ESTIMACIÓN MONOCULAR Y EFICIENTE DE LA POSE USANDO MODELOS 3D COMPLEJOS

A. Rubio, M. Villamizar, L. Ferraz, A. Penate-Sanchez, A. Sanfeliu y F. Moreno-Noguer
Institut de Robòtica i Informàtica Industrial, CSIC-UPC
Llorens Artigas 4-6, 08028 Barcelona, Spain

Resumen

El siguiente documento presenta un método robusto y eficiente para estimar la pose de una cámara. El método propuesto asume el conocimiento previo de un modelo 3D del entorno, y compara una nueva imagen de entrada únicamente con un conjunto pequeño de imágenes similares seleccionadas previamente por un algoritmo de "Bag of Visual Words". De esta forma se evita el alto coste computacional de calcular la correspondencia de los puntos 2D de la imagen de entrada contra todos los puntos 3D de un modelo complejo, que en nuestro caso contiene más de 100,000 puntos. La estimación de la pose se lleva a cabo a partir de estas correspondencias 2D-3D utilizando un novedoso algoritmo de PnP que realiza la eliminación de valores atípicos (outliers) sin necesidad de utilizar RANSAC, y que es entre 10 y 100 veces más rápido que los métodos que lo utilizan.

Palabras clave: Estimación de la pose · Robust Efficient Procrustes Perspective-*n*-point · Bag of Visual Words

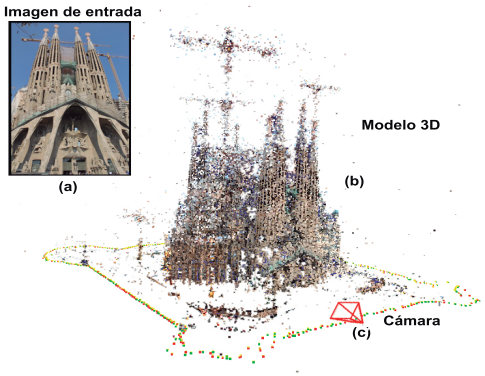


Figura 1: Definición del problema: Dada una imagen de entrada (a) y un modelo 3D del entorno conocido (b), el problema consiste en encontrar la pose de la camára que capturó la imagen con respecto al modelo (c). La principal dificultad que se aborda en este trabajo es realizar la correspondencia de puntos de forma eficiente y fiable para modelos 3D complejos, los cuales pueden tener una gran cantidad de puntos asociados. Para nuestro caso, el modelo de la Sagrada Familia contiene más de 100,000 puntos.

1. Motivación

El presente trabajo es una contribución científica al proyecto Europeo ARCAS [3], cuyo objetivo principal es el diseño y desarrollo de robots voladores para llevar a cabo tareas cooperativas de ensamblaje en entornos interiores y exteriores. Dentro de este proyecto también se aborda el desarrollo de los algoritmos de percepción, navegación y control necesarios para que estos vehículos áereos no tripulados puedan realizar las tareas de forma autónoma y segura.

Un elemento importante para realizar satisfactoriamente estas tareas corresponde con a la localización autónoma de los robots dentro de su entorno de trabajo. Este aspecto ha sido abordado en los últimos años desde diferentes puntos de vista. Por un lado se encuentran los sistemas basados en cámaras de infrarrojos y de alta frecuencia tales como el sistema VICON [31], el cual ha demostrado unos excelentes resultados para la localización y navegación de robots. No obstante, estos sistemas están limitados a entornos interiores donde las condiciones de iluminación estén controladas. Además, aparte de un alto coste, estos sistemas requieren la implantación de una red de cámaras dentro del entorno, hecho que restringe nuevamente el sistema a aplicaciones a escenarios estructurados.

Otra alternativa para localizar el robot de forma robusta es dotarlo de múltiples sensores, como por ejemplo cámaras estéreo o láseres, y realizar posteriormente una fusión de los datos provenientes de cada uno de ellos. Aunque estos sistemas aumentan la fiabilidad del robot, tienen un impacto negativo en su capacidad de carga y computacional. Esto es un aspecto muy crítico para ejecutar las tareas, especialmente en robots pequeños y de bajo coste.

Por el contrario, en este documento se propone un sistema eficiente y robusto basado simplemente en un cámara monocular y un modelo 3D del entorno conocido, como se muestra en la Fig. 1. En particular, el método propuesto es capaz de estimar la pose (rotación y traslación) de la cámara de manera rápida usando modelos altamente complejos que contienen un número muy alto de puntos.

