyendo la ecuación 22 dentro de la ecuación 21, dando como resultado la ecuación 24. Para ello, se establece un umbral mínimo al que deben llegar los keypoints para no ser rechazados.

$$D\left(\hat{p}\right) = D + \frac{1}{2}\frac{\delta D^T}{p}\hat{p} \tag{24}$$

La función $D(x, y, \sigma)$ tiene una gran respuesta ante puntos situados sobre los bordes. Muchos de esos puntos no serán suficientemente estables. Un keypoint que esté situado sobre un borde tendrá una respuesta muy pobre en la dirección del borde, pero muy elevada en la dirección perpendicular.

La realización de este computo es el mismo que el que propusieron en 1988 Harris and Stephens. En esta fasa, mucho de los keypoints deducidos en la fase anterior son eliminados, quedando los que realmente son invariantes al cambio.

[13]

Orientation assignment

Esta etapa del algoritmo, se centra en calcular las orientaciones de cada keypoint. Una vez terminada esta etapa, se podrá dar paso a la creación de los descriptores (keypoint descriptor), Al asignar una orientación coherente a cada keypoints en función de las propiedades de la imagen local, el descriptor del keypoint se puede representar en relación con esta orientación y, por lo tanto, lograr la invariabilidad en la rotación de la imagen.

La escala del keypoint es usada para seleccionar la imagen suavizada de Gauss, L, con la escala más cercada, de modo que todos los cálculos se realicen de manera invariable. Para cada muestra de imagen, $L(x, y)$, a esta escala, la magnitud del gradiente, $m(x, y)$, y la orientación, $\theta(x, y)$, precalculada usando la difierencia de pixeles:

$$\begin{aligned} m\left(x,y\right) &= \sqrt{\left(L\left(x+1,y\right)-L\left(x-1,y\right)\right)^2+\left(L\left(x,y+1\right)-L\left(x,y-1\right)\right)^2} \\ \theta\left(x,y\right) &= \tan^{-1}\left(\left(L\left(x,y+1\right)-L\left(x,y-1\right)\right)/\left(L\left(x+1,y\right)-L\left(x-1,y\right)\right)\right) \end{aligned} \tag{25}$$

Una vez realizado el calculo, se genera un histograma a partir de de las orientaciones de gradiente, un para cada keypoint. Los picos más altos de cada histograma son las direcciones dominantes de los gradientes locales, y por tanto, la orientación final, en ocasiones no nos quedaremos con el pico más alto.

[13]

Keypoint Descriptor

En las operaciones anteriores, hemos asignado a cada keypoint una localización, escala y orientación, estos parámetros forman un sistema de coordenadas 2D local, en el que se describe localmente cada región de la imagen, por lo que proporciona la invarianza a esos mismos parámetros.

El siguiente paso es proporcionar la invariabilidad, sobre variaciones como podrían ser cambios de luz o punto de vista 3D. El vector se normaliza a

la unidad, un cambio en el contraste de la imagen en el que cada valor de píxel se multiplica por una constante multiplicará los gradientes por la misma constante, por lo que este cambio de contraste se cancelará mediante la normalización del vector. Un cambio de brillo en el que se agrega una constante a cada píxel de la imagen no afectará los valores de gradiente, ya que se calculan a partir de las difierencias de píxeles. Por lo tanto, el descriptor es invariante para los cambios afines en la iluminación. Sin embargo, los cambios de iluminación no lineales también pueden ocurrir, debido a la saturación de la cámara, o debido a cambios de iluminación que afectan a las superficies 3D, con difierentes orientaciones en difierentes cantidades. Estos efectos pueden causar un gran cambio en las magnitudes relativas de algunos gradientes, pero es menos probable que afecten las orientaciones de los gradientes. Por lo tanto, reducimos la influencia de grandes magnitudes de gradiente al limitar los valores en el vector de características de la unidad para que cada uno no sea mayor que 0.2, y luego renormalizar a la longitud de la unidad. Esto significa que igualar las magnitudes para grandes gradientes ya no es tan importante, y que la distribución de orientaciones tiene mayor énfasis. El valor de 0.2 se determinó experimentalmente usando imágenes que contenían difierentes iluminaciones para los mismos objetos 3D. [13]

Una vez realizadas estas modificaciones, el proceso de construcción de los descriptores se da por finalizado. El método SIFT posteriormente comparará cada uno de los descriptores.

