Partitionnement de coordonnées spatiales dans un environnement multi-caméra

Marc-Olivier Buob[1], Pierre Escamilla[1], Jeongran Lee[1]

Nokia Bell Labs {marc-olivier.buob, pierre.escamilla,jeongran.lee}@nokia-bell-labs.com

Mots-clés : *Optimisation, programme linéaire en nombre entiers, vision par ordinateur.*

1 Introduction

On considère un ensemble de caméras filmant une scène dans laquelle évoluent des agents sur un terrain plat (typiquement, des sportifs sur un terrain de sport). Chaque caméra ne perçoit qu'une partie du terrain. Un détecteur (par exemple YOLO [1]) permet de retrouver sur chaque frame un rectangle englobant de pixels qui entoure chaque agent (détection). À l'aide des caractéristiques des caméras (position, lentille), on peut déduire les coordonnées monde approximatives de chaque détection. L'objectif est d'unifier les coordonnées ainsi obtenues de sorte à placer l'ensemble des agents sur une carte.

Ce problème comporte plusieurs difficultés. D'une part, les incertitudes liées aux caractéristiques des caméras introduisent du bruit sur les coordonnées obtenues. D'autre part, un détecteur peut manquer un agent qui est pourtant dans son champ de vision (faux négatifs).

2 Formulation du problème

On considère une capture obtenue depuis m caméras et comportant n agents. On suppose qu'un agent peut être vu par plusieurs caméras, mais est capturé au plus une fois par chaque caméra (on néglige donc les faux positifs). On prend en compte les faux négatifs avec probabilité p sans à priori sur les performances du détecteur. Ainsi, deux détections capturées par deux caméras distinctes ne peuvent correspondre à un même agent que si elles sont proches.

On note $V = \{0, \ldots, n-1\}$ l'ensemble des détections et $C = \{0, \ldots, m-1\}$ l'ensemble des caméras. Soit $c : V \to C$ la fonction qui associe à chaque détection la caméra permettant de l'obtenir. Soit $d : V \times V \to \mathbb{R}^+$ la distance entre deux détections ainsi définie :

$$d(u,v) = \begin{cases} \|u - v\|_2 & \text{si } c(u) \neq c(v) \wedge \|u - v\|_2 \leq D \\ +\infty & \text{sinon} \end{cases} \tag{1}$$

où $\|.\|_2$ est la norme euclidienne. Les variables de décisions $z_{uv} \in \{0, 1\}$ indiquent si deux détections $u \in V, v \in V, u < v$ appartiennent au même cluster (et sont donc associées à un même agent). La fonction objectif de notre programme linéaire en nombres entiers (PLNE) est définie par :

$$\min \sum_{(u,v) \in V^2, u<v} (d(u,v) \cdot z_{uv} + F \cdot (1 - z_{uv})) \tag{2}$$

où $F \gg \max_{(u,v) \in V^2, u<v} d(u,v)$ est une pénalité appliquée lorsque deux détections ne sont pas assignées au même cluster. Notons que prendre des valeurs de F moindres permettrait d'ajuster la confiance accordée aux capacités de localisation des détecteurs.

1. https://pjreddie.com/darknet/yolo/

Pour chaque agent, on suppose que la distance entre les coordonnées obtenues par les différents capteurs ne peut excéder la distance $D \in \mathbb{R}^{+*}$. De plus, chaque agent est détecté au plus une fois par caméra. On a donc :

$$d(u,v) > D \vee c(u) = c(v) \implies z_{uv} = 0 \quad (\forall (u,v) \in V^2, u < v) \tag{3}$$

Comme une détection $u \in V$ est associée à au plus une autre détection d'une autre caméra :

$$\sum_{v \in V, c(v) = c_v} z_{\min(u,v),\max(u,v)} \leq 1 \quad (\forall u \in V, \forall c_v \in C \setminus \{c(u)\}) \tag{4}$$

En s'inspirant de [1], il ne reste plus qu'à ajouter les contraintes de transitivité inhérentes aux clusters. Pour tout $(u,v,w) \in V^3, u < v < w$:

$$\begin{cases} z_{uv} + z_{uw} - z_{vw} \leq 1 \ ; \ z_{uv} + z_{vw} - z_{uv} \leq 1 \ ; \ z_{uw} + z_{uv} - z_{vw} \leq 1 \\ z_{uw} + z_{vw} - z_{uv} \leq 1 \ ; \ z_{vw} + z_{uv} - z_{uw} \leq 1 \ ; \ z_{vw} + z_{uw} - z_{uv} \leq 1 \end{cases} \tag{5}$$

3 Validation expérimentale

$(m;n)$	$(5\,;\,5)$	$(6\,;\,10)$	$(6\,;\,20)$
#variables, #contraintes	(250,5 ; 6 011,6)	(1 456,5 ; 91 193,8)	(5 799.0 ; 708 360,3)
Temps de résolution (s)	0,48	3,51	18,91
Indice de Rand ajusté	0,95	0,92	0,88

TAB. 1 – Résultats expérimentaux moyennés sur 100 expériences.

Dans nos expériences, $p = 0,1$ et $D = 5$. Les positions suivent la loi $\mathcal{U}([-13\,;\,13] \times [-30\,;\,30])$. Pour chaque position, la détection associée à la caméra i est obtenue en ajoutant un bruit gaussien de matrice de covariance diagonal $\sigma_i \sim \mathcal{U}([0\,;\,2]^2)$. Nos expériences sont réalisées avec PuLP sur un PC muni d'un CPU i7@1.8Ghz. Le tableau 1 montre que notre problème passe bien à l'échelle et conduit à un indice de Rand ajusté [2] satisfaisant, *i.e.* proche de 1.

4 Travaux connexes

[2] présente un modèle valable pour deux caméras qui donne lieu à une formulation sous forme d'un programme linéaire. [3] généralise notre problème lorsque la performance (précision, rappel) des détecteurs est connu, au prix d'un modèle plus coûteux à résoudre.

5 Conclusion

Nous avons présenté un PLNE permettant de rassembler les détections liées à un même agent dans un environnement multi-caméras. On peut aisément étendre ce modèle pour prendre en compte d'autres caractéristiques (couleurs des vêtements, vélocité, etc.) propres aux agents .

Références

[1] Ágoston, Kolos Cs and E.-Nagy, Marianna. Mixed ILP formulation for K-means clustering problem. *Central European Journal of Operations Research*, p1–17, 2023.

[2] Kettnaker Vera and Zabih Ramin. Bayesian multi-camera surveillance. *CVPR 1999*, p2470–2477, 1991.

[3] Hofmann, Martin and Wolf, Daniel and Rigoll, Gerhard. Hypergraphs for Joint Multi-view Reconstruction and Multi-object Tracking. *CVPR 2013*, p3650–3657, 2013.

2. https://en.wikipedia.org/wiki/Rand_index#Adjusted_Rand_index