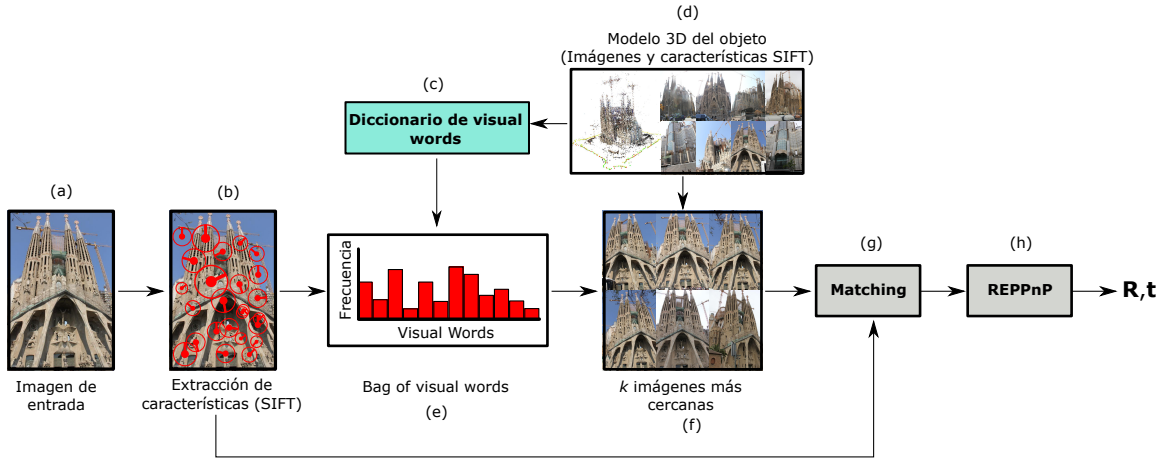


Figura 2: Diagrama general del método propuesto para estimar la pose de la cámara de forma eficiente y robusta con respecto a un modelo 3D conocido.

2. Introducción y estado del arte

El problema de la estimación de la pose de la cámara se ha abordado desde una gran cantidad de enfoques, desde la estimación basada en marcadores visuales [1, 4] hasta la utilización de técnicas 'menos invasivas' basadas en características o descriptores de la imagen [24, 27]. Este problema supone la base para numerosas aplicaciones en robótica y visión por computador, como la realidad aumentada o la fotogrametría [22].

El objetivo del problema Perspective-*n*-Point (P*n*P) es determinar la pose de la cámara (posición y orientación) a partir de un conjunto de n correspondencias 2D-3D conocidas entre la imagen y un modelo 3D. A pesar de haber sido estudiado durante más de un siglo (la primera solución data de 1841 [14]), recientemente la comunidad científica se ha esforzado en la búsqueda de alternativas precisas y eficientes para este problema.

En este contexto, múltiples métodos han resuelto el problema. No obstante, la mayoría de ellos presentan un alto coste computacional, alta sensibilidad al ruido o son válidos solamente para valores bajos de n. Entre estos métodos, cabe mencionar los diseñados para resolver el problema de P3P [18, 13], los métodos diseñados para valores grandes de n [8, 17, 15], los enfoques centrados en la reducción de complejidad [11], o los métodos más precisos, que son también los más complejos [2, 26].

La primera solución cerrada con coste *O(n)* fue el Efficient P*n*P [19, 23] (nombrado EP*n*P de aquí en adelante). La principal contribución de este método fue la introducción de unos puntos de control en 3D, reduciendo el problema a la estimación de su posición con técnicas de linearización. Trabajos posteriores han mejorado la precisión de este método a través de reformular el problema P*n*P original [16, 33, 32] y sustituir el enfoque lineal por soluciones polinomiales [20].

Para mejorar la robustez de EP*n*P en la estimación de la pose de la cámara, este método puede combinarse con una técnica para eliminar outliers, como por ejemplo RANSAC [12]. En aras de incrementar la eficiencia y evitar tareas con un elevado coste computacional, se ha seleccionado un método reciente que lleva a cabo el rechazo de outliers al mismo tiempo que estima la pose. Este método, llamado Robust Efficient Procrustes P*n*P (REPP*n*P) [9], reduce considerablemente el tiempo empleado en el proceso gracias a esta simultaneidad en los cálculos. La precision de este metodo ha sido mejorada en [10] considerando la incerteza asociada a cada correspondencia.

En el presente trabajo se presupone la existencia de un modelo 3D conocido. Esto quiere decir que se conocen los descriptores de m puntos 3D. En el caso de estudio, estos descriptores son los SIFT [21] de los m puntos de interés, obtenidos a partir de un conjunto de imágenes de entrenamiento que se ha usado para la construcción del modelo. Así, los descriptores de una imagen de entrada serán comparados con los descriptores del modelo, obteniendo una serie de correspondencias entre puntos 2D y 3D. Este proceso puede conllevar un elevado coste computacional, ya que el número m de puntos 3D puede ser extremadamente alto (observar la Fig. 1).

A fin de evitar la comparación con un número excesivo de puntos, se ha añadido una etapa previa, consistente en un *Bag of Visual Words* (BoVW), un método de clasificación usado para obtener un cierto número k de imágenes similares a la de entrada. Refiérase a la Fig. 2. Estas imágenes provienen de las usadas para la construcción del modelo 3D. De este modo, se compararán los descriptores de ésta con los puntos 3D del modelo que aparecen únicamente en las k imágenes similares, aliviando notablemente el coste computacional.

En el resto del artículo se describen cada uno de los componentes del método que se propone. Primero se describe la construcción del modelo 3D y después el algoritmo de estimación de la pose, con sus dos partes principales: el bag of visual words, para obtener una primera estimación muy cruda de la pose, y el REPP-nP, para el posterior refinamiento. Para evaluar el funcionamiento del método, finalmente se presenta una serie de experimentos con un modelo de la Basílica de la Sagrada Familia con $m = 100,532$ puntos. Obsérvese que calcular las correspondencias entre los puntos de una imagen de entrada y más de $100,000$ puntos de referencia al mismo tiempo que se eliminan outliers es un desafío de gran magnitud.

