Architecture des systèmes d'information

Traitement d'image

Points clés d'une image indépendants de son échelle

Abécassis Zoé - Fonteneau Clémence - Prieur Maxime

Abdelaziz Bensrhair

Septembre - Décembre 2020

I. Introduction

Notre projet vise à étudier une méthode permettant de récupérer des points clés d'une image afin de comparer sa similitude à d'autres images sans tenir compte de sa taille, de son inclinaison, du changement de point de vue, du bruit ou encore du changement de luminosité. Proposée par David G.Lowe, cette méthode peut également être utilisée pour la reconnaissance d'objet par le biais de la méthode du plus proche voisin, d'une transformation de Hough pour identifier les groupes auxquels appartient un objet puis d'une vérification selon la méthode des moindres carrés. Chaque point clé d'une image est distinctif des autres. Ici, l'indépendance d'échelle fait référence au zoom de l'image ou bien à la résolution du capteur. On réduit le coût d'extraction de la caractéristique par une approche de **filtrage en cascade** où les opérations coûteuses sont appliqués à des zones ayant réussi un test préalable. On utilise le principe **SIFT** (scale invariant feature transform) transformant les données de l'image d'entrée en coordonnées indépendantes de l'échelle et relatives aux caractéristiques locales. SIFT se résume en 4 étapes :

1. **Détection d'extremums sur l'espace d'échelle** : Au moyen de la différence de deux fonctions Gaussiennes, on identifie les points constants selon l'orientation et la taille de l'image.

2. **Emplacement des points clés** : Pour tous les points obtenus précédemment, on associe un modèle contenant l'emplacement et l'échelle des points les plus stables.

3. **Affectation de l'orientation** : On attribue une ou plusieurs orientation(s) aux points clés selon la directions locale des gradients de l'image. Des opérations seront appliquées en considérant les affectations ce qui donnera "l'invariance" sur les transformations.

4. **Description des points clés** : On mesure les gradients locaux de l'image a une échelle choisie tout autour des points clés.

L'approche SIFT génère un grand nombre de points clés couvrant l'ensemble de l'image et de ses échelles. Ceci est important pour la reconnaissance d'objet puisqu'il faut au minimum 3 points clés afin de reconnaître un objet en arrière plan d'une image riche en éléments. Les points clés issus de la méthode SIFT sont extraits d'images de référence. On sélectionne une image on obtenant ses caractéristiques et en regardant sa distance euclidienne entre les vecteurs de points clés par rapport aux caractéristiques des images de la base de données déjà obtenue. Sur une image riche, beaucoup d'éléments d'arrière plan auront une fausse correspondance avec les images de la BD. Cela crée de fausses correspondances. On peut effectuer un filtrage en identifiant un sous ensemble de point clé au moyen d'une table de hachage correspondant à la transformée de Hough. Chacun des sous ensemble/cluster de 3 ou plus points en accord avec l'objet est vérifié plus en détail. On fait d'abord une estimation des moindres carrés pour une estimation affine de la position de l'objet. On termine par le calcul de la probabilité de présence d'un objet selon un ensemble de points clés. Les points passant ces tests sont identifiés comme correctes.

II. Origines

Cette méthode a été présentée par David G. Lowe pour la toute première fois en 1999 lors de la conférence international de la vision par ordinateur (ICCV) ayant pour but l'extraction de caractéristiques et de détection d'objet au sein d'images en niveaux de gris. Le nom de la méthode provient de son principe de transformation des données de l'image en coordonnées invariantes selon l'échelle et lié à des caractéristiques locales. En 2004 Lowe fournit des précisions supplémentaires sur sa méthode dans un journal sur la vision par ordinateur.

La méthode de Lowe se base sur des travaux antérieurs, notamments sur les correspondances entre images stéréoscopiques de Hans Moravec puis amélioré par Harris et Stephens. On retrouve des propositions variés sur les caractéristiques à considérer. Parmis celles-ci, des zones ou des segments de lignes. Un peu plus tard, Harris propose une approche en utilisant un descripteur de coins permettant d'améliorer le détecteur de Moravec. Quelques années seulement avant la présentation de Lowe, Cordelia Schmid et Roger Mohr démontrent l'importance des caractéristiques locales invariantes liées dans la détection et la recherche de correspondances. En faisant ça les deux chercheurs développent un détecteur faisant fis de la rotation de l'image mais toujours sensible à la variation d'échelle et d'angle d'observation. Ce problème est donc résolu par Lowe et son descripteur SIFT.

III. Détection d'extremum sur l'espace d'échelle

La première étape de la détections des points d'intérêts est d'identifier l'emplacement et l'échelle des points pouvant être retrouvés selon différentes perspectives sur un même objet. On peut pour cela, chercher les points stables peu importe les échelles au moyen d'une fonction continue d'échelle sur un espace discret , l'espace-échelle.

