Aide à la navigation dans un ensemble de vidéos par reformulation de trajectoires

T. Malon S. Chambon V. Charvillat A. Crouzil

IRIT, Université de Toulouse, Toulouse, France

thierry.malon@irit.fr

Résumé

Nous présentons une approche de classement d'une collection de vidéos avec recouvrement. A partir d'une trajectoire requête tracée dans l'une des vidéos, nous classons les autres vidéos dans l'ordre de celles qui permettent de voir au mieux cette trajectoire. Notre approche estime une carte de correspondance entre régions de différentes vidéos en se basant sur la corrélation linéaire de leur taux d'occupation au cours du temps. L'idée principale est que deux régions de deux vidéos différentes se correspondent d'autant mieux qu'elles sont systématiquement et simultanément occupées. Nous utilisons ensuite les correspondances entre régions pour estimer la trajectoire reformulée dans chacune des autres vidéos puis les classons en fonction de la visibilité qu'elles en offrent.

Mots Clef

Reformulation de trajectoire, vidéo-surveillance, vues multiples, vues avec recouvrement, correspondance.

Abstract

We present an approach for ranking a collection of videos with overlapping fields of view. The ranking depends on how they allow to visualize as best as possible, i.e. with significant details, a trajectory query drawn in one of the videos. Our approach estimates a correspondence map between regions from different videos using the linear correlation between their occupation rate by objects over time. The main idea is that two areas from two different videos that systematically offer presence of objects simultaneously are very likely to correspond to each other. Then, we use the correspondence between cells to find the reformulated trajectory in the other videos and rank the videos based on the visibility they offer.

Keywords

Trajectory reformulation, video surveillance, multiple views, overlapping fields of view, matching.

1 Introduction

La prolifération des appareils d'enregistrement vidéo pose de nouveaux défis. En effet, de nos jours, il est facile de trouver de nombreuses vidéos d'un même événement. Disposer de plusieurs vues avec recouvrement permet une meilleure compréhension d'une scène plutôt qu'en utilisant une seule vue. Cependant, regarder l'intégralité de chaque vidéo est une tâche longue, pénible et coûteuse. Il paraît donc crucial de permettre à l'utilisateur de naviguer facilement dans une collection de vidéos pour lui permettre un gain de temps et d'efficacité.

De nombreux travaux récents se sont intéressés au problème de la visualisation simultanée de multiples vidéos. Lorsque les paramètres de calibrage des caméras sont connus et que l'on dispose de nombreuses images de la scène, il est possible d'en obtenir une reconstruction 3D statique en détectant des points d'intérêt dans différentes images et en les appariant [1]. Pour prendre en compte la continuité temporelle disponibles dans des vidéos, les auteurs de [9] ont proposé une reconstruction 4D de la scène dans laquelle les parties statiques sont enrichies des parties dynamiques (parties en mouvement dans le temps). Ces reconstructions contiennent des éléments issus d'une multitude de points de vue. les résultats fournies offrent donc une bonne représentation de la scène qui permet une compréhension globale des évènements.

Cependant, il n'est pas toujours possible de procéder à de telles reconstructions, car elles nécessitent le respect de ces deux hypothèses : connaissance des paramètres de calibrage des caméras et points de vue des caméras assez proches. Lorsqu'une reconstruction 3D n'est pas possible, une autre famille de méthodes consiste à naviguer parmi les différentes vidéos en affichant au fil du temps le point de vue de celle qui décrit le mieux la scène. Dans [4], des scores sont attribués à chaque vue à partir de l'activité des objets, leurs tailles, leurs positions et leurs nombres, ainsi qu'en fonction d'événements s'y produisant. Une vue est alors d'autant plus intéressante que son score est élevé. La vue affichée est régulièrement et automatiquement mise à jour pour montrer la meilleure vue. Dans cet article, nous supposons que nous ne disposons pas du calibrage et nous

n'utiliserons donc pas de méthode de reconstruction. Ce scénario nous semble plus proche d'un cas réel.

Dans le domaine de la vidéo-surveillance, les enquêteurs ont généralement des dizaines de caméras à traiter dont certaines pouvant présenter du recouvrement dans leurs champs de vue. Les tâches qu'ils sont régulièrement amenés à effectuer comprennent notamment :

1. Trouver un véhicule qui a suivi une trajectoire donnée ;
2. Trouver une autre vidéo dans laquelle apparaît une personne qui a été aperçue dans une première vidéo avec plus de détails ;
3. Trouver toutes les personnes qui sont entrées ou sorties d'un bâtiment ;
4. Trouver tous les instants où il y a eu de l'activité dans une région d'intérêt.

Pour naviguer entre les vidéos, les enquêteurs préfèrent généralement pouvoir formuler manuellement des requêtes sur des éléments d'intérêt et choisir parmi une liste classée de vidéos qui correspondent à leur requête, plutôt que d'être redirigés automatiquement d'une vidéo à l'autre.