3. Construcción del modelo 3D

El modelo 3D de la Sagrada Familia se ha construido con el sistema *Bundler* [28], desarrollado para determinar la estructura de objetos o entornos a partir de movimiento y colecciones de imágenes desordenadas. Este algoritmo extrae los descriptores SIFT de un conjunto de N imágenes de un mismo escenario u objeto, para después buscar las correspondencias y construir a partir de ellas el modelo final, a la vez que se recuperan las poses desde donde fueron tomadas las imágenes.

Para la construcción de este modelo se utilizaron 478 imágenes, obteniendo un total de $100,532$ puntos. En la Fig. 1 (b) se muestra el modelo 3D recuperado, incluyendo en amarillo la pose recuperada por *Bundler* para cada una de las imagenes de entrenamiento, así como una imagen de entrada (Fig. 1 (a)) y la pose correspondiente de la cámara en rojo (Fig. 1 (c)). Para cada punto 3D del modelo, *Bundler* proporciona la siguiente información:

- Descriptor SIFT
- Coordenadas 3D
- Color RGB
- Imágenes en las que el punto aparece

La siguiente información sobre cada imagen también es devuelta por el programa:

- Matriz de calibración y factor de distorsión
- Matriz de rotación, $\mathbf{R}$
- Vector de traslación, $\mathbf{t}$

Esta información es usada como valor "ground truth" a la hora de evaluar la precisión del algoritmo.

4. Algoritmo de estimación de la pose

La Fig. 2 muestra el diagrama general de los pasos para determinar la pose de la cámara ($\mathbf{R}$ y $\mathbf{t}$) a partir de una imagen de entrada (Fig. 2 (a)). Como paso previo y dado el conjunto de $N = 478$ imágenes usadas para construir el modelo, correspondiente a imágenes del escenario u objeto adquiridas desde diferentes puntos de vista, se crean dos subconjuntos disjuntos y equitativos de imágenes. Un subconjunto de entrenamiento usado para construir el diccionario de visual words (Fig. 2 (c)), y otro subconjunto conteniendo imágenes de prueba para evaluar el desempeño y la precisión del método propuesto. Por lo tanto, para evitar falsear los resultados, cuando se calcule la pose de una imagen de prueba se tendrán en cuenta únicamente los puntos del modelo 3D pertenecientes al conjunto de entrenamiento.

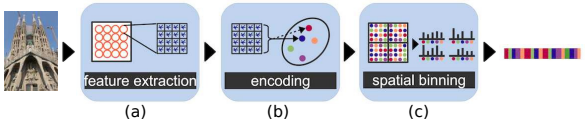


Figura 3: Pasos del algoritmo BoVW. Figura inspirada en [6].

Una vez computado el diccionario de visual words, el siguiente paso consiste en calcular los descriptores SIFT (Fig. 2 (b)) sobre la imagen de entrada (perteneciente al conjunto de prueba). Estos descriptores son usados después para codificar la apariencia de la imagen mediante el cómputo del histograma de Bag of Visual Words (BoVW) (Fig. 2 (e)).

A continuación, este BoVW es usado para encontrar las imágenes más cercanas en el conjunto de entrenamiento. Esto se realiza mediante el algoritmo de k-vecinos más cercanos usando los histogramas de BoVW como muestras en un espacio multidimensional (Fig. 2 (f)). La ventaja de este enfoque como paso previo a la correspondencia de los puntos 2D-3D (Fig. 2 (g)), es que acota considerablemente la posible pose de la cámara, y por consiguiente, reduce el número de comparaciones entre los descriptores de la imagen de entrada y todos los m descriptores de modelo completo. Esto significa una reducción importante en el coste computacional de estimar la pose de la cámara para modelos complejos.

El último paso consiste en realizar el REPP*n*P (Fig. 2 (h)) con las coordenadas 2D de los descriptores SIFT de la imagen de entrada y los puntos 3D del modelo que aparecen en las k imágenes más cercanas para estimar la rotación $\mathbf{R}$ y la traslación $\mathbf{t}$ de la cámara con respecto al modelo 3D.

4.1. Bag of visual words (BoVW)

Como se ha mencionado anteriormente, el algoritmo de Bag of Visual Words en combinación con el algoritmo de k-vecinos más cercanos se usa para tener una primera aproximación de la parte del modelo que se ve en la imagen de entrada. En la Fig. 3 se observan los pasos realizados para computar el BoVW sobre una imagen de entrada. No obstante, es indispensable crear

Cuadro 1: Valores medios del método propuesto y de la comparacion con el modelo 3D completo.

	Modelo completo	$k = 6$	$k = 8$	$k = 12$	$k = 16$
Tiempo (s)	45,6694	12,3344	14,1634	16,9470	20,2644
Número de puntos 3D	100532	24855	29046	35422	43005
Número de puntos relacionados	470	424	436	447	452
Error de rotación	0,0061	0,0059	0,0135	0,0150	0,0056
Error de traslación	0,0231	0,0203	0,0323	0,0294	0,0208

