UNIVERSITÉ DE MONTRÉAL

SUIVI D'OBJETS PAR CARACTÉRISTIQUES LOCALES ENCODANT LA STRUCTURE

WASSIM BOUACHIR
DÉPARTEMENT DE GÉNIE INFORMATIQUE ET GÉNIE LOGICIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION
DU DIPLÔME DE PHILOSOPHIÆ DOCTOR
(GÉNIE INFORMATIQUE)
DÉCEMBRE 2014

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

SUIVI D'OBJETS PAR CARACTÉRISTIQUES LOCALES ENCODANT LA STRUCTURE

présentée par : BOUACHIR Wassim
en vue de l'obtention du diplôme de : Philosophiæ Doctor
a été dûment acceptée par le jury d'examen constitué de :

Mme CHERIET Farida, Ph.D., présidente
M. BILODEAU Guillaume-Alexandre, Ph.D., membre et directeur de recherche
M. KADOURY Samuel, Ph.D., membre
M. COULOMBE Stéphane, Ph.D., membre externe

À ma femme et à mes parents
Ce travail est le votre...

REMERCIEMENTS

Je tiens à exprimer ma reconnaissance et ma profonde gratitude à mon directeur de recherche M. Guillaume-Alexandre Bilodeau. Qu'il soit vivement remercié pour l'encadrement qu'il m'a offert tout au long de mon doctorat, pour sa disponibilité perpétuelle et pour les soutiens financiers et scientifiques dont j'ai bénéficié auprès de lui. Je voudrais aussi remercier le Fonds de Recherche du Québec – Nature et Technologies (FRQNT) de m'avoir accordé une bourse de recherche doctorale qui a financé partiellement mon doctorat.

Je salue mes collègues au laboratoire LITIV : Dorra, Pierre-Luc, Jean-Philippe et Tanushri. Nos discussions m'ont permis d'enrichir ma réflexion, tout en ajoutant une dimension sociale à nos activités dans le laboratoire. Je leur souhaite une bonne continuation et les meilleurs succès dans leurs carrières.

Je remercie finalement chaque membre du jury : Mme Farida Cheriet, M. Samuel Kadoury et M. Stéphane Coulombe, qui ont eu l'amabilité d'examiner ce travail.

RÉSUMÉ

Durant les deux dernières décennies, le suivi visuel d'objets a retenu une attention considérable de la communauté de la vision par ordinateur. Cet intérêt accru s'explique par les avancées importantes réalisées dans la modélisation de contenu visuel et par la large gamme d'applications utilisant les algorithmes de suivi (vidéosurveillance, robotique, indexation multimédia, interaction homme/machine, etc.). Bien que de nombreuses méthodes sont proposées dans la littérature, le suivi d'objet demeure un problème non résolu à cause du nombre élevé de facteurs environnementaux.

Cette thèse présente de nouvelles idées et méthodes pour suivre un objet en mouvement dans des scénarios du monde réel. Elle vise à résoudre les principales difficultés de suivi dans des environnements non contraints, où la seule connaissance disponible sur la cible est sa position sur la première trame de la séquence vidéo. Les problèmes de suivi traités comprennent l'occultation de la cible, l'apparition de distracteurs, les rotations, les changements des conditions d'illumination et les changements d'apparence de la cible. Dans notre travail, nous présentons des algorithmes de suivi complets, ainsi que des procédures génériques pouvant être intégrées dans d'autres méthodes de suivi.

Les méthodes proposées dans cette thèse s'appuient sur les points caractéristiques SIFT (Scale Invariant Features Transform) pour assurer la distinctivité du modèle d'apparence et son invariance aux changements des conditions d'illumination. Premièrement, nous présentons un algorithme qui combine une approche de recherche probabiliste par caractéristiques de couleurs avec un suivi déterministe par points caractéristiques. La recherche probabiliste consiste à un filtre de particules appliqué dans une première étape pour réduire la région de détection des caractéristiques locales et simplifier leur appariement. La prédiction finale repose ensuite sur l'appariement entre les descripteurs locaux de la cible et ceux détectés dans la région de recherche réduite sur la trame courante. L'évaluation qualitative et quantitative sur plusieurs séquences vidéos démontre la validité de la méthode proposée et sa compétitivité avec des algorithmes de suivi populaires de la littérature.