Prenons l'exemple d'une attaque à la bombe dans un bâtiment. Après l'évènement, les enquêteurs reçoivent de grandes quantités de vidéos issues de caméras de surveillance ou filmées par des témoins. Partant d'une vidéo dans laquelle le bâtiment est visible, ils peuvent vouloir trouver toutes les personnes qui sont entrées dans le bâtiment ou en sont sorties aux alentours du moment de l'explosion. Parmi le grand nombre de vidéos dont ils disposent, celle dont ils partent n'est pas nécessairement celle qui capture le mieux l'entrée du bâtiment, que ce soit en raison d'éventuelles occultations, des conditions d'éclairage ou de l'angle de la caméra.

Il leur serait donc pratique de pouvoir tracer une trajectoire requête dans cette vue, par exemple une flèche pointant vers l'entrée, et d'obtenir une liste, triée par ordre de pertinence, des vues dans lesquelles la trajectoire correspondante à la requête est plus étirée et offre donc une meilleure visibilité. Dans cet esprit, dans [2], l'utilisateur peut sélectionner une région d'intérêt dans une vidéo parmi une collection de vidéos présentant du recouvrement, et être redirigé vers celle qui offre la meilleure entropie vis-à-vis des objets contenus dans la région désignée.

Pour des cas où les vues sont disjointes, les auteurs de [3] proposent de conjointement calculer la topologie du réseau de caméras et de ré-identifier des piétons parmi les différentes caméras. Ils renforcent itérativement la topologie en s'appuyant sur la ré-identification et inversement. Dans le même contexte, les auteurs de [6, 7] proposent d'étudier la structure d'un réseau de caméras en estimant les liens entre régions de différentes caméras et le délai temporel séparant des régions liées. Pour cela, ils découpent chaque vue en cellules et mesurent l'activité de chaque cellule dans le temps à partir de vidéos dans lesquelles sont détectés des personnes. Les cellules d'une même caméra présentant une activité similaire sont groupées en régions, puis un délai est estimé entre certaines régions de différentes caméras : par exemple, la région correspondant au haut d'un escalier dans une caméra peut être lié à la région correspondant au bas de ce même escalier dans une autre caméra avec un délai de quelques secondes. Leur analyse canonique des corrélations appliquée aux fonctions d'activité donne des résultats suffisamment fiables et précis pour estimer la topologie spatiale et temporelle d'un réseau de caméras. Le contexte de ce travail est un peu différent, vu que les vidéos étudiées se recouvrent mais les résultats encourageants qu'ils ont obtenus nous ont amener à proposer d'estimer des correspondances entre les vidéos que nous étudions en utilisant des fonctions d'activité et une analyse canonique des corrélations entre ces fonctions.

La plupart des travaux qui s'appuient sur des caméras avec recouvrement supposent que les paramètres de calibrage des caméras sont connus ou peuvent être estimés. Dans un cas général, c'est-à-dire avec des vidéos de caméras de surveillance ou filmées avec un smartphone, il est difficile d'avoir une estimation fiable de ces paramètres. Dans ce travail, nous ne chercherons pas à les calculer. Dans le contexte des vidéos avec recouvrement, nous pouvons également citer le travail de [5]. Les lignes de vue de caméras en recouvrement sont obtenues en détectant le point de contact au sol d'un objet détecté dans une caméra à l'instant où ce même objet apparaît ou disparaît dans une autre vidéo. Dans ce papier, nous exploitons également des détections faites sur les vidéos.

La contribution principale de cet article est de proposer une nouvelle approche pour reformuler une trajectoire tracée dans une vue en ses trajectoires correspondantes dans d'autres vues en recouvrement pour proposer un classement de ces vues en fonction de la visibilité que chacune offre de la trajectoire reformulée. Nous commençons par introduire les fonctions d'activité pour estimer les cartes de correspondance entre paires de vidéos. Puis, lorsqu'un utilisateur trace une requête trajectographique, nous proposons un score de reformulation pour trouver la trajectoire correspondante dans chaque autre vidéo en recouvrement. Enfin, nous définissons un score de visibilité pour classer les vidéos en considérant la longeur de la trajectoire projetée dans cette caméra. En effet, nous supposons que plus une trajectoire apparaît comme étendue dans une vue, plus cette trajectoire a une chance d'être détaillée, c'est-à-dire de fournir un point de vue plus informatif où on peut espérer obtenir de nouveaux détails sur cette trajectoire et sur l'action relative à cette trajectoire.

Nous exposons notre approche dans la section 2. Après une brève vue d'ensemble, nous expliquons dans la section 2.1 comment nous formulons des cartes de correspondance pour chaque cellule de chaque caméra. La section 2.2 détaille notre méthode de reformulation de trajectoire, puis notre politique de classement des vidéos est exposée dans la section 2.3. Nous présentons nos expérimentation dans la section 3 avant de conclure dans la section 4.