On utilise pour cela la fonction Gaussienne. Ainsi, en notant (x,y) les coordonnées sur l'image et σ, un facteur d'échelle caractéristique, l'espace-échelle d'une image est définit comme la fonction, $L(x,y,\sigma)$**,** appelé le gradient de facteur d'échelle σ**.** L provient de la convolution d'un filtre Gaussien de paramètre σ.

$$G(x,y,\sigma) = \frac{1}{2\sigma^2\pi} e^{-(x^2+y^2)/2\sigma^2} \quad \textbf{(1)}$$

Avec une image d'entrée $I(x,y)$ on obtient :

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \quad \textbf{(2)}$$

Où * est la convolution en x et y. Cette convolution lisse l'image tel que les détails de rayon inférieur à σ sont estompés. Ainsi, les objets de mêmes dimensions que σ sont détectés par l'étude d'une image résultant d'une différence de fonctions gaussiennes convolutionnées :

$$D(x,y,\sigma) = L(x,y,k\sigma) - L(x,y,\sigma) \quad \textbf{(3)}$$

$D(x, y, \sigma)$ est la différence entre deux échelles voisines séparée par une constante k. D ne contient plus que les objets observables dans des facteurs d'échelle variant entre σ et kσ. Ainsi, un point d'intérêt (x,y,σ) est définis comme un extremum ou la différence des gaussiennes est atteints selon ses voisins, soit l'ensemble de 26 autres points $\{D(x+\delta_x, y+\delta_y, s\sigma),\ \delta_x \in \{-1,0,1\},\ \delta_y \in \{-1,0,1\},\ s \in \{k^{-1}, 1, k\}\}$.

A. Détection des extremums locaux de $D(x, y, \sigma)$

Tous les points de l'échantillons sont comparés à ses 8 voisins sur l'image courante et ses 9 voisins sur l'image au dessus et en dessous. Le point est sélectionné seulement si il est plus gros ou plus petit que l'ensemble des voisins.

Un des problèmes est de déterminer la fréquence d'échantillonnage dans l'image et le domaine d'échelle requis pour détecter les extremums sans se tromper. Malheureusement il n'y a pas de seuil puisque les extremum peuvent être proches. Ainsi, il faut faire un choix entre efficacité et traitement complet. Les extremum proches ne sont pas stables lors de petites perturbations de l'image. On peut donc déterminer expérimentalement le meilleur choix en étudiant un ensemble de fréquence d'échantillonnage et en utilisant celles ayant les meilleures résultats après une simulation réaliste de la tâche de correspondance.

B. Fréquence d'échantillonnage

Il est conseillé d'adopter un modèle en pyramide pour améliorer le temps de calcul d'images floutées sur un large éventail d'échelles. La base correspond à l'image originale sur un "octave". La méthode de Lowe utilise un sigma de départ de 1.6. On accède à l'octave suivant en doublant le facteur d'échelle, ou de façon équivalente, en divisant la résolution de l'image par deux. Sur une même octave, on calcul un nombre de D constant séparés par un sigma de facteur $\sqrt{2}$. La progression géométrique permets aux valeurs des différences de gaussiennes au sein des échelles soient comparable entre elles en se passant d'un facteur de normalisation dans les calculs.

Il est la plupart du temps préférable d'utiliser un grand nombre d'échantillons d'échelles (mais cela augmente la quantité de calculs demandés). L'espace-échelle de la différence de fonction Gaussienne a un grand nombre d'extremums, il est très coûteux de tous les obtenir. Heureusement, on peut détecter les sous espaces les plus utiles et les plus stables même avec un échantillon d'échelle grossier.

IV. Emplacement précis des points clés

De nombreux points candidats trouvés par l'étape précédente sont instables. Leurs localisation est également plus grande (à faible résolution l'emplacement est approximatif). On effectue donc des traitements supplémentaire afin d'une part de converger la position des points dans l'optique d'améliorer la précision en x,y et σ. Dans un second temps cela sert également à rejeter les points ayant un faible contraste ou faiblement localisés aux abords d'un contour. L'implémentation initiale de cette approche localise simplement les attributs selon l'échelle et l'emplacement du point central de l'échantillon. Brown a cependant développé une méthode pour associer une fonction quadratique 3D au points locaux de l'échantillon afin de déterminer l'emplacement interpolé du maximum. L'interpolation utilise la formule de Taylor en utilisant les 3 premiers termes de la fonction d'espace échelle. On fait varier $D(x,y,\sigma)$ de manière à placer l'origine au centre du point échantillonné.

$$D(x) = D + \frac{dD^T}{dx} + \frac{1}{2}x^T \frac{d^2D}{dx^2}x \quad \textbf{(5)}$$