En second lieu, nous présentons un nouvel algorithme de suivi nommé SAT (Structure-Aware Tracker). Comme son nom l'indique, SAT utilise un modèle d'apparence qui tient compte de la structure interne de l'objet suivi. Notre idée clé est inspirée de travaux antérieurs sur un nouveau paradigme de suivi, dit orienté contexte. Dans le suivi orienté contexte, la structure de la scène est encodée par les relations géométriques entre la cible et d'autres éléments l'entourant. Dans SAT, nous encodons la structure de l'objet en exploitant la disposition spatiale de ses points caractéristiques. Cette technique permet d'atteindre une stabilité

de suivi notable, même lorsque la cible est partiellement occultée. D'autre part, nous proposons une méthode discriminative pour évaluer la qualité de suivi itérativement. Lorsque l'évaluation montre une qualité acceptable, le modèle d'apparence est adapté aux éventuels changements d'apparence de la cible. Notons que la procédure d'évaluation proposée est généralisable pour la majorité des algorithmes de suivi, vu qu'elle est indépendante du modèle d'apparence principal de la cible.

En dernier lieu, nous présentons un troisième algorithme de suivi nommé SCFT (Salient Collaborating Features Tracker). Cette méthode exploite les descripteurs d'une manière optimale pour capturer la structure interne de l'objet et tenir compte des changements de pose et d'échelle. En effet, SCFT utilise les orientations dominantes et les échelles caractéristiques des points SIFT pour calculer les rotations bidimensionnelles de l'objet et estimer sa taille. En outre, nous proposons d'évaluer la saillance des points caractéristiques durant le suivi afin de baser les prédictions sur les caractéristiques locales les plus fiables. On note que la procédure d'évaluation de saillance peut être adoptée par d'autres méthodes de suivi basées sur le vote des éléments locaux.

Dans la partie expérimentale de ce travail, nous réalisons plusieurs tests confirmant la robustesse des méthodes présentées. Nos expériences montrent que l'exploitation efficace des descripteurs locaux de points caractéristiques permet de prédire avec précision la position de la cible, malgré les facteurs perturbateurs de la scène. Par ailleurs, les algorithmes proposés (notamment SAT et SCFT) ne sont pas affectés par l'occultation partielle, vu qu'un nombre réduit de caractéristiques locales visibles suffit pour prédire l'état global de l'objet. Les évaluations comparatives de nos méthodes confirment la pertinence des idées développées dans cette thèse, en démontrant que nous surpassons une variété d'algorithmes récents de la littérature dans divers scénarios difficiles de suivi.

ABSTRACT

Object tracking is a central problem in computer vision with many applications, such as automated surveillance, robotics, content-based video indexation, human-computer interaction, etc. During the two last decades, we observed an increasing interest in developing novel solutions for the tracking problem. This interest is explained by the significant progress achieved in feature extraction and visual modeling. Despite numerous methods proposed in the literature, object tracking remains an unsolved problem due to the large number of environmental perturbation factors.

This thesis presents novel ideas and methods for object tracking in real world scenarios. We aim to address the main tracking difficulties in unconstrained environments, including target occlusion, presence of distractors, object rotations, illumination changes, and target appearance change. Our work proposes complete tracking algorithms that may be adopted directly by several tracking systems, as well as generic procedures that are relevant to the development of future tracking algorithms.

The proposed algorithms rely on SIFT (Scale Invariant Features Transform) keypoints due to their distinctiveness and invariance to illumination changes and image noise. Firstly, we present a novel tracking algorithm whose prediction combines a color-based probabilistic approach with a deterministic keypoint matching method. During the first algorithmic step, we apply probabilistic tracking through particle filtering. This allows to reduce the target search space while simplifying local matches between the reference model and candidate regions. The target position is then found by matching keypoints and selecting the candidate region (particle) having the best matching score. Qualitative and quantitative evaluations on challenging video sequences show the validity of the proposed tracker and its competitiveness with popular state-of-the-art trackers.

Secondly, we present a new tracker named SAT (Structure-Aware Tracker). The proposed algorithm uses a novel appearance model to encode the internal structure of the target. Our idea of representing the target structural properties is inspired by previous works on context tracking. According to the context tracking approach, it is necessary to consider the target context to ensure the tracker robustness. Thereby, context trackers base their predictions on the structure of the scene, encoded by the geometric relations between the target and surrounding elements. SAT is different from context trackers, in the sense that it encodes the object internal structure by exploiting the object keypoints spatial layout. This technique allows to achieve robust and stable tracking, especially when the target is partly occluded. Moreover, we propose a discriminative method for evaluating the tracking quality after each

prediction. Every time the verification procedure shows a good tracking quality, the target appearance model is updated to be adapted to possible appearance changes. Note that the quality evaluation procedure of SAT is generalizable for a wide variety of tracking algorithms, since it does not depend on the target main appearance model.

Furthermore, we introduce SCFT (Salient Collaborating Features Tracker), a novel tracking method that exploits keypoint descriptors efficiently to handle the target pose and scale changes. More concretely, the proposed method uses the information on the main orientation and the detection scale of the local feature to compute respectively in-plane rotations and scale changes. On the other hand, SCFT evaluates local features saliency in order to distinguish between good features and outliers, and base the prediction on the most reliable ones. Our saliency evaluation method can be used directly or adapted for several existing tracking methods, where the target state is found through local features votes.

Our experimental work includes multiple tests on challenging video sequences, showing the robustness of the presented methods. Through the performed experiments, we demonstrate that our efficient exploitation of local descriptors ensures a high tracking precision, under several environmental perturbation factors. Moreover, the proposed methods (especially SAT and SCFT) are not affected by partial occlusions, since in this case, the global target state is effectively predicted using a few number of visible local features. We performed extensive comparative evaluations on challenging video sequences, against recent state-of-the-art methods. The obtained results support the relevance of the proposed ideas, demonstrating that we outperform recent methods in various real world scenarios.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

AS	Algorithme de Suivi
AST	Adaptive Structural Tracker
BRISK	Binary Robust Invariant Scalable Keypoints
CLE	Center Location Error
FAST	Features from Accelerated Segment Test
GHT	Generalized Hough Transform
KLT	Kanade-Lucas-Tomasi
LBP	Local Binary Patterns
MIL	Multiple Instance Learning
MSIT	online Multiple Support Instance Tracker
OAB	Online AdaBoost
OMSIT	Online Multiple Support Instance Tracker
OR	Overlap Ratio
PTZ	Pan/ Tilt/ Zoom
RANSAC	RANdom SAmple Consensus
RF	Reservoir of Features
RGB	Red, Green, Blue
RI	Région d'Intérêt
RVB	Rouge, Vert, Bleu
SAT	Structure-Aware Tracker
SBCT	Sparsity-Based Collaborative Tracker
SCFT	Salient Collaborating Features Tracker
SCMT	Sparsity-based Collaborative Model Tracker
SIFT	Scale Invariant Feature Transform
SLIC	Simple Linear Iterative Clustering
SPT	SuperPixel Tracker
SURF	Speeded Up Robust Features
TLD	Tracking-Learning-Detection
TSL	Teinte, Saturation, Luminosité

CHAPITRE 1

INTRODUCTION

De nos jours, les installations de vidéosurveillance génèrent quotidiennement une quantité colossale de données multimédia. L'analyse manuelle des données (enregistrées ou visionnées en temps réel) est une tâche fastidieuse présentant plusieurs limites. Il a été par exemple démontré que l'attention de l'opérateur humain chute considérablement après 20 minutes de concentration et qu'il est impossible de suivre attentivement neuf à douze caméras plus de 15 minutes [1]. Doter les systèmes de vidéosurveillance de fonctionnalités intelligentes est devenu une nécessité promettant plusieurs avantages, tels que le fonctionnement en continu du système en libérant le personnel de sécurité, la détection automatique des évènements d'intérêt, le déclenchement automatique des alarmes et l'identification automatique des personnes.