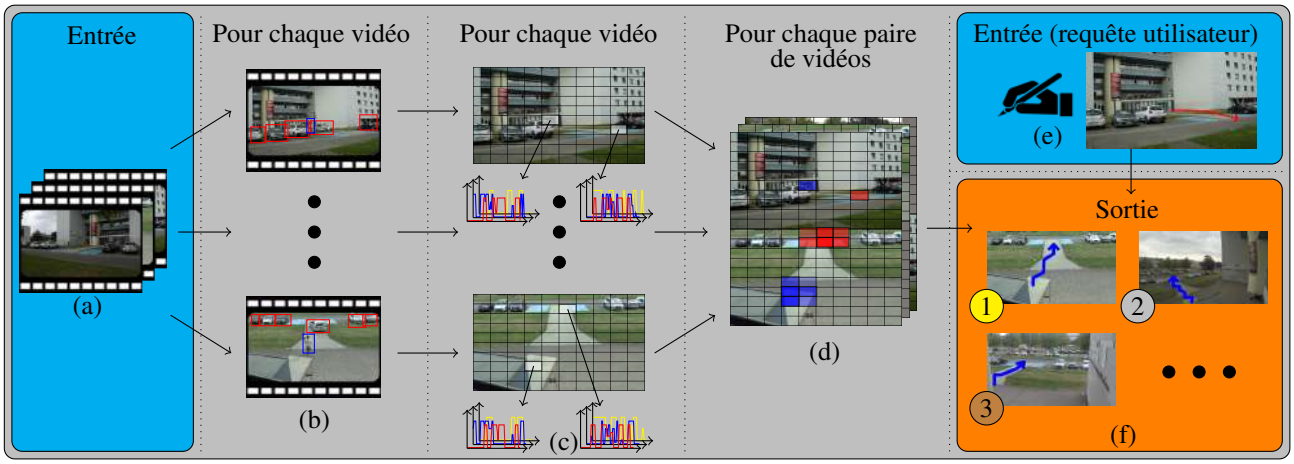

FIGURE 1 – Vue d'ensemble de l'approche : (a) En entrée, une collection de vidéos, (b) détection des objets et de leurs catégories, (c) calcul des fonctions d'activité, (d) calcul des cartes de correspondance, (e) requête trajectographique d'un utilisateur, (f) classement des vidéos en fonction de leur score de visibilité.

2 Méthode proposée

Nous supposons que chaque vidéo a été filmée depuis un point de vue statique, et que l'ensemble des vidéos est temporellement synchronisé : les vidéos commencent donc au même instant et ont la même durée. Nous ne disposons d'aucun paramètre de calibrage des caméras et ne chercheront pas à les estimer.

Les étapes successives de notre approche sont décrites par la figure 1. À partir d'une collection de vidéos, en (a), nous détectons les objets d'intérêt et leur assignons une catégorie, voir (b) où des boîtes englobantes rouges correspondent aux véhicules et une boîte englobante bleue correspond à un piéton. Cette étape peut être effectuée par exemple en utilisant ROLO [10]. Ensuite, en (c), nous décomposons chaque vue en cellules et associons à chaque cellule une fonction d'activité par catégorie d'objet. La fonction d'activité est définie comme étant le taux d'occupation de la cellule par un objet, d'une catégorie donnée, au cours du temps. Puis, pour chaque paire de vidéos, en (d), nous assignons, à chaque cellule de la première vidéo de la paire, une carte de correspondance indiquant la correspondance avec les cellules de la deuxième vidéo. Cette mise en correspondance est effectuée en fonction de la corrélation entre les fonctions d'activité (deux cellules sont mises en valeur dans la vue du haut et la carte de correspondance apparaît sur la vue du dessous).

Un utilisateur peut tracer manuellement une trajectoire requête dans l'une des vues, en (e). À partir de la carte de correspondance dans les autres vues de chaque cellule traversée par la requête, nous obtenons une trajectoire reformulée en choisissant la séquence des cellules correspondantes qui offre le meilleur score de reformulation. Ce score maximise le rapport entre la correspondance des cellules de la trajectoire reformulée avec celles de la trajectoire d'origine, d'une part, et la distance entre les cellules consécutives de la trajectoire reformulée, d'autre part. Un classement des vidéos en fonction de ce score de reformulation et de la visibilité qu'elles offrent de la trajectoire reformulée est finalement obtenu en (f).

2.1 Cartes de correspondance

Dans cette section, nous définissons la notion de fonction d'activité et explicitons notre façon de calculer la carte de correspondance entre chaque région d'une vue et les régions qui lui correspondent dans une autre vue.

Nous découpons chaque vidéo V en N cellules $c_{\mathbf{i}}^V$ selon une grille régulière. L'impact du choix de la valeur de N est étudié dans la section 3. Le traitement des images d'une vidéo sera ainsi fait par cellule et non par pixel, afin de réduire les temps de calculs, d'une part, et d'avoir une information plus riche que celle contenue dans un seul pixel, d'autre part.