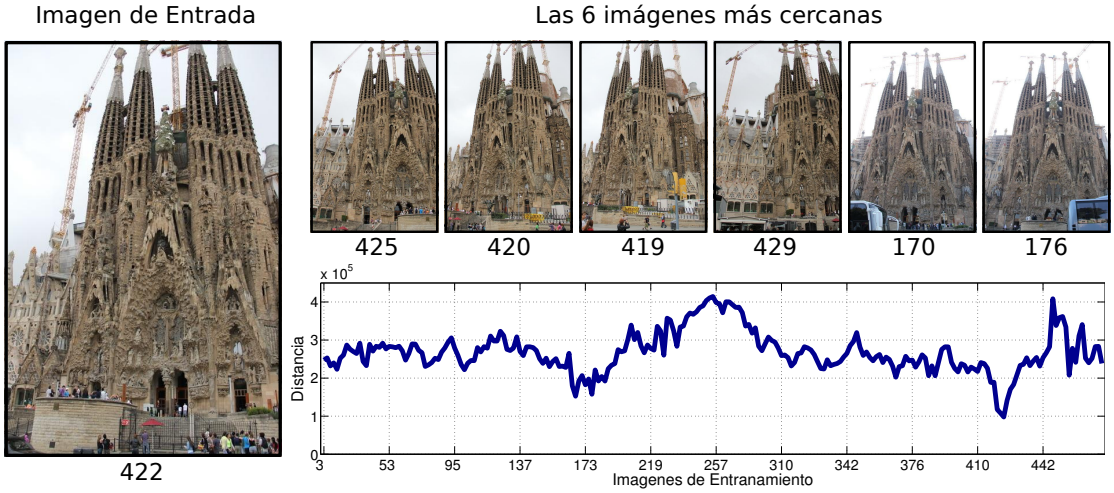


Figura 4: Imagen de entrada 1 (izqda.), imágenes similares según el BoVW ($k = 6$) y distancia a todas las imágenes de entrenamiento (abajo).

previamente un diccionario de visual words con el cual describir la apariencia de esta imagen de entrada. Más específicamente, este diccionario se genera agrupando los descriptores PHOW (Pyramid Histogram Of visual Words) [5] de las imágenes del conjunto de entrenamiento mediante el algoritmo *k-means*.

Los descriptores SIFT de la imagen de entrada (extraídos en una primera etapa, como se observa en la Fig. 3 (a)) se asignan a los correspondientes visual words del diccionario (fase de *encoding*, Fig. 3 (b)) por medio de cuantización VQ (que asigna cada SIFT al centroide de un visual word más cercano en distancia Euclídea) y por codificación Fisher [25]. Finalmente, tal como se recomienda en [29], se utilizan histogramas espaciales (Fig. 3 (c)) como forma de introducir en el proceso información espacial de la posición de las características dentro de la imagen.

Para el presente trabajo se ha computado un vocabulario con 100 visual words, dividiendo las imágenes en 8 histogramas espaciales.

4.2. Eliminación de outliers y estimación de la pose con REPPnP

El paso final del algoritmo presentado en este trabajo consiste en relacionar los puntos 2D de la imagen de entrada con los correspondientes puntos 3D del modelo a través de sus descriptores SIFT. Como ya ha sido mencionado en la sección 2, es frecuente valerse de métodos como el EP*n*P. Las soluciones clásicas del P*n*P asumen normalmente correspondencias 2D-3D corrompidas por ruido, pero no por puntos mal relacionados. Estos puntos (outliers) son rechazados habitualmente en una etapa previa por medio del RANSAC combinado con algoritmos P3P [18]. Estos enfoques implican el cálculo total de la pose y la reproyección de un gran número de puntos en cada iteración del método.

El REPP*n*P consiste en un método iterativo igualmente, pero que usa un criterio mucho más directo que el geométrico usado por los métodos previos. Este método considera el error algebraico del sistema lineal derivado de la formulación del P*n*P y demuestra convergencia con hasta un 50 % de outliers mientras reduce hasta 100 veces el tiempo empleado por las estrategias clásicas P3P-RANSAC-PnP.

Con el método propuesto se pasa de tener que comparar de forma exhaustiva los descriptores de la imagen de entrada contra todos los $100,532$ puntos 3D del modelo a comparar contra una cuarta parte de ellos (para $k = 6$), y encontrando de entre éstos más de 400 correspondencias 2D-3D de media. Los valores medios del número de puntos 3D que se consideran para la comparación y del número de correspondencias encontradas se muestran en el cuadro 1.

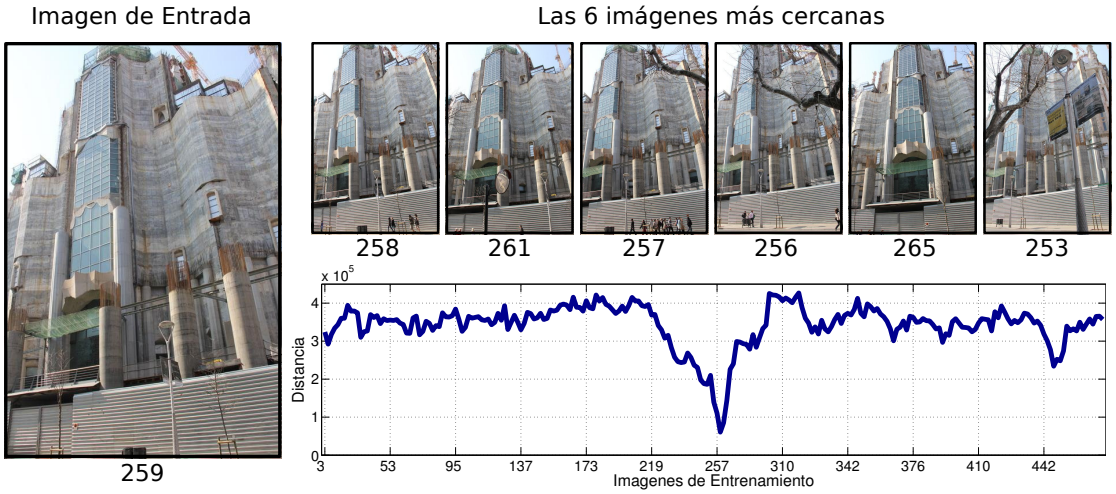


Figura 5: Imagen de entrada 259 (izqda.), imágenes similares según el BoVW ($k = 6$) y distancia a todas las imágenes de entrenamiento (abajo).