L'algorithme de suivi (AS) représente la composante principale de la plupart des systèmes de vidéosurveillance intelligente, ainsi que dans un large éventail d'autres applications de la vision artificielle telles que les applications d'interaction personne-machine, la navigation des robots et l'indexation basée sur le mouvement dans les séquences vidéos. D'une manière générale, la tâche de suivi consiste à localiser un objet en mouvement sur chaque trame d'une séquence vidéo. À l'initialisation, l'AS construit un modèle de l'objet d'intérêt. Le modèle d'apparence peut être basé sur des connaissances à priori (p. ex. images prises hors-ligne de différents points de vue, différentes conditions d'illumination, etc.) ou seulement sur l'annotation de la région de l'objet sur la première trame (p. ex. en utilisant une boite englobante). L'algorithme estime ensuite l'état de l'objet sur toutes les trames subséquentes. Selon le domaine d'application, l'état de l'objet peut se limiter à une position dans l'image (p. ex. position du centre de l'objet), ou inclure d'autres informations telles que la taille, l'orientation, le contour exact de l'objet, etc. La figure 1.1 illustre le fonctionnement général d'un AS sans modèle d'objet à priori.

Le suivi automatique d'objets en mouvement est l'un des domaines de recherche les plus actifs de la vision par ordinateur. Le problème de suivi demeure non résolu à cause de nombreuses difficultés du monde réel. L'algorithme de suivi doit maintenir une précision de localisation acceptable dans des situations complexes du monde réel, selon les exigences de l'application. Parmi les difficultés rencontrées, on retrouve :

- le manque d'informations sur l'apparence des objets à cause de la projection d'un monde tridimensionnel sur des images à deux dimensions ;
- la présence d'éléments perturbateurs dans l'environnement de l'objet tel que les change-

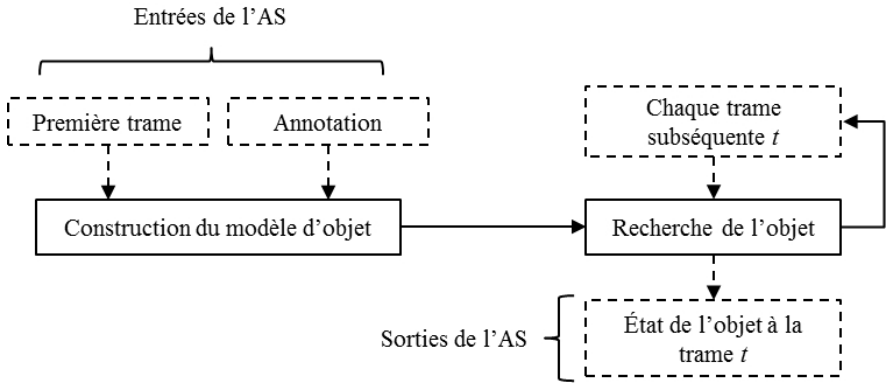


Figure 1.1 Illustration d'un algorithme général de suivi sans modèle à priori de l'objet

ments des conditions d'illumination, la réflexion de la lumière, la similarité d'apparence avec l'arrière-plan, etc. ;
- la complexité de l'apparence des objets suivis (p. ex. objets articulés) et les changements d'apparence et de point de vue durant le suivi ;
- l'interaction de la cible avec d'autres objets pouvant l'occulter ou distraire l'AS ;
- les mouvements complexes et aléatoires difficilement modélisables des objets.

Étant donné la complexité du suivi et la multitude des facteurs perturbateurs, la majorité des travaux existants simplifient la tâche en imposant des contraintes sur certains éléments du problème tels que l'apparence des objets, les mouvements effectués et l'environnement de l'objet. Il est possible, par exemple, de supposer que le déplacement de la cible entre deux trames successives ne dépasse pas une limite supérieure, ou que celle-ci se déplace à une vitesse constante. On peut également supposer le statisme de l'arrière-plan ou l'immobilité de la caméra afin de détecter efficacement les objets en mouvement.

La littérature de la vision par ordinateur propose une multitude de méthodes de suivi. Les méthodes proposées se distinguent principalement par (1) le modèle d'apparence décrivant la cible et (2) la stratégie de recherche utilisée pour en prédire l'état. Ces deux composantes représentent les deux aspects majeurs d'un AS et doivent être conçues en tenant compte de l'environnement et de la finalité du suivi.

Dans cette thèse, nous nous intéressons au problème de suivi d'objets génériques en déplacement arbitraire, sans aucune connaissance préalable autre que l'état de l'objet dans la première trame de la séquence vidéo. Cette tâche est souvent appelée suivi sans modèle à priori (en anglais *model-free tracking*). Dans le suivi sans modèle à priori, la seule information fournie à l'AS est la région de l'objet cible annotée manuellement sur la première image, le plus souvent délimitée par une forme géométrique. Cette limitation accroit les difficultés de suivi pour deux raisons principales :

- le manque d'informations sur l'apparence de la cible vue qu'une seule vue est disponible ;
- l'imprécision dans la distinction entre l'objet et son arrière-plan, car des pixels de ce dernier sont souvent inclus dans la boite englobante (ou forme géométrique délimitant l'objet).

Le reste de ce chapitre introductif est organisé comme suit. Les problématiques de recherche sont discutées dans la section suivante. Dans la section 1.2, nous énonçons les objectifs du projet. Les sections 1.3 et 1.4 présentent respectivement les contributions de ce travail et la structure de la thèse.

1.1 Problématiques

Malgré les avancées importantes réalisées en vision par ordinateur (notamment dans la modélisation des Régions d'Intérêt (RI), le calcul des caractéristiques visuelles, la modélisation des mouvements et la détection d'objets), la conception d'un modèle d'apparence robuste et d'une stratégie de recherche efficace sans connaissance à priori du modèle d'apparence demeure un défi de taille. Les difficultés émanent à la fois des environnements non contraints et des apparences et mouvements complexes des objets suivis.

Occultation partielle. Cette situation se présente lorsqu'une partie de l'objet suivi est cachée derrière un autre élément de la scène. Le plus souvent, l'élément occultant est un autre objet en mouvement (figure 1.2a) ou une structure de l'arrière-plan (figures 1.2b et 1.2c). Il s'agit d'une situation fréquente dans les environnements non contraints, vu que les objets suivis interagissent (activement ou passivement) avec d'autres éléments de la scène. Une telle situation peut affecter la précision de l'AS, étant donné qu'il ne dispose plus des caractéristiques de la partie cachée de l'objet. L'occultation partielle est aussi l'un des facteurs majeurs causant la contamination du modèle de la cible. Le modèle risque ainsi d'évoluer d'une manière imprévue en incluant des caractéristiques de l'objet occultant la cible. Les prédictions deviennent moins exactes au fur et à mesure que le temps passe et l'AS peut dévier pour suivre l'objet occultant ou l'arrière-plan. Ce comportement non souhaitable est connu sous le nom de dérive de l'AS.

Occultation totale, sortie du champ visuel. L'occultation totale de l'objet cible et la sortie du champ de vue sont des facteurs critiques affectant la performance de suivi. Dans les deux cas, l'objet devient complètement invisible (pour une longue ou courte durée) et le suivi doit être récupéré dès qu'une partie de la cible réapparait. L'AS peut être doté d'un mécanisme de récupération par recherche exhaustive dans toute l'image, ce qui risque d'être couteux. Il est possible également d'imposer des contraintes de mouvement (telle qu'une

(a) (b) (c)

Figure 1.2 Illustrations du problème d'occultation partielle. La figure 1.2a montre une situation d'occultation partielle par un objet en mouvement (séquence *faceocc* [2]). Dans les figures 1.2b (séquence *wdesk*) et 1.2c (séquence *tiger1* [3]) la cible est partiellement cachée par des éléments de l'arrière-plan.

vitesse constante) pour prédire la position de l'objet, mais ces hypothèses sont souvent violées dans des scénarios concrets du monde réel. La figure 1.3 montre un exemple d'occultation totale d'une personne cible par une autre personne traversant le champ de vision de la caméra.

Apparition d'éléments distracteurs Ce problème se présente lorsque la scène surveillée comprend des objets similaires à la cible du point de vue de l'apparence. Ces objets peuvent être confondus avec la cible et risquent de détourner l'algorithme de suivi, d'où leur qualification de distracteurs. La figure 1.4 présente l'exemple typique de suivi de visage dans une scène moyennement chargée (quatre personnes). Ce scénario représente un défi majeur causant la dérive de l'AS. Il est à noter que la distraction de l'AS peut aussi être causée par un arrière-plan dont la couleur ou la texture est similaire à celle de la cible.

(a) (b)

Figure 1.3 Exemple d'occultation totale d'une personne en mouvement par un autre individu traversant le champ de vision de la caméra (séquence *jp2*).

Figure 1.4 Suivi de visage en présence de distracteurs sur la séquence vidéo *jp1*.

Rotation dans le plan de l'image Plusieurs AS supposent que la pose de l'objet ne change pas durant le suivi. À titre d'exemple, plusieurs applications de vidéosurveillance de personnes représentent des parties du corps humain (torse et/ou tête) par des ellipses verticales de pose fixe ou avec un changement mineur d'orientation [4, 5]. Dans le monde réel, des rotations dans le plan de l'image (en anglais *in-plane rotations*) peuvent engendrer des changements de pose considérables. L'efficacité et la flexibilité d'un AS dépendent en partie de sa capacité à tenir compte des changements d'orientation.

Rotation tridimensionnelle Une séquence vidéo capturée par une caméra correspond à des projections d'un monde tridimensionnel sur des images à deux dimensions. Dans le suivi sans modèle à priori, une seule vue de l'objet est connue par l'AS. Une rotation tridimensionnelle de la cible peut ainsi produire un grand changement d'apparence, affectant la qualité de suivi (p. ex. une personne tournant la tête lors d'un suivi de visage). Si la rotation est graduelle, le modèle d'apparence doit être en mesure de s'adapter aux changements d'apparence. Cependant, le changement brusque de l'angle de vue de l'objet doit être traité comme une occultation totale ou une sortie du champ visuel (c.-à-d. récupération après une perte totale de l'objet).

Changement d'illumination Les images enregistrées par une caméra dépendent du contenu physique de la scène aussi bien que des conditions d'éclairage. Considérons l'exemple d'un objet en mouvement dans une pièce éclairée par une lampe à incandescence. Un AS utilisant des caractéristiques de couleur risque d'être perturbé par le déplacement de la cible vers une fenêtre l'exposant à la lumière du jour. Le changement des conditions d'éclairage peut s'effectuer d'une manière abrupte ou graduelle. Dans l'exemple de la figure 1.5, le visage suivi devient progressivement plus illuminé pendant que l'individu suivi se déplace d'une pièce

sombre vers un espace plus éclairé.

Mouvements de caméra Certains systèmes de suivi supposent le statisme de la caméra pour détecter les objets en mouvement [7, 8]. En effet, il est plus facile de modéliser l'arrière-plan lorsque la caméra est fixe. Les régions de l'image où des changements se produisent sont ainsi extraites et considérées comme des cibles potentielles (car ne suivant pas le modèle de l'arrière-plan). Cependant, la contrainte de la caméra stationnaire n'est souvent pas valide. Par exemple, un suivi hors-ligne peut être effectué sur une séquence vidéo enregistrée par une caméra déplacée et orientée par un opérateur humain. Une autre application entrainant le dynamisme de l'arrière-plan est celle du suivi en ligne automatisé par caméra PTZ (*Pan/ Tilt/ Zoom*).

Objets articulés Les objets non rigides (p. ex. main, corps humain, animal) peuvent subir des changements importants d'apparence suite aux déformations des parties articulées (voir la figure 1.6). Si la catégorie des objets à suivre est connue à l'avance, il est possible de spécifier l'apparence par des modèles sophistiqués représentant les différentes parties articulées, leur disposition et les transformations possibles. À cet effet, des travaux de suivi visuel proposent des modèles structurels représentant le corps humain en tenant compte de la cinématique humaine [9, 10]. Toutefois, trouver une solution commune qui tient compte de l'éventuelle nature articulée des objets est une autre problématique à résoudre dans la conception de modèles d'apparence génériques.

(a) (b)

Figure 1.5 Exemple de changement des conditions d'éclairage lors d'un suivi de visage sur la séquence vidéo *David* [6].

(a) (b)

Figure 1.6 Exemple de suivi d'objet articulé sur la séquence *tiger1* [3].

1.2 Objectifs de