Pour une vidéo V, notons $d_\omega^V(t)$ l'ensemble des objets de la catégorie ω détectés au temps t. La fonction d'activité de la catégorie ω dans la vue V, notée $a_{\mathbf{i}}^{V,\omega}$, est le taux de recouvrement de la cellule $c_{\mathbf{i}}^V$ par des détections d_ω^V des objets de la même catégorie ω au cours du temps (voir figure 3) :

$$a_{\mathbf{i}}^{V,\omega}(t) = \frac{|c_{\mathbf{i}}^V \cap d_\omega^V(t)|}{|c_{\mathbf{i}}^V|} \tag{1}$$

où $|\cdot|$ désigne le nombre de pixels. On notera qu'une cellule est associée à une fonction d'activité par catégorie d'objets (voir figure 1.(c)).

Soient V_1 et V_2 deux vidéos temporellement synchronisées présentant du recouvrement dans leurs champs de vue. Deux cellules de deux vidéos différentes $c_{\mathbf{i}}^{V_1}$ et $c_{\mathbf{i}'}^{V_2}$ ont de fortes chances de se correspondre si elles présentent simultanément et systématiquement de l'activité aux mêmes temps, c'est-à-dire si leurs fonctions d'activité sont fortement corrélées positivement.

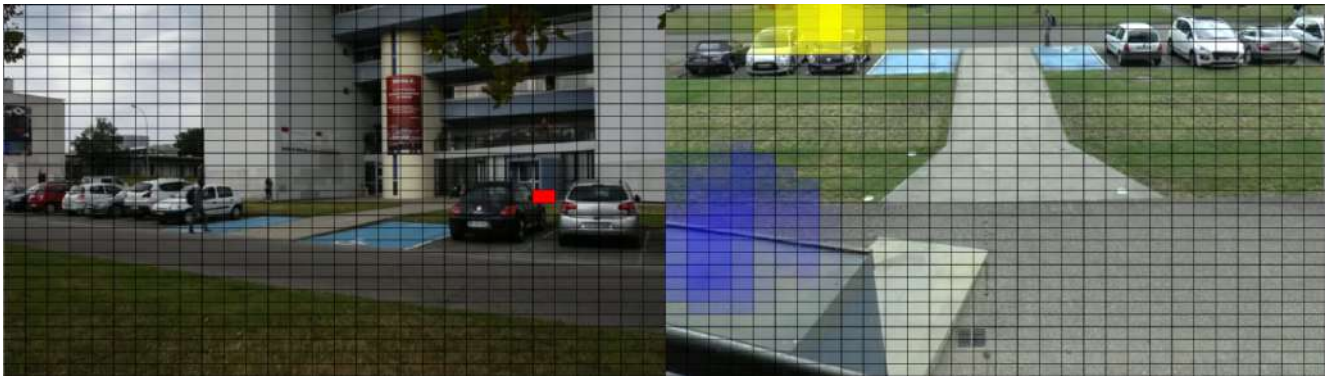

FIGURE 2 – Deux vues avec recouvrement du jeu de données ToCaDa [8]. A gauche : une cellule est colorée en rouge. A droite : les cartes de corrélation correspondant à la cellule dans une autre vue pour les catégories « piéton » (en bleu) et « moto » (en jaune). Comme la cellule en rouge peut être couverte à la fois par les pieds d'une personne marchant à l'arrière plan derrière les voitures garées et par le haut d'une moto passant au premier plan, les cartes de correspondance couvrent deux groupes disjoints de cellules.

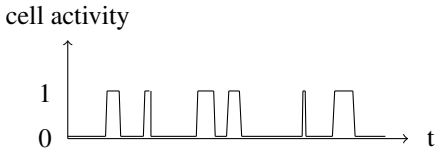

FIGURE 3 – Fonction d'activité d'une cellule pour une catégorie donnée : nous calculons la proportion de la cellule qui est occupée par un object de la catégorie au cours du temps.

Nous définissons ainsi le taux de correspondance entre deux cellules pour la catégorie ω par :

$$\mathcal{C}_\omega(c_{\mathbf{i}}^{V_1}, c_{\mathbf{i}'}^{V_2}) = \max(0, \text{corr}(a_{\mathbf{i}}^{V_1,\omega}, a_{\mathbf{i}'}^{V_2,\omega})) \quad (2)$$