D et ses dérivées sont évalués sur le point et $x = (x,y,\sigma)^T$ est une faible variation de ce point. L'emplacement de l'extremum $\hat{x}$ est obtenu en prenant la dérivée en zéro de la fonction selon x. On obtient alors :

$$\hat{x} = -\frac{d^2D^{-1}}{dx^2}\frac{dD}{dx} \quad \textbf{(6)}$$

La matrice Hessienne et la dérivée de D sont approximées par l'utilisation de différence sur les points échantillonnés voisins. Si l'offset $\hat{x}$ est plus grand que 0.5, toute dimensions confondues, alors cela signifie que l'extremum est en fait un point échantillonné proche dans l'espace des échelles discret. Si c'est le cas, on change le point et on effectue l'interpolation sur un autre point. L'offset $\hat{x}$ final est ajouté à l'emplacement du point pour obtenir l'estimation interpolée de l'extremum, ceci permet d'améliorer la précision. $D(\hat{x})$, la valeur de la fonction à l'extremum est utilisée pour rejeter selon un seuil déterminés, les extremums instables au contraste faible. Avec l'équation **(1)** et **(2)** , on obtient :

$$D(\hat{x}) = D + \frac{1}{2}\frac{dD^T}{dx}\hat{x} \quad \textbf{(7)}$$

A. Elimination des réponses de bord

Pour une meilleure stabilité, ce n'est pas suffisant de rejeter les points avec un faible contraste. La différence de Gaussiennes aura une réponse importante sur les contours/arêtes, donnant lieu à des extremums locaux instables et très sensibles au bruit. Un point d'intérêt à éliminer selon les deux directions principales à sa position, se manifeste par une courbure élevée le long du contour ou ce dernier est positionné en comparaison à la courbure dans la direction perpendiculaire. La principale courbure peut être calculée depuis une matrice Hessienne 2x2, H, à l'endroit et à l'échelle où se situe le point.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

La courbure principale est représentée par les valeurs propres de la matrice H. Les dérivées sont estimées en prenant les différences des points échantillonnés voisins. Les valeurs propres de H sont proportionnelles à la courbure principale de D. Tiré de l'approche utilisée par Harris et Stephens, on peut éviter le calcul explicite des valeurs propres en utilisant seulement leur ratio. Soient α, la valeur propre ayant la plus grande valeur et β la plus petite. On peut calculer la somme des valeurs propres avec la trace de H et les produits des valeurs propre par le déterminant.

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta \quad \textbf{(8)}$$

$$Det(H) = D_{xx} * D_{yy} = \alpha * \beta \quad \textbf{(9)}$$

Si le déterminant est négatif, la courbure a different signe et le point est donc abandonné. Avec r, le ratio entre la plus grande valeur propre et la plus petite on a :

$$\frac{Tr(H)^2}{DET(H)} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r+1)^2}{r} \quad \textbf{(10)}$$

L'équation est minimale lorsque les deux valeurs propres sont égales et cela augmente avec r. C'est pourquoi, pour vérifier que le ratio de la courbure principale est sous le seuil, r_{th}, on vérifie que :

$$R = \frac{tr(H)^2}{detH)} = \frac{(r+1)^2}{r} < \frac{(r_{th}+1)^2}{r_{th}} \quad \textbf{(11)}$$

Dans la méthode de Lowe, $r_{th} = 10$. Ceci est efficace pour calculer avec moins de 20 opérations flottantes sur chacun des points clés. Si le critère n'est pas vérifié le point est considéré comme le long d'un contour et est rejeté.

V. Assignation de l'orientation

En attribuant une orientation cohérente à chaque point clé en fonction des propriétés locales de l'image, le descripteur de point clé peut être représenté en fonction de son orientation et ainsi être insensible aux rotations d'image. L'inconvénient de cette approche est qu'elle limite les descripteurs qui peuvent être utilisés et écarte des informations d'image en n'exigeant pas que toutes les mesures soient fondées sur une rotation cohérente.

L'échelle du point clé est utilisée pour sélectionner l'image lissée par méthode Gaussienne, L, avec l'échelle la plus proche, afin que tous les calculs soient effectués sans variance par rapport à l'échelle. Pour chaque échantillon d'image L(x,y), à cette échelle, la **magnitude du gradient**, m(x, y), et l'**orientation**, θ(x, y), sont pré-calculées en utilisant les différences de pixels :

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}$$

$$\theta(x,y) = tan^{-1}\left(\frac{(L(x,y+1) - L(x,y-1))^2}{(Lx+1,y) - L(x-1,y))^2}\right)$$

Un histogramme d'orientation est formé à partir des orientations de gradient des points d'échantillonnage dans une région autour du point clé. L'histogramme d'orientation comporte 36 cases couvrant la gamme d'orientations de 360 degrés. Chaque échantillon ajouté à l'histogramme est pondéré par sa magnitude de gradient et par une fenêtre circulaire gaussienne pondérée avec un σ qui est 1,5 fois celui de l'échelle du point clé.

Les pics de l'histogramme d'orientation correspondent aux directions dominantes des gradients locaux. Le pic le plus élevé de l'histogramme est détecté, puis tout autre pic local se situant à l'intérieur de la plage 80% du pic le plus haut est également utilisé pour créer un point clé avec cette orientation. Par conséquent, pour de multiples pics d'ampleur similaire, de multiples points clés seront créés au même emplacement et la même échelle, mais dans des orientations différentes. Seulement 15 % environ seront assignés de multiples orientations, mais celles-ci contribuent de manière significative à la stabilité de l'appariement. Enfin, une parabole est ajustée aux 3 valeurs d'histogramme les plus proches de chaque pic pour interpoler la position du pic dans un souci de précision.