5. Resultados

En esta sección, la eficiencia y la precisión del método propuesto son evaluadas. Previamente, se estudia el desempeño del BoVW para la selección de imágenes similares.

5.1. Selección de imágenes similares

El desempeño del proceso de selección de imágenes similares usando el BoVW y el algoritmo k-vecinos más cercanos se muestra en las Fig. 4 y 5. En la parte izquierda se observa la imagen de entrada, mientras que en la parte derecha aparecen las 6 imágenes más cercanas de acuerdo con las distancias mostradas en la parte inferior. Nótese que las imágenes seleccionadas han sido adquiridas desde poses muy similares.

5.2. Eficiencia del método propuesto

El cómputo total del tiempo del proceso se desglosa en cinco componentes distintos: el tiempo que se tarda en seleccionar las imágenes más cercanas usando BoVW, el tiempo que se tarda en seleccionar los descriptores SIFT de esas imágenes que aparecen en el modelo 3D para la posterior comparación, el tiempo de extracción de los descriptores SIFT de la imagen de entrada, el tiempo usado para relacionar esos descriptores con los correspondientes del modelo 3D y, finalmente, el tiempo empleado por el algoritmo REPP*n*P para estimar la pose. Se ha medido también el tiempo que emplea el algoritmo RANSAC combinado con OP*n*P en estimar la pose tras la etapa de BoVW y k-vecinos más cercanos. Este último tiempo no forma parte del proceso propuesto y se incluye únicamente para comparar con la estrategia que se usa actualmente en los métodos del estado del arte. Cabe destacar que el OP*n*P ha sido presentado recientemente en [32], siendo uno de los algoritmos P*n*P más rapidos y precisos que existen.

Se han medido los tiempos de cálculo de cada una de las etapas del proceso para las 239 imágenes de entrenamiento. Los valores medios de esos tiempos se muestran en la tabla 1 y en la Fig. 6 (arriba izqda.). Se puede apreciar una reducción significativa de la duración del proceso de estimación gracias al enfoque de BoVW aplicado en la etapa previa. Otro hecho apreciable en las figuras es el aumento del tiempo con k, lo cual está directamente relacionado con el aumento del número de puntos comparados, como se explica a continuación.

La Fig. 6 (arriba izqda. y dcha.) muestra que el tiempo más relevante es el utilizado para relacionar los SIFT de la imagen de entrada con los del modelo (etapa de "*matching*"). La Fig. 6 (abajo izqda.) muestra el porcentaje en el cual se reduce este tiempo para distintos valores de k tomando como referencia el tiempo empleado en realizar dicha operación con el modelo completo. Este tiempo se reduce hasta en un 75 % (para $k = 6$) con el algoritmo propuesto dado que se pasa de comparar con 100,532 puntos a comparar con un número mucho menor. El cuadro 1 muestra el número de puntos 3D que aparecen en las imágenes seleccionadas por el BoVW (puntos considerados para relacionar con los de la imagen) y el número de estos puntos para los que se han encontrado correspondencias con los de la imagen de entrada para diferentes valores de k (valores medios para todas las imágenes de prueba).

Por otra parte se puede observar que las otras operaciones envueltas en el método propuesto tales como el cómputo de BoVW, la selección de SIFT, REPP*n*P o RANSAC combinado con OP*n*P muestran tiempos despreciables. Para una mejor visualización refiérase a la Fig. 6 (arriba dcha.) que muestra un zoom de los tiempos empleados.

Finalmente, en la Fig. 6 (abajo dcha.) se muestra una comparación de los tiempos empleados por REPPnP y