où corr est la corrélation linéaire entre deux variables aléatoires. Pour chaque cellule $c_{\mathbf{i}}^{V_1,\omega}$, nous définissons aussi une carte de correspondance $\mathcal{C}_\omega(c_{\mathbf{i}}^{V_1,\omega}, V_2)$ qui contient les scores de corrélations entre $c_{\mathbf{i}}^{V_1,\omega}$ et l'ensemble des cellules de V_2. La figure 1.(d) illustre les cartes de correspondance : une paire de vues est affichée, et deux cellules sont mises en avant dans la vue du haut en bleu et en rouge, respectivement pour les catégories « piéton » et « véhicule ». La vue du bas montre les cartes de correspondance de ces deux cellules : en bleu, des cellules qui ont été occupées par un objet de la catégorie « piéton » lorsque la cellule en bleu de la vue du haut l'était simultanément, et de même en rouge pour la catégorie « voiture ».
Bien qu'il soit très improbable que deux cellules de deux vidéos différentes, et qui ne se correspondent pas, présentent systématiquement la même activité, cela peut se produire lorsque la durée des vidéos est courte. Si l'on dispose d'heures de vidéos synchronisées, ce qui est généralement le cas avec des caméras de surveillance, les régions qui se correspondent peuvent être d'autant mieux apprises. D'autre part, la distinction des catégories permet d'éviter de faire correspondre une cellule d'une vue qui pourrait être couverte par des objets de différentes tailles à de très nombreuses cellules dans une autre vue (voir figure 2).

2.2 Reformulation de trajectoire

Dans cette section, nous présentons notre méthode de reformulation de trajectoire qui s'appuie sur les cartes de correspondance. Une trajectoire est définie par une succession de segments connectés. Pour une trajectoire requête donnée, nous noterons $\mathcal{S}$ la séquence de M cellules non consécutivement identiques $(c_{\mathbf{i}_1}, ..., c_{\mathbf{i}_M})$ qui sont traversées par les segments formant la trajectoire requête, voir figure 5.
Pour trouver la trajectoire qui correspond à une trajectoire requête, dans une autre vue, nous voulons trouver les indices des cellules correspondantes $(\mathbf{i'}_1, ..., \mathbf{i'}_M)$. Nous obtenons la trajectoire reformulée dans V_2 en joignant les centres des cellules de la séquence de cellules $(\mathbf{i'}_1, ..., \mathbf{i'}_M)$. Pour trouver les indices, nous maximisons le rapport entre les corrélations des cellules traversées par la requête et les cellules traversées par la trajectoire reformulée, d'une part, tout en assurant la continuité de la trajectoire reformulée, d'autre part. Dans ce but, nous pénalisons les successions de cellules dans la trajectoire reformulée qui ne sont pas adjacentes. Le score de reformulation d'une séquence de cellules $\mathcal{S}$ entre deux vues V_1 et V_2 est donc donné par l'expression :

$$\underset{(\mathbf{i'}_1,...,\mathbf{i'}_M)}{\text{argmax}} \frac{\frac{1}{M}\sum_{k=1}^{M} \mathcal{C}_\omega(c_{\mathbf{i}_k}^{V_1}, c_{\mathbf{i'}_k}^{V_2})}{1 + \sum_{k=1}^{M-1} \max(0, ||\mathbf{i'}_k - \mathbf{i'}_{k+1}|| - 1)} \quad (3)$$

Comme attendu, le numérateur favorise des cellules de V_2 qui ont une bonne correspondance avec les cellules traversées dans V_1 tandis que le dénominateur pénalise des cellules consécutives qui ne sont pas adjacentes dans la trajectoire reformulée.

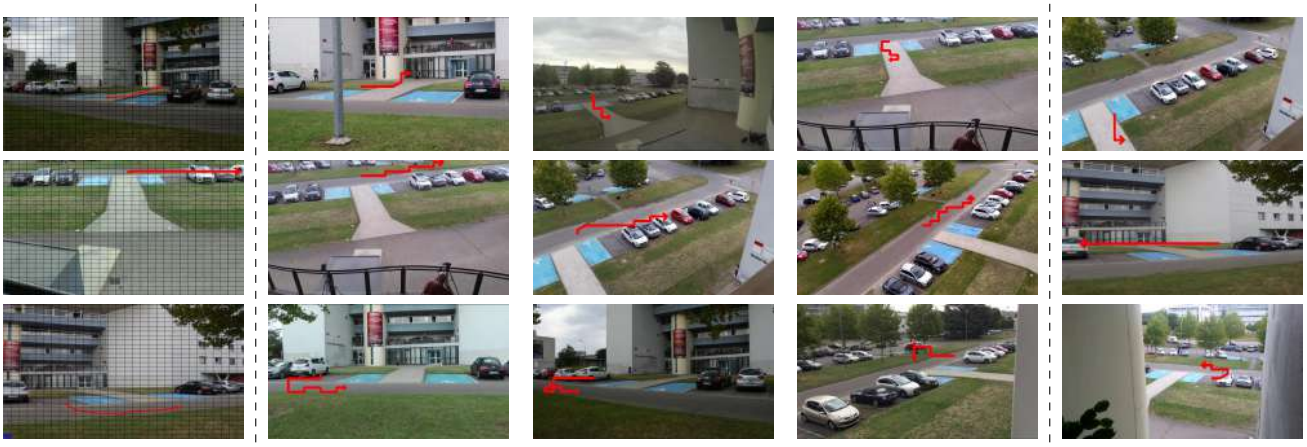

FIGURE 4 – Classement des meilleures vues. Colonne 1 : trois trajectoires requêtes sont tracées, en rouge, respectivement pour les catégories « piéton », « moto » et « voiture ». Colonnes 2 à 4 : les 3 vues qui offrent les meilleurs scores de visibilité sont affichées par ordre décroissant. Colonne 5 : une vue avec un faible score de visibilité.