2.2.2. SURF

Speeded up robust features o SURF, es un algoritmo derivado de SIFT, nació para solucionar los problemas que tenía SIFT en tiempo de ejecución, que es donde se hace la extracción de características y la comparación de descriptores, por lo que según su autor en estos aspectos, SURF es más eficiente y robusto.

El algoritmo SURF consta de tres etapas:

- interést Point Detection.
- interést Point Description.
- Search (matching relationship)

No vamos a entrar en detalle dentro de cada punto, ya que el algoritmo es muy similar a SIFT, lo que si se va a describir es en que se difierencian.

En SURF se aproxima el DoG con box filters. En vez de utilizar gaussianas para promediar la imagen, se utilizan cuadrados (aproximaciones). Hacer la

convolución de la imagen con un cuadrado es mucho más rápido si se utiliza la imagen integral, esto se puede hacer en paralelo para difierentes escalas. SURF utiliza un detector BLOB que se basa en la matriz de Hesse para encontrar los puntos de interés, para la asignación de orientación, utiliza respuestas wavelet en direcciones horizontales y verticales mediante la aplicación de pesos gaussianos adecuados. La descripción de la característica, también se utiliza las respuestas wavelet. Se selecciona un vecindario alrededor del punto clave y se divide en subregiones y luego, para cada subregión, las respuestas wavelet se toman y se representan para obtiener el descriptor de característica SURF. El signo de Laplaciano que ya se calcula en la detección se utiliza para los puntos de interés subyacentes, que distingue las manchas brillantes sobre fondos oscuros. En caso de coincidencia, las características se comparan solo si tienen el mismo tipo de contraste (basado en el signo) que permite una coincidencia más rápida. [14]

2.2.3. ORB

Oriented FAST and rotated BRIEF u ORB, creado por Ethan Rublee en 2011 [15], según los autores este algoritmo es más eficiente y robusto que SURF, y por tanto, que SIFT. Resultado de la fusión del detector de puntos de interés FAST y del descriptor BRIEF. El algoritmo no se ve afectado significativamente por el ruido de la imagen. ORB es capaz de utilizarse en tiempo real.

FAST

Features from Accelerated Segment Test o FAST, es una técnica que localiza bordes y esquinas. Este método recibe como parámetro el umbral de la difierencia de intensidad entre el píxel central y aquellos situados en un círculo alrededor del centro. Este detector no produce una medida que cuantifique que tanto un punto puede ser considerado como una esquina, y de esta manera también produce altas respuestas a lo largo de bordes y esquinas. Por esta razón, se emplea una medida empleada en la técnica de detección de esquinas de Harris para ordenar los puntos de acuerdo a su importancia, resaltando aquellos correspondientes a esquinas y obtiener un número determinado de puntos de interés. Así, en el algoritmo debe establecerse un umbral suficientemente bajo para obtiener más de los N puntos de interés deseados, para, una vez hallados, ordenarlos según la medida de Harris y seleccionar únicamente los N puntos más significativos. [16]

BRIEF

Binary Robust Independent Elementary Features o BRIEF, BRIEF es un descriptor binario, basado en simples tests de difierencia de intensidad. Existen varias técnicas propuestas en la literatura para acelerar el proceso de encontrar correspondencias y reducir consumo de memoria, como reducción dimensional (Principal Component Analysis), cuantización usando algunos bits de los descriptores existentes e incluso binarización. Pero todas estas aproximaciones requieren primero la computación del descriptor entero mientras que BRIEF computa directamente los vectores binarios para trozos de imágenes. El ser un descriptor binario es lo que le hace más rápido que otros enfoques distintos como el que usa SIFT o SURF. [16]

En la práctica en nuestro tema sobre la generación de mosaicos con imágenes submarinas a través de medios acústicos, el artículo [12] compara los tres algoritmos, SIFT, SURF y ORB, sobre las distintitas transformaciones que han de hacerse, rotación, translación, escala. ORB se comporta bien en ciertas situaciones, como por ejemplo, donde hay pocas sombras, pero en general SIFT, está por encima, incluso de SURF.