VI. Description locale de l’image

Les opérations précédentes ont assigné un emplacement, une échelle et une orientation d'image à chaque point clé. L'étape suivante consiste à calculer un descripteur pour la région locale de l'image qui est très distinctif tout en étant aussi invariant que possible aux variations restantes, telles que le changement d'éclairage ou de point de vue 3D.

A. Représentation des descripteurs

Dans un premier temps dans le calcul de descripteurs de points clé, les amplitudes et orientations des gradients sont prélevées autour de la localisation du point clé, en utilisant l’échelle du point clé pour déterminer le niveau de flou gaussien nécessaire pour l’image. Afin d’obtenir des points invariants, les coordonnées du descripteur et l’orientation des gradients sont pivotés de manière relative à l’orientation du point clé. Dans un souci d’efficacité, les gradients sont précalculés pour tous les niveaux de la pyramide comme décrit dans la section 5.

Une fonction de pondération gaussienne avec σ égale à la moitié de la largeur de la fenêtre de descripteur est utilisée pour attribuer un poids à la magnitude de chaque point d'échantillonnage. Le but de cette fenêtre gaussienne est d'éviter les changements soudains dans le descripteur avec de petits changements dans la position de la fenêtre, et de donner moins d'importance aux gradients qui sont loin du centre du descripteur, car ils sont les plus affectés par les erreurs de mauvais alignement.

Le descripteur du point clé permet un déplacement significatif des positions de gradient en créant des histogrammes d'orientation sur 4x4 régions d'échantillon. La figure montre alors huit directions pour chaque histogramme d'orientation, la longueur de chaque flèche correspondant à l'importance de cette entrée d'histogramme. Un échantillon de gradient à gauche peut décaler jusqu'à 4 positions d'échantillon tout en contribuant au même histogramme à droite, ce qui permet des décalages de position locaux plus importants.

Il est important d'éviter tout effet de frontière dans lequel le descripteur change abruptement alors qu’un échantillon passe en douceur d'un histogramme à l'autre ou d'une orientation à l'autre. Par conséquent, une interpolation trilinéaire est utilisée pour distribuer la valeur de chaque échantillon de gradient dans des boîtes d'histogrammes adjacentes. En d'autres termes, chaque entrée dans une cellule est multipliée par un poids de 1 - d pour chaque dimension, où d est la distance de l'échantillon à la valeur centrale de la cellule, mesurée en unités d'espacement des cellules de l'histogramme.

Le descripteur est formé d'un vecteur contenant les valeurs de toutes les entrées de l'histogramme d'orientation. Nos expériences montrent que les meilleurs résultats sont obtenus avec un tableau 4x4 d'histogrammes avec 8 bacs d'orientation dans chaque tableau. Par conséquent, les expériences de cet article utilisent un vecteur caractéristique de 4x4x8 = 128 éléments pour chaque point clé.

Enfin, le vecteur caractéristique est modifié pour réduire les effets de changements d'éclairage. Tout d'abord, le vecteur est normalisé à la longueur unitaire. Cela permet d'annuler les changements de contraste. Un changement de luminosité dans lequel une constante est ajoutée à chaque pixel de l'image n'affectera pas les valeurs de gradient, car celles-ci sont calculées à partir des différences de pixels. Par conséquent, le descripteur est invariant aux changements affines de luminosité. Cependant, des changements d'éclairage non linéaires peuvent également se produire en raison de la saturation de la caméra ou de changements d'éclairage, affectant les surfaces 3D avec des orientations et des quantités différentes. Ces effets peuvent entraîner un changement important des amplitudes relatives de certains gradients, mais ils sont moins susceptibles d'affecter l'orientation des gradients. Par conséquent, nous réduisons l'influence des grandes amplitudes de gradient en limitant les valeurs du vecteur de caractéristique unitaire à un seuil 0,2, puis en les normalisant en longueur unitaire. Cela signifie que l'appariement des grandeurs pour les grands gradients n'est plus aussi important, et que la distribution des orientations est plus importante. La valeur de 0,2 a été déterminée expérimentalement en utilisant des images contenant des éclairages différents pour les mêmes objets 3D.

B. Test des descripteurs

Deux paramètres peuvent être utilisés pour varier la complexité du descripteur : le nombre d'orientations dans les histogrammes, r, et la largeur, n, du tableau n×n des histogrammes d'orientation. La taille du vecteur descripteur résultant est rn^2. Au fur et à mesure que la complexité du descripteur augmentera, il sera en mesure de mieux faire de distinctions dans une grande base de données, mais il sera aussi plus sensible aux distorsions et occlusions de forme.

Pour une transformation du point de vue dans laquelle une surface plane est inclinée de 50 degrés par rapport au spectateur et un bruit d'image de 4% est ajouté, les résultats montrent que le pourcentage de points-clés trouvant une correspondance correcte avec le voisin le plus proche parmi une base de données de 40 000 points-clés pour un seul histogramme d'orientation (n = 1) est très peu discriminant, mais ces résultats s'améliorent lorsqu'on s'approche d'un tableau 4x4 d'histogrammes à 8 orientations. Par la suite, l'ajout de plus d'orientations ou d'un descripteur plus grand peut en réalité nuire à la correspondance en rendant le descripteur plus sensible à la distorsion. Ces résultats étaient globalement similaires pour d'autres degrés de changement de point de vue et de bruit, bien que dans certains cas plus simples, la discrimination ait continué de s'améliorer (à partir de niveaux déjà élevés) avec des tailles de descripteurs de 5x5 et plus. Bien qu'une dimensionnalité du descripteur de 4x4 puisse sembler élevée, nous avons constaté qu'il donne toujours de meilleurs résultats que les descripteurs de dimensions inférieures pour une gamme de tâches d'appariement et que les coûts informatiques liés au rapprochement demeurent faibles lorsque les méthodes approximatives des plus proches voisins décrites ci-dessous sont appliquées.

C. Sensibilité aux changements affines

La sensibilité du descripteur au changement affine est examinée à la figure 9. Le graphique montre la fiabilité de la sélection de l'emplacement et de l'échelle des points clés, de l'affectation de l'orientation et de l'appariement du voisin le plus proche à une base de données en fonction de la rotation en profondeur d'un plan par rapport à un observateur. On peut voir que chaque étape du calcul a réduit la répétabilité en augmentant la distorsion affine, mais que la précision finale de l'appariement reste supérieure à 50 % jusqu'à un changement de point de vue de 50 degrés.

Pour obtenir un appariement fiable sur un angle d'observation plus large, l'un des détecteurs à constante affine pourrait être utilisé pour sélectionner et rééchantillonner des régions d'image, comme nous l'avons vu à la section 2. Comme nous l'avons mentionné, aucune de ces approches n'est vraiment invariante sur le plan affine, car elles partent toutes d'emplacements initiaux déterminés de façon non invariante sur le plan affine. Dans ce qui semble être la méthode la plus invariante sur le plan affine, Mikolajczyk (2002) a proposé et réalisé des expériences détaillées avec le détecteur Harris-affine. Il a constaté que la répétabilité de son point clé est inférieure à celle donnée ici jusqu'à un angle d'observation d'environ 50 degrés, mais qu'il conserve ensuite une répétabilité de près de 40 % jusqu'à un angle de 70 degrés, ce qui est plus efficace dans les changements affines extrêmes. Les inconvénients sont des coûts de calcul beaucoup plus élevés, une réduction du nombre de points clés et une stabilité plus faible pour les petites modifications affines du fait d'erreurs dans l'attribution d'une image affine cohérente dans un environnement bruité. Dans la pratique, la plage de rotation autorisée pour les objets 3D est considérablement inférieure à celle des surfaces planes, de sorte que l'invariance affine ne constitue généralement pas le facteur limitant la capacité à s'adapter au changement de point de vue transversal. Si une large gamme d'invariance affine est souhaitée, par exemple pour une surface connue pour être plane, alors une solution simple est d'adopter l'approche de Pritchard et Heidrich (2003) dans laquelle des caractéristiques SIFT supplémentaires sont générées à partir de 4 versions de l'image d'apprentissage avec une transformation affine correspondant à des variations de 60 degrés des points de vue. Cela permet l'utilisation de fonctions SIFT standard sans coût supplémentaire lors du traitement de l'image à reconnaître, mais entraîne une augmentation de la taille de la base de données des caractéristiques par un facteur 3.

D. Couplage dans des bases de données de grande taille

Pour mesurer le degré de distinctivité des caractéristiques, il reste une question importante à régler, à savoir comment la fiabilité de l'appariement varie en fonction du nombre de caractéristiques de la base de données à apparier. Par exemple, pour une base de données de 112 images avec une rotation de la profondeur du point de vue de 30 degrés et un bruit d'image de 2 %, on peut constater que la fiabilité de l'appariement diminue en fonction du nombre de distracteurs, mais tout porte à croire que de nombreuses correspondances correctes continueront d'être trouvées pour des bases de données de très grande taille.

VII. Application à la reconnaissance d'objet

L'une des applications principales du SIFT consiste à trouver dans une image donnée (dite *image question* ou *image suspecte*), des objets déjà présents dans une collection d'images de référence pré-établie.

La reconnaissance d'objet s'effectue tout d'abord en faisant correspondre chaque point clé indépendamment à la base de donnée de points clés extraite d'images d'entraînement. Un grand nombre de ces correspondances seront incorrectes à cause des caractéristiques ambiguës ou de celles qui proviennent de l'encombrement présent à l'arrière plan. Par conséquent, on identifie d'abord des groupes d'au moins 3 caractéristiques qui s'accordent sur un objet et sa pose, car ces groupes ont une plus grande probabilité d'être corrects que des correspondances de caractéristiques individuelles. Ensuite, chaque groupe est vérifié en effectuant un ajustement géométrique détaillé au modèle. Le résultat permet d'accepter ou de rejeter l'interprétation.

A. Association de points clés

On trouve le meilleur candidat pour chaque point clé en identifiant son plus proche voisin dans la base de données des images d'entraînement. Le plus proche voisin est défini comme le point clé ayant une distance euclidienne minimale pour le vecteur descripteur invariant, comme décrit dans la partie VI.

Cependant, plusieurs caractéristiques d'une image n'auront aucune association correcte dans la base de données d'entraînement parce qu'elles sont présentes dans l'arrière plan de l'image ou n'ont pas été détectées dans les images d'entraînement. Par conséquent, il faut donc un moyen de rejeter les caractéristiques qui n'ont aucune bonne association à la base de données.

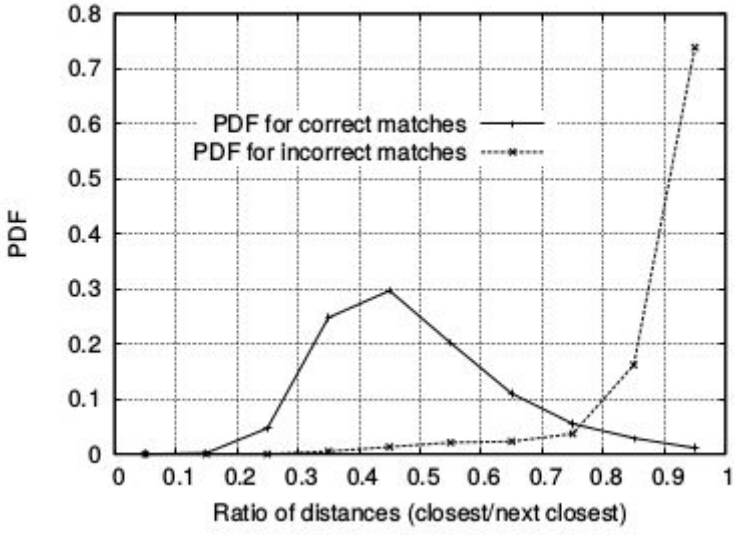

Figure 11

Ce graphe montre la valeur de la fonction de densité de probabilité (probability density function : PDF) en fonction du rapport de la distance entre le voisin le plus proche et le deuxième plus proche de chaque point clé, pour les correspondances correctes et incorrectes. Les correspondances pour lesquelles le plus proche voisin était une correspondance correcte ont une fonction de densité de probabilité centrée à un ratio bien

plus bas que celles pour les correspondances incorrectes. Pour implémenter la reconnaissance d'objets, on rejette toutes les associations pour lesquelles le rapport de distance est supérieur à 0.8, ce qui élimine 90% des fausses associations tout en rejetant moins de 5% des associations correctes.

B. Indexation efficace du plus proche voisin

Pour identifier les plus proches voisins exacts de points dans des espaces à nombre de dimensions élevé comme c'est le cas ici (notre descripteur de points clés possède un vecteur de caractéristiques de dimension 128), le moyen le plus efficace reste la recherche exhaustive.

Lowe utilise un **arbre kd** pour indexer les descripteurs des images de référence. Les arbres *k*-d sont des arbres binaires, dans lesquels chaque nœud contient un point en dimension *k*. . Chaque nœud non terminal divise l'espace en deux demi-espaces. Les points situés dans chacun des deux demi-espaces sont stockés dans les branches gauche et droite du nœud courant. Dans notre cas, on a un arbre 128d où sont répartis les points clés de l'image.

Est utilisé ensuite un algorithme de recherche modifié par rapport à l'approche classique, appelé ***Best bin first (BBF)***. Cet algorithme renvoie le voisin le plus proche avec une probabilité élevée. Il permet une recherche exhaustive rapide pour plusieurs raisons. Tout d'abord, les nœuds de l'arbre kd sont explorés dans l'ordre de leur distance au descripteur de l'image question, grâce à l'utilisation d'une **file de priorité** basée sur un **tas binaire**, ce qui permet de déterminer l'ordre de recherche de manière efficace. Ensuite, le nombre de boîtes (feuilles de l'arbre kd) -ou *bins*- à explorer pour trouver le plus proche voisin d'un descripteur donné est limité à une valeur maximale fixée. Dans notre implémentation, on stoppe la recherche après avoir vérifié les 200 premiers potentiels plus proches voisins. Pour une base de données de 100 000 points clé, cela permet d'accélérer d'environ 2 ordres de grandeur la recherche exacte du voisin le plus proche tout en entraînant une perte de moins de 5% du nombre de correspondances correctes.

Une des raisons pour lesquelles l'algorithme BBF fonctionne si bien pour ce problème est que l'on considère seulement les correspondances pour lesquelles la distance du voisin le plus proche est inférieure de 0.8 fois la distance du deuxième voisin le plus proche (comme expliqué dans la partie précédente). Par conséquent, il n'y a pas besoin de résoudre les cas les plus difficiles dans lesquelles beaucoup de voisins se trouvent à des distances similaires.

C. Groupement avec la transformée de Hough

Afin de maximiser la performance de la reconnaissance d'objet pour les petits ou à occlusion élevée, on souhaite être en mesure d'identifier des objets avec le plus petit nombre de correspondances possibles. Une reconnaissance fiable est possible avec seulement **3 caractéristiques**. Une image typique contient au moins 2000 caractéristiques qui peuvent provenir de différents objets mais aussi du bruit de fond.

Chaque correspondance individuelle entre points-clés obtenue à l'étape précédente constitue une hypothèse quant à la pose (point de vue sous lequel il est photographié) de l'objet sur l'image question par rapport à l'image de référence concernée. On souhaite

grouper les hypothèses cohérentes entre elles de façon à mettre en correspondance les objets des images et non plus seulement des points isolés.

La t**ransformée de Hough** est une technique de reconnaissance de formes qui identifie des groupes de correspondances, que l'on appelle *clusters*, par un système de vote. Quand on constate que les groupes de caractéristiques votent pour la même pose pour un objet, la probabilité que l'interprétation soit correcte est beaucoup plus élevée que pour les caractéristiques seules. Chacun de nos points clés spécifie 4 paramètres : la **position** (2D), l'**échelle** et l'**orientation** ; et chaque point clé associé dans la base de données enregistre les paramètres du point clé concernant l'image d'entraînement dans laquelle il a été trouvé. Par conséquent, il est possible de créer une entrée de transformée de Hough qui prédit la localisation, l'orientation et l'échelle du modèle à partir de l'hypothèse de correspondance.

Cette prédiction a une large marge d'erreur, puisque la similitude (transformation) impliquée par ces 4 paramètres est seulement une approximation de l'espace de pose complet de 6 degrés de liberté pour un objet 3D et ne tient pas compte non plus des déformations non rigides. Par conséquent, on utilise de larges tailles de compartiments de 30 degrés pour l'orientation, un facteur 2 pour l'échelle, et 0.25 fois la dimension maximale de l'image d'entraînement projetée (en utilisant l'échelle prévue) pour la localisation. Afin d'éviter le problème des effets de limites dans l'assignation de compartiments, chaque association de point clé vote pour les 2 plus proches compartiments dans chaque dimension, ce qui donne un total de 16 entrées pour chaque hypothèse et élargit encore la plage de pose.

Dans la plupart des implémentations de la transformée de Hough, un tableau multidimensionnel est utilisé pour représenter les compartiments. Cependant, un grand nombre de compartiments potentiels vont rester vides, et il est difficile de calculer l'étendue des valeurs des compartiments potentiels à cause de leur dépendance mutuelle (par exemple, la dépendance de la discrétisation de la localisation sur l'échelle sélectionnée). Ces problèmes peuvent être évités en utilisant une **fonction de hachage** pseudo-aléatoire des valeurs de compartiment pour insérer les votes dans une table de hachage unidimensionnelle, dans laquelle les collisions sont facilement détectées.

Lorsque plusieurs correspondances votent pour une même boîte, c'est-à-dire pour la même pose d'un objet, la probabilité que la correspondance soit correcte est largement supérieure à ce que pourrait donner une correspondance isolée. Ainsi, les boîtes contenant au moins trois entrées sont considérées comme des clusters fiables, et retenus pour la suite de l'analyse.

D. Solution pour les paramètres affines

La transformée de Hough est utilisée pour identifier tous les groupes avec au moins 3 entrées dans un compartiment. Chacun de ces groupes est ensuite soumis à une procédure de vérification par l'application de la méthode des **moindres carrés** aux paramètres de la **transformation affine** reliant le modèle (image de référence) à l'image question.

Une transformation affine tient correctement compte de la rotation 3D d'une surface plane sous projection orthographique, mais l'approximation peut être mauvaise pour la rotation 3D d'objets non plans. Si on imagine placer une sphère autour d'un objet, alors la rotation de la sphère de 30 degrés ne déplacera aucun point dans la sphère de plus de 0,25 fois le diamètre projeté de la sphère. Pour les exemples d'objets 3D typiques, une solution

affine fonctionne bien étant donné qu'on autorise les erreurs résiduelles jusqu'à 0.25 fois le diamètre projeté de l'objet.

La transformation affine d'un point modèle $[x\,y]^t$ en un point image $[u\,v]^t$ peut être écrit de la façon suivante :

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

Où le modèle de translation est $[t_x\,t_y]^t$ et la rotation affine, l'échelle, l'écartement sont représentées par les paramètres m_i.

L'équation ci-dessus peut être réécrite afin de réunir les inconnus dans un vecteur colonne.

$$\begin{bmatrix} x & y & 0 & 0 & 1 & 0 \\ 0 & 0 & x & y & 0 & 1 \\ & & \cdots & & & \\ & & \cdots & & & \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} u \\ v \\ \vdots \end{bmatrix}$$

Cette équation montre une simple association, mais d'autres correspondances peuvent être ajoutées, chaque correspondance contribuant deux lignes supplémentaires à la première et à la dernière matrice. Au moins 3 correspondances sont nécessaires pour fournir une solution.

On peut écrire ce système linéaire de la façon suivante :

$$Ax = b$$

La solution des moindres carrés pour les paramètres x peut être déterminée en résolvant les équations normales correspondantes.

$$x = [A^t A]^{-1} A^t b$$

Cette équation minimise la somme des carrés des distances entre les emplacements des modèles projetés et les emplacements des images correspondantes. Cette approche des moindres carrés pourrait aisément être étendu à la résolution pour les poses 3D et les paramètres internes d'objets flexibles et articulés.

Les valeurs aberrantes peuvent désormais être retirées en vérifiant l'accord entre chaque élément de l'image et le modèle. Étant donnée la solution des moindres carrés la plus efficace, nous avons maintenant besoin que chaque correspondance se situe dans la moitié de la marge d'erreur utilisée pour les paramètres des compartiments de transformation Hough. S'il reste moins de 3 points après l'élimination des valeurs aberrantes, l'association est rejetée. Au fur et à mesure que les valeurs aberrantes sont éliminées, la solution des moindres carrés est résolue avec les points restants, et le processus est répété. De plus, une phase d'appariement descendante est utilisée pour ajouter d'autres associations qui correspondent à la position du modèle projeté et qui avaient été précédemment écartées.

La **décision finale** d'accepter ou de rejeter une hypothèse de modèle est fondée sur un modèle probabiliste détaillé présenté dans un article précédent. Cette méthode détermine en premier lieu le nombre attendu de fausses correspondances, connaissant la taille estimée du modèle, le nombre de points-clés dans la région et la précision de la correspondance. Une **analyse par inférence bayésienne** donne ensuite la probabilité que l'objet soit présent sur la base du nombre réel de caractéristiques correspondantes trouvées. Nous acceptons un modèle si la probabilité finale d'une interprétation correcte est supérieure

à 0,98. Pour les objets qui projettent sur de petites régions d'une image, trois caractéristiques peuvent être suffisantes pour une reconnaissance fiable. Pour les objets de grande taille couvrant la majeure partie d'une image fortement texturée, le nombre de fausses correspondances attendu est plus élevé, et jusqu'à 10 correspondances de caractéristiques peuvent être nécessaires.

VIII. Mise en pratique de la méthode sur python

Nous avons dans un premier temps codés toute la partie permettant l'obtention des descripteurs de points clés en donnant une image en entrée.

Ensuite nous ....

IX. Conclusion

En résumé :

Étape	Techniques utilisées	Avantages
Extraction des points-clés (des images de référence et de l'image question)	Pyramide de gradients, différence de gaussiens, assignation d'orientation	Précision, stabilité, invariance aux modifications d'échelle et à la rotation
Calcul des descripteurs (des images de référence et de l'image question)	Échantillonnage et lissage des plans locaux d'orientation de l'image	Stabilité relative aux transformations affines et à la luminosité
Indexation des descripteurs des images de référence	Arbre kd	Efficacité
Recherche des correspondances avec les descripteurs de l'image question	Plus proche voisin approximatif (*Best Bin First*)	Rapidité
Identification de clusters	Transformée de Hough et table de hachage	Modèle de transformation fiable
Vérification du modèle	Moindres carrés linéaires	Élimination des fausses correspondances
Validation de l'hypothèse	Inférence bayésienne	Fiabilité

Aujourd'hui la méthode SIFT bien que breveté est très utilisée dans des applications telles que la détection d'objet, la cartographie, l'assemblage de photo pour par exemple créer les panorama,la recherche d'image par contenu, le suivi de mouvement ou bien la modélisation 3D. Cependant d'autre méthodes venant compléter la reconnaissance, mettant en valeur d'autre caractéristiques d'une image ou même surpassant les performance de SIFT existent. C'est notamment le cas avec la méthode SURF (Speeded Up Robust Features), plus efficace en terme de répétitivité, distinctivité et robustesse.

X. Annexe

Lexique

Références

- https://fr.wikipedia.org/wiki/Scale-invariant_feature_transform?fbclid=IwAR2MxIV9HeyMq01Cnwc4s6hhsMaDwBmQv5T9KE10PzgzPfn44Q68CLoicO8#Origines_de_la_m%C3%A9thode , Site internet Wikipedia

- https://www.youtube.com/watch?v=NPcMS49V5hg&fbclid=IwAR0PUT88LD4uWF0nScXh8m-lsgzDHTKAtUIlV4FcrniQYYMixcPw1u6r67I, Vidéo UCF CRCV

- https://www.quora.com/What-are-some-interesting-applications-of-object-detection, Site internet Quora

- https://towardsdatascience.com/sift-scale-invariant-feature-transform-c7233dc60f37, Site internet towardsdatascience

- https://medium.com/@lerner98/implementing-sift-in-python-36c619df7945, Site internet medium.com

- https://lear.inrialpes.fr/~jegou/teaching/IPR_ppv.pdf, *Algorithmes pour la fouille dans les très grandes bases d'images: le problème des plus proches voisins* | Hervé Jégou, INRIA, 2009

- https://medium.com/analytics-vidhya/a-detailed-guide-to-the-powerful-sift-technique-for-image-matching-with-python-code-acb0cb1d305e | A Detailed Guide to the Powerful SIFT Technique for Image Matching (with Python code), Aishwarya Singh