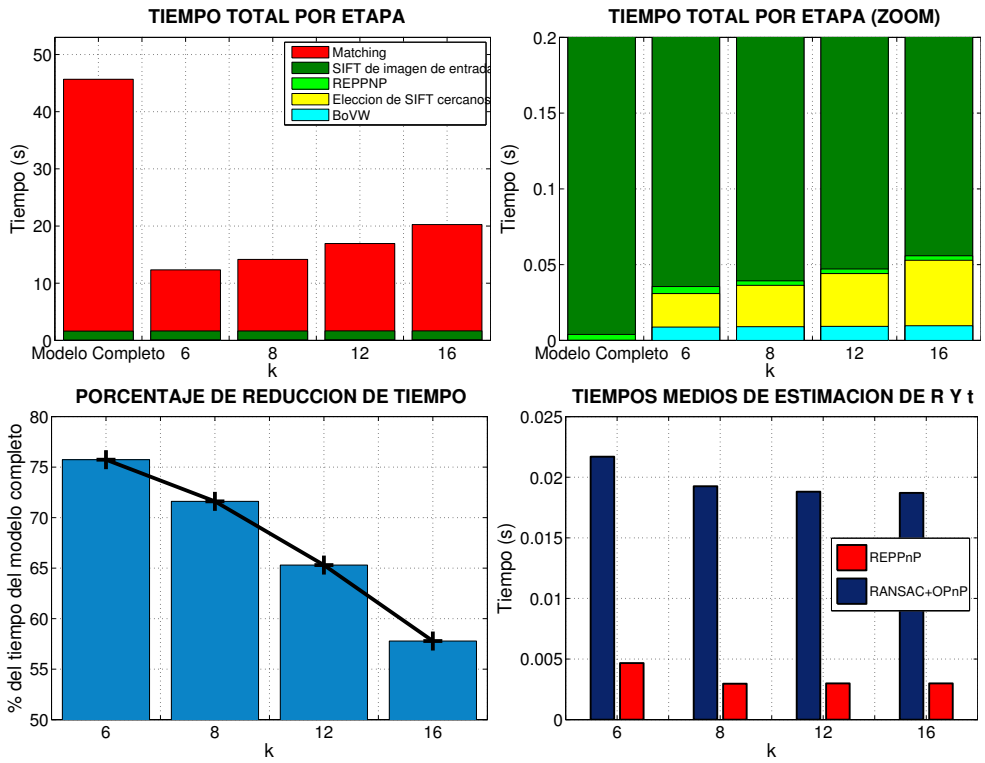


Figura 6: Análisis del tiempo de cálculo: Arriba-izquierda: Tiempos de cálculo del algoritmo propuesto para diferentes valores de k. El caso *modelo completo* es equivalente a $k = 1$ Obsérvese que la mayor parte del tiempo se invierte en calcular un conjunto inicial de correspondencias, entre las cuales aún puede haber outliers. Arriba-derecha: Zoom para una mejor visualización de los tiempos empleados para calcular el BoW y 'elegir los SIFT más cercanos'. Abajo-izquierda: Porcentaje de reducción de tiempo en la etapa de "*matching*" con respecto al modelo completo. Para $k = 6$, por ejemplo, se observa una reducción del 75 %. Abajo-derecha: Tiempo necesario para calcular la pose y eliminar outliers con REPPnP (usado en el artículo) y RANSAC+OPnP, que corresponde a la estrategia utilizada habitualmente. En media REPPnP ofrece un incremento de velocidad de hasta $10\times$.

RANSAC junto a OPnP para la estimación de la pose, en la cual se aprecia que el primer método reduce considerablemente la duración de este proceso. El tiempo empleado por ambos algoritmos se reduce al aumentar el valor de k.

5.3. Precisión del método propuesto

Con respecto a la precisión del método, el error de estimación se calcula como el error en la rotación (norma L_2 entre cuaterniones) y el error en la traslación (distancia Euclídea), ambos calculados tomando como "ground truth" la **R** y la **t** proporcionadas por el algoritmo *Bundler* mencionado en la sección 3. Como puede verse en la Fig. 7, para los valores de $k = 6$ y $k = 16$ el método mejora el error en la estimación al ser comparado con el uso del modelo completo con m puntos 3D. En el resto de experimentos realizados el error es algo mayor. No obstante, en todos los experimentos se observa una reducción muy significativa frente al error obtenido empleando RANSAC y OP*n*P.

En la Fig. 8 se observa la localización de los descriptores SIFT de la imagen de entrada y la reproyección de los puntos 3D asociados a esos descriptores, así como los errores de rotación y traslación asociados a cada método. La reproyección se ha llevado a cabo con las **R** y **t** obtenidas por medio del método propuesto con REPP*n*P y con las obtenidas con RANSAC combinado con OP*n*P. En dicha figura se presentan tres casos concretos para recalcar de nuevo las ventajas del uso del REPP*n*P. En todos ellos este método presenta una reproyección correcta, mientras que el RANSAC combinado con OP*n*P provoca reproyecciones erróneas para ciertas imágenes.

6. Conclusiones y trabajos futuros

La estimación de la pose de la cámara con respecto a un modelo de grandes dimensiones se ha acelerado notablemente gracias al BoVW para aproximar la región de la que proviene la imagen en un primer paso, y al mismo tiempo, el error en el resultado mejora con respecto al obtenido comparando con el modelo total para ciertos valores de k, por lo que simplificar el problema no provoca una pérdida de precisión. De ese modo, una reducción del 75 % en tiempo de cálculo es una ventaja importante a la hora de utilizar modelos formados por un elevado número de puntos.