2.3. Métodos no simbólicos vs métodos simbólicos

Como hemos podido leer en las secciones anteriores, y podemos intuir. Los métodos simbólicos son más intuitivos que los métodos no simbólicos, ya que como se han descrito, se enfocan principalmente en dos pasos: un proceso de segmentación asociado con un paso de clasificación para detectar y etiquetar características, y por último un proceso de correspondencia entre dichas características extraídas. Los principales inconvenientes de estos métodos son el tiempo el tiempo requeridopara la segmentación y extracción de características.

Los métodos no simbólicos se basan en información de bajo nivel, mientras que los simbólicos en información de alto nivel. Los métodos no simbólicos, destacan principalmente en su rapidez y su mayor tolerancia hasta cierto punto a distorsiones, variaciones de iluminación y ruido en las imágenes del sonar. [4]

2.4. SLAM

La exploración submarina se ha incrementado significativamente en la última década, en difierentes campos. Esto no es una sorpresa ya que el 71 % de la tierra es agua, y solo el 5 % puede considerarse explorada [**?**], pero más allá de la exploración, también han surgido otras tareas, como puede ser:

inspección de tuberías, inspección de redes, rescate, cartografía, una gran cantidad de campos que darían para un libro. La mayoría de estas tareas en la última década se han estado realizando con AUVs o ROVs, de aquí la importancia de tener un buen sistema de posicionamiento.

El papel que juega los sonars acusticos en los equipos submarinos, es vital ya que nos permite no solo ver el fondo cuando no hay luz, sino que como hemos hablado en las subsecciones anteriores, pueden ayudarnos a mejorar la navegación, Simultaneous Localization and Mapping o SLAM, tiene como propósito construir un mapa del entorno y usar ese mismo mapa para localizar. Se han desarrollado técnicas SLAMs para trabajar con sistemas de sonars de barrido lateral [17]. Por ello, la función principal de conseguir generar estos mosaicos es poder referenciarlos, y con esa referencia no solo explorar el fondo, sino más bien llegar a usarlos para la localización.

Types of dependency	Similarity measures
Second order dependency [25]	Correlation
	Sum of Absolute Differences (SAD)
	Sum of Squared Differences (SSD)
	Correlation coefficient of Moravec
	Median of grey level differences
	Least median of squared differences
	Seitz measure
	Variance of Squared Differences (VOSD)
	Kurtosis
Linear dependency [25]	Zero Normalized Cross Correlation (ZNCC)
	Normalized Covariance (NCOV)
	Cross correlation coefficient
	Zero Normalized Sum of Squared Differences (ZNSSD)
	Smooth Median Absolute Deviation (SMAD)
	M-estimator (Geman MacClure)
	Pseudo-norm ($\alpha = 0.5$)
	Phase correlation [28]
Functional dependency [26]	Woods criterion
	Correlation ratio
Law dependency [27]	Dissimilarity of χ^2
	Distance to independence
	Mutual Information
	Kolmogorov distance
	Kullback Leibler divergence
	K-divergence of Lin
	L-divergence of Lin
	Hellinger distance
	Toussaint distance
	W-divergence of Kagan
	Matusita distance
	Battacharya coefficient
	χ^{α} divergence of Vajda
	α order information of Bose Einstein
	α order information of Rényi ($\alpha = 3$)
	α order information of Femi Dirac
	Cluster Reward Algorithm (CRA) [13]

Cuadro 1: Medidas de Similitud (Similarity Measure) [4]

3. Objetivo, metodologías y herramientas

Referencias

[1] Bennell James D. . Mosaicing of sidescan sonar images to map seabed features. En: Jon Davies. Marine Monitoring Handbook. 2001. PG1-5.