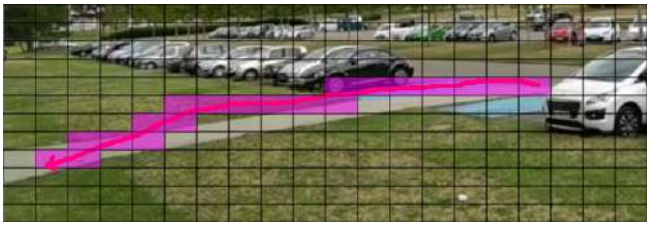

FIGURE 5 – En rouge, une trajectoire requête dessinée dans une vue par l'utilisateur. Les cellules traversées par cette requête sont mises en évidence en rose. C'est à partir des cartes de correspondance entre ces cellules et une autre vue que nous reformulons la trajectoire.

2.3 Sélection et classement des vidéos

Dans cette section, nous exposons notre méthode de sélection et de classement des vidéos en fonction de la visibilité qu'elles offrent d'une trajectoire requête. Les enquêteurs ont bien souvent des dizaines de vidéos à traiter à la fois. Parmi ces vidéos, certaines peuvent être filmées depuis des caméras proches et présenter du recouvrement, tandis que d'autres caméras peuvent être plus éloignées. À partir d'une vidéo de départ, ils peuvent vouloir visualiser ce qui se passe hors champ, ou visualiser plus en détails une trajectoire suivie par un véhicule ou une personne, en naviguant de vidéo en vidéo à partir de requêtes trajectographiques.

Nous proposons ainsi de reformuler la trajectoire tracée par l'utilisateur dans la vidéo de départ dans toutes les autres caméras. Il est assez improbable qu'une vidéo ne présentant aucun recouvrement avec la vidéo de départ parvienne à proposer une reformulation de trajectoire avec un score de reformulation élevé : les rares corrélations apprises entre deux vues indépendantes peuvent être dues à des coïncidences de présence d'objets de même catégorie simultanément dans deux vues, et leurs valeurs sont généralement très faibles. Ainsi, les vues où la trajectoire requête donne un score de reformulation faible peuvent être filtrées en appliquant un seuil σ, déterminé dans nos expérimentations.

Les trajectoires reformulées restantes sont classées selon leur score de visibilité, défini comme le produit de leur score de reformulation et de leur longueur. Nous avons choisi ce critère car nous supposons que plus une trajectoire apparaît longue, plus la vue offre de détails et donc plus elle a d'intérêt pour un enquêteur.

3 Expérimentations

Nous souhaitons évaluer la qualité de nos cartes de correspondance, de nos reformulations de trajectoire ainsi que la pertinence de notre classement des vues qui offrent une meilleure visualisation. Nous avons donc besoin d'un jeu de données contenant une collection de vidéos synchronisées avec du recouvrement dans leurs champs de vue. D'autres vidéos ne présentant pas de recouvrement peuvent également être ajoutées pour vérifier qu'elles sont correctement filtrées et ne font pas partie des vues proposant un bon score de visibilité. Il est préférable que les vidéos durent assez longtemps pour que les cartes de correspondance puissent être apprises : plus la durée commune des vidéos est importante, moins l'on risque de corréler les actions de régions qui ne se correspondent pas.

Nous avons utilisé le jeu de données ToCaDa [8]. Il contient 25 vidéos dans lesquelles une trentaine d'objets de trois catégories (piéton, moto et voiture) sont présents. Parmi toutes les vues, 15 présentent de larges recouvrements et sont situées autour d'un même bâtiment, tandis que les autres vues filment des régions plus éloignées, sans recouvrement, sur un campus universitaire. Les vidéos sont synchronisées et durent près de 5 minutes, ce qui paraît suffisant pour estimer les cartes de correspondance. Nous nous sommes servis des étiquettes de catégorie et des détections

de boîtes englobantes fournies dans le jeu de données pour calculer les cartes de correspondance des cellules.