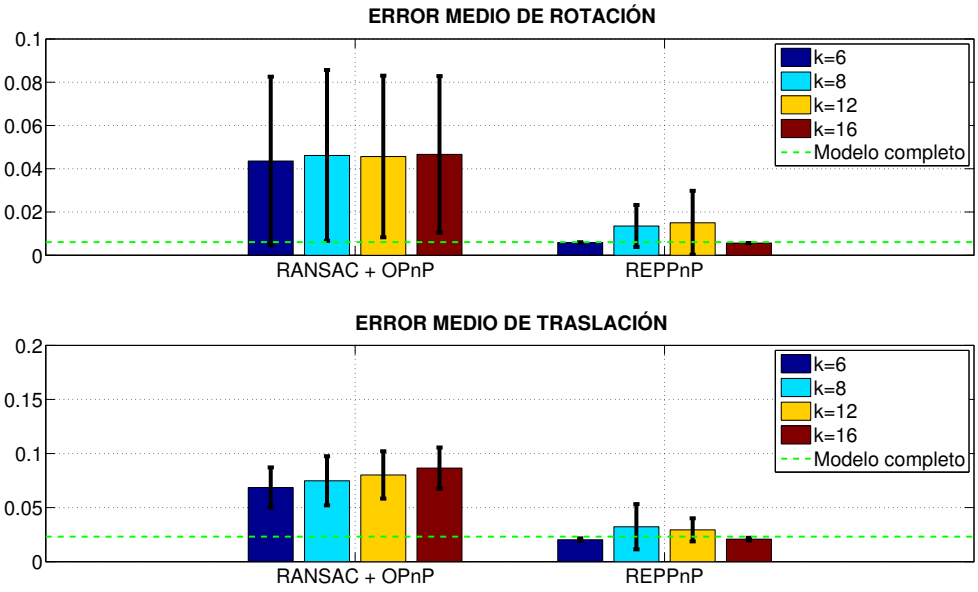


Figura 7: Error de rotación (gráfica superior) calculado como la norma L_2 entre cuaterniones y error de traslación (gráfica inferior) en la estimación de R para la comparación con el modelo completo y para diferentes valores de k.

Como trabajo futuro, el método propuesto puede acelerarse aun más si se sustituyen los descriptores SIFT por algún otro tipo de descriptor más rápido (como BRIEF o FAST). También se considerará el uso de técnicas eficientes de detección basadas en árboles de decisión, boosting y descriptores binarios [34, 35, 36, 37] para estimar la pose inicial de la cámara a partir de la información de apariencia. Finalmente, también se trasladará la implementación del algoritmo de MATLAB a OpenCV y C++, con el objetivo de lograr estimación de la pose en tiempo real.

Agradecimientos

Este trabajo ha estado financiado en parte por los proyectos RobTaskCoop DPI2010-17112, ERA-Net Chistera ViSen PCIN-2013-047, y por el proyecto EU ARCAS FP7-ICT-2011-2876. Agradecemos tambien a Adrián Peñate la cesión del modelo 3D de la Sagrada Familia y los datos correspondientes provenientes de *Bundler*.

Referencias

[1] A. Amor-Martinez, A. Ruiz, F. Moreno-Noguer and A. Sanfeliu, On-board Real-time Pose Estimation for UAVs using Deformable Visual Contour Registration. In ICRA, 2014.

[2] A. Ansar and K. Daniilidis. Linear pose estimation from points or lines. In PAMI, 2003.

[3] European project ARCAS: Aerial Robotics Cooperative Assembly System (www.arcas-project.eu).

[4] Biblioteca de realidad aumentada ARToolKit (http://www.hitl.washington.edu/artoolkit/).

[5] A. Bosch, A. Zisserman and X. Munoz. Image classifcation using random forests and ferns. In ICCV, 2007.

[6] K. Chatfield, V. Lempitsky and A. Vedaldi and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In BMVC, 2011.

[7] D. DeMenthon and L. Davis. Exact and approximate solutions of the perspective-three-point problem. In PAMI, 1992.

[8] M. Dhome, M. Richetin, J.T. Lapreste and G. Rives. Determination of the attitude of 3d objects from a single perspective view. In PAMI, 1989.

[9] L. Ferraz, X. Binefa and F Moreno-Noguer. Very Fast Solution to the PnP Problem with Algebraic Outlier Rejection. In CVPR, 2014.

[10] L. Ferraz, X. Binefa and F. Moreno-Noguer. Leveraging Feature Uncertainty in the PnP Problem, In BMVC, 2014.

[11] P.D. Fiore. Efficient linear solution of exterior orientation. In PAMI, 2001.

[12] M.A. Fischler and R.C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications ACM, 1981.

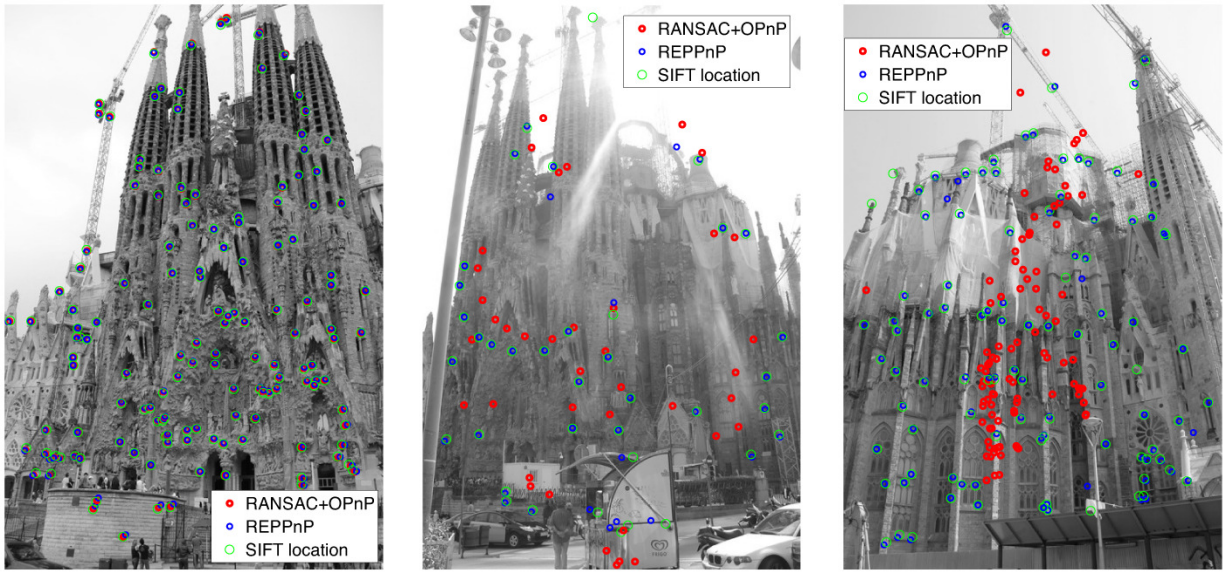
[13] X.S. Gao, X.R. Hou, J. Tang and H.F. Cheng. Complete solution classification for the perspective-three-point problem. In PAMI, 2003.

[14] J. A. Grunert. Das pothenotische problem in erweiterter gestalt nebst über seine anwendungen in geodäsie. In Grunerts Archiv für Mathematik und Physik, 1841.

[15] R.M. Haralick, D. Lee, K. Ottenburg and M. Nolle, M. Analysis and solutions of the three point perspective pose estimation problem. In CVPR, 1991.

[16] J.A. Hesch and S.I. Roumeliotis. A direct least-squares (DLS) method for P*n*P. In ICCV, 2011.

[17] R. Horaud, B. Conio, O. Leboulleux and B. Lacolle. An analytic solution for the perspective 4-point problem. Computer Vision, Graphics, and Image Processing, 1989.



Errores de Rotación					
REPnP:	0,0068	REPnP:	0,0042	REPnP:	0,0045
RANSAC+OPnP:	0,0007	RANSAC+OPnP:	0,0812	RANSAC+OPnP:	0,6845
Errores de Traslación					
REPnP:	0,0306	REPnP:	0,0109	REPnP:	0,0094
RANSAC+OPnP:	0,0383	RANSAC+OPnP:	0,1497	RANSAC+OPnP:	0,6825

Figura 8: Para 3 imágenes del conjunto de prueba: posiciones originales de los puntos SIFT de la imagen, y su reproyección con las **R** y **t** recuperadas por el algoritmo con REPPnP con $k = 6$ y con RANSAC + OPnP

[18] L. Kneip, D. Scaramuzza and R. Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In CVPR, 2011.

[19] V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnp: An accurate o(n) solution to the pnp problem. In IJCV, 2009.

[20] S. Li, C. Xu and M. Xie. A robust o(n) solution to the perspective-n-point problem. In PAMI, 2012.

[21] D. Lowe. Distinctive image features from scale-invariant keypoints. In IJCV, 2004.

[22] C. McGlove, E. Mikhail and J. Bethel. (Eds.). Manual of photogrametry. American society for photogrammetry and remote sensing, 2004.

[23] F. Moreno-Noguer, V. Lepetit and P. Fua. Accurate noniterative o(n) solution to the pnp problem. In ICCV, 2007.

[24] F. Moreno-Noguer, V. Lepetit and P. Fua. Pose Priors for Simultaneously Solving Alignment and Correspondence. In ECCV, 2008

[25] F. Perronnin, J. Sanchez and T. Mensink. Improving the fisher kernel for large-scale image classification. In ECCV, 2010.

[26] L. Quan and Z. Lan. Linear N-point camera pose determination. In PAMI, 1999.

[27] E. Serradell, M. Özuysal, V. Lepetit, P. Fua and F. Moreno-Noguer. Combining Geometric and Appearance Priors for Robust Homography Estimation. In ECCV, 2010.

[28] N. Snavely, S.M. Seitz and R. Szeliski. Photo Tourism: Exploring image collections in 3D. In SIGGRAPH, 2006.

[29] S. Lazebnik, C. Schmid and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In CVPR, 2006.

[30] A. Vedaldi and B. Fulkerson. *VLFeat*: An Open and Portable Library of Computer Vision Algorithms, 2008.

[31] www.vicon.com.

[32] Y. Zheng, Y. Kuang, S. Sugimoto, K. Aström and M. Okutomi. Revisiting the pnp problem: A fast, general and optimal solution. In ICCV, 2013.

[33] Y. Zheng, S. Sugimoto and M. Okutomi. Aspnp: An accurate and scalable solution to the perspective-n-point problem. Trans. on Information and Systems, 2013.

[34] M. Villamizar, A. Sanfeliu and J. Andrade-Cetto. Orientation invariant features for multiclass object recognition. In Iberoamerican Congress on Pattern Recognition, 2006.

[35] M. Villamizar, A. Sanfeliu and J. Andrade-Cetto. Local boosted features for pedestrian detection. In Iberian Conference on Pattern Recognition and Image Analysis, 2009.

[36] M. Villamizar, A. Garrell, A. Sanfeliu and F. Moreno-Noguer. Online human-assisted learning using random ferns. In ICPR, 2012.

[37] M. Villamizar, J. Andrade-Cetto, A. Sanfeliu and F. Moreno-Noguer. Bootstrapping boosted random ferns for discriminative and efficient object classification. In Pattern Recognition, 2012.