[2] Zitová Barbara , Flusser Jan. Image Registration Methods. En: Todorovic S. Image and Vision Computing. 21, 2003, PG 977-1000.

[3] Zhao Jianhu, Wang Aixue, Zhang Hongmei, Wang Xiao. Mosaic method of side-scan sonar strip images using corresponding features. En: IET. 6, 2013, PG616-623

[4] Chailloux Cyril, Le Caillec Jean-Marc, Gueriot Didier, Zerr Benoit. Intensity-Based Block Matching Algorithm for Mosaicing Sonar Images. En: IEEE. 4, 2011, PG627-644.

[5] Horn K.P., Schunck G. Brian. Determining Optical Flow. En: ScienceDirect. Artificial Intelligence. 1, 1981, PG185-203

[6] Salgado de la Nuez Agustín Javier. Métodos Variacionales para la Estimación del Flujo Óptico y Mapas de Disparidad [master's thesis]. Lugar: Universidad de las Palmas de Gran Canaria. 2010. 217P.

[7] Barron J. L. Beauchemin S. S. . The Computation of Optical Flow. En: ACM Computing Surveys. 1995, 35P.

[8] Vandrish P., Vardy A., Walker D., Dobre A. O. .Side-scan Sonar Image Registration for AUV Navigation. 6. P1-7.

[9] Pazmiño Reyes Ricardo Esteban. Implementación y comparación de los algoritmos de determinación de flujo óptico de Horn-Schunck y Lucas-Kanade para rastreo de objetos. En: Universidad San Fransico De Quito. 46P.

[10] Lucas Bruce D., Kanade Takeo. An Iterative Image Registration Technique with an Application to Stereo Vision. Proceedings DARPA Image Understanding Workshop. 1981. P674-679.

[11] Chailloux Cyril, Zerr Benoit. Non Symbolic Methods to register sonar images. En: Oceans. 1. 2005. PG276-281.

[12] Dhana Lakshmi M., Vimal Raj M., Sakthivel Murugan S. . Feature Matching and Assessment of Similarity rate on geometrically Distorted Side Scan Sonar Images. En: TEQIP III Sponsored International Conference on Microwave Integrated Circuits, Photonics and Wireless Networks (IMICPW). 2019, PG208-212.

[13] Lower G. David. Distinctive Image Features from Scale-Invariant Keypoints. En: Int. J. Comput. Vision. 60. 2004, PG91-110.

[14] Akalanka Perera Shehan . A Comparison of SIFT , SURF and ORB. 2018. Disponible en: https://medium.com/@shehan.a.perera/a-comparison-of-sift-surf-and-orb-333d64bcaaea

[15] E. Rublee, V. Rabaud, K. Konolige and G. Bradski, .ºRB: An efficient alternative to SIFT or SURF,"2011 International Conference on Computer Vision, Barcelona, 2011, pp. 2564-2571.

[16] Godoy Olivera Yesmar Andrés, Ducuara Oyuela Álvaro Isledier. ANÁLISIS COMPARATIVO DE LAS TÉCNICAS SURF Y ORB PARA LA DETECCIÓN DE PUNTOS DE INTERÉS EN FOTOGRAFÍAS AÉREAS [master's thesis]. Lugar: Universidad de Ibagué. 2019. 71P.

[17] Reed S., Ruiz I. T., Capus C., Petillot Y. . The fusion of large scale classified side-scan sonar image mosaics. En: IEEE Transactions on Image Processing, vol. 15, 7. 2006. PG2049-2060.

[18] Palomer, A., Ridao, P., Ribas, D. . Multibeam 3D Underwater SLAM with Probabilistic Registration. Sensors 2016, 16, 560.