Nous avons d'abord évalué la qualité des cartes de correspondance entre les 15 vidéos avec recouvrement. Nous estimons qu'une carte de correspondance est de bonne qualité si, lorsqu'à un instant donné, un même objet apparaît simultanément dans deux vidéos, les cartes de correspondance des cellules couvertes par la boîte englobante de l'objet dans la première vidéo offrent un score de correspondance élevé au niveau des cellules couvrant la boîte englobante de la deuxième vidéo. Plus formellement, nous définissons le taux de correspondance entre une paire de boîtes englobantes B_1 et B_2 correspondant à un même objet vu simultanément dans V_1 et V_2 par :

$$\sum_{(c_{\mathbf{i}}^{V_1}, c_{\mathbf{j}}^{V_2})} \frac{|c_{\mathbf{i}}^{V_1} \cap B_1|}{|B_1|} \frac{|c_{\mathbf{j}}^{V_2} \cap B_2|}{|B_2|} \frac{\mathcal{C}_\omega(c_{\mathbf{i}}^{V_1}, c_{\mathbf{j}}^{V_2})}{||\mathcal{C}_\omega(c_{\mathbf{i}}^{V_1}, V_2)||_2} \quad (4)$$

Le premier terme du produit pondère chaque cellule de la première vue par le pourcentage de la boîte englobante qu'elle couvre. Le second terme pondère de la même façon dans la deuxième vue. Le troisième terme pondère finalement par la corrélation entre paires de cellules issues des deux boîtes englobantes. Le taux de correspondance global est obtenu en moyennant l'expression (4) sur l'ensemble des boîtes englobantes simultanées d'un objet, puis en moyennant sur tous les objets communs à la paire de vidéos, puis en moyennant une dernière fois sur toutes les paires possibles de vidéos.

La figure 6 présente les taux de correspondance pour différentes configurations et différents nombres de cellules. Nous avons évalué le comportement de la méthode :

- lorsqu'il n'y a pas de dinstinction entre les catégories : chaque cellule dispose alors d'une seule fonction d'activité mesurant indifféremment la présence d'objets quelle que soit leur catégorie ;
- en créant un décalage d'une seconde entre les paires de vidéos.
- en n'apprenant les cartes de correspondance que sur la moitié de la durée des vidéos au lieu de la totalité.

Les résultats ont révélé que l'étape de calcul des cartes de correspondance est très sensible à ces perturbations, en particulier à la désynchronisation. On remarque également que la distinction des catégories permet d'améliorer grandement le taux de correspondance moyen. On ne s'attend bien entendu pas à avoir des taux de correspondance élevés dans la mesure où la carte de correspondance d'une cellule dans une vidéo couvre généralement un grand nombre de cellules dans l'autre vidéo. En revanche, nous espérons que ces taux de correspondance soient suffisants pour nous permettre de correctement effectuer l'étape suivante : la reformulation.

Pour mesurer la qualité de nos reformulations de trajectoire, nous avons tracé 10 trajectoires requêtes au niveau du sol dans différentes caméras et leur avons appliqué notre

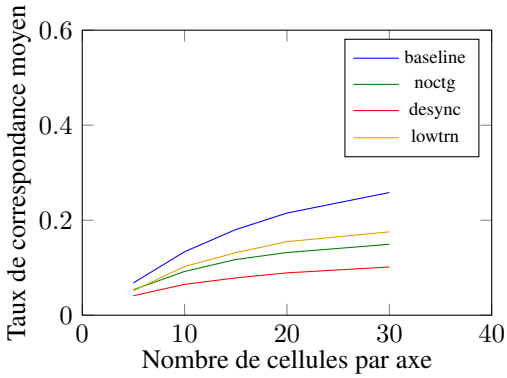

FIGURE 6 – Taux de correspondance moyen sur le jeu de données ToCaDa [8] pour différents nombres de cellules et différentes configurations : sans dégradation (baseline), sans distinction des catégories (noctg), avec une désynchronisation d'une seconde entre les vidéos (desync) et en apprenant les cartes de correspondance sur seulement la moitié du jeu de données (lowtrn).

méthode de reformulation pour obtenir les trajectoires reformulées dans les autres caméras. Pour se comparer à une trajectoire de référence, nous avons calculé l'homographie plane entre chaque paire de caméras à partir des coins des places de parking bleues. Puis, nous avons mesuré la distance *Dynamic Time Warping* (DTW) [11], en pixels, entre les trajectoires obtenues à partir de l'homographie et les trajectoires reformulées.

La figure 7 présente la distance DTW moyenne, en pixels, sur des vidéos de tailles 960×540 pour différents nombres de cellules. La trajectoire reformulée est plus fiable à mesure que le nombre de pixels augmente. En effet, comme nous obtenons la trajectoire reformulée en liant les centres des cellules, celle-ci peut s'approcher davantage de la trajectoire obtenue par homographie lorsqu'il y a davantage de cellules. Ces résultats valident notre méthode de reformulation de trajectoire.

Concernant le classement des vidéos, les vues qui ne présentent pas de recouvrement sont correctemnt filtrées en utilisant un seuil σ = 0,3. En effet, quasiment aucune correspondance ne peut être apprise du fait de l'absence de présence simultanée d'objets de la même catégorie au cours du temps. La figure 4 présente sur chaque ligne :

- à gauche, une trajectoire requête tracée manuellement dans une vue ;
- au milieu, les trois vidéos correspondant aux 3 meilleurs scores de visibilité, avec la trajectoire reformulée correspondante ;
- à droite, une vue classée parmi les dernières. Ces vues sont généralement des vues dans lesquelles on n'arrive pas à reformuler la totalité de la trajectoire requête, ou une vue où cette trajectoire reformulée occupe peu d'espace et ne permet pas de voir plus de détails.

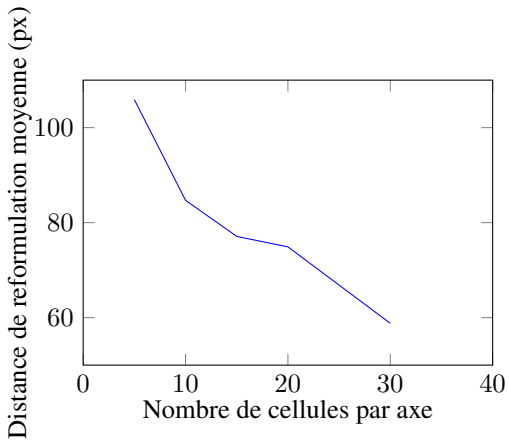

FIGURE 7 – Distance de reformulation moyenne DTW en pixels pour différents nombres de cellules.

Parmi les 5 meilleures vues obtenues sur nos 10 trajectoires requêtes, 72% d'entre elles donnent effectivement une meilleure ou aussi bonne visibilité de la trajectoire requête.

4 Conclusion

À partir d'une collection de vidéos et sans se servir des paramètres de calibrage des caméras, la méthode proposée permet de classer avec succès un sous-ensemble des vidéos qui présentent du recouvrement avec une vidéo dans laquelle une requête trajectographique est tracée en fonction de la visibilité qu'elles offrent de la trajectoire reformulée. Les vidéos qui ne sont pas liées à la scène sont également correctement filtrées. Il n'est fait usage d'aucune méthode de ré-identification, la simple présence simultanée d'objets de même catégorie dans différentes caméras permet d'apprendre les correspondances spatiales.

La méthode est particulièrement adaptée à un contexte de vidéo-surveillance en extérieur du fait des heures d'enregistrement vidéo disponibles depuis divers points de vue, permettant d'obtenir des cartes de correspondance robustes. De futurs travaux pourront approfondir notre approche en relâchant notamment la contrainte de synchronisation des vidéos et en essayant d'estimer le décalage temporel. D'autre part, les trajectoires étant formulées au niveau du sol, les cartes de correspondance gagneraient peut-être à ne mesurer la présence simultanée qu'au niveau des cellules du sol, c'est-à-dire au bas des détections. Cela permettrait d'éviter qu'une cellule dans une vue ait une carte de correspondance qui couvre des très nombreuses cellules dans une autre vue. Le voisinage d'une cellule pourrait également être exploité pour éviter des discontinuités entre les cartes de correspondance de certaines cellules adjacentes. Enfin, la photométrie des objets détectés pourrait être utilisée pour mieux apprendre les correspondances entre régions, en distinguant davantage les objets, non plus seulement par catégorie, mais aussi par leur apparence.

Références

[1] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, et R. Szeliski, Building Rome in a day, *IEEE International Conference on Computer Vision*, 2009.

[2] A. Carlier, L. Calvet, P. Gurdjos, V. Charvillat et W. T. Ooi, Querying Multiple Simultaneous Video Streams with 3D Interest Maps, *Visual Content Indexing and Retrieval with Psycho-Visual Models*, 2017.

[3] Y.-J. Cho, S.-A. Kim, J.-H Park, K. Lee et K.-J. Yoon, Joint person re-identification and camera network topology inference in multiple cameras, *Computer Vision and Image Understanding*, 2019.

[4] F. Daniyal, M. Taj et A. Cavallaro, Content and task-based view selection from multiple video streams, *Multimedia tools and applications*, 2010.

[5] S. Khan et M. Shah, Consistent labeling of tracked objects in multiple cameras with overlapping fields of view, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003.

[6] C. C. Loy, T. Xiang et S. Gong, Time-delayed correlation analysis for multi-camera activity understanding, *International Journal of Computer Vision*, 2010.

[7] C. C. Loy, T. Xiang et S. Gong, Incremental activity modeling in multiple disjoint cameras, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.

[8] T. Malon, G. Roman-Jimenez, P. Guyot, S. Chambon, V. Charvillat, A. Crouzil, A. Péninou, J. Pinquier, F. Sèdes et C. Sénac, Toulouse campus surveillance dataset : scenarios, soundtracks, synchronized videos with overlapping and disjoint views, *ACM Multimedia Systems Conference*, 2018.

[9] A. Mustafa, H. Kim, J.-Y. Guillemaut et A. Hilton, Temporally coherent 4D reconstruction of complex dynamic scenes, *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[10] G. Ning, Z. Zhang, C. Huang, X. Ren, H. Wang, C. Cai et Z. He, Spatially supervised recurrent convolutional neural networks for visual object tracking, *IEEE International Symposium on Circuits and Systems*, 2017.

[11] Z. Zhang, K. Huang et T. Tan, Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes, *IEEE International Conference on Pattern Recognition*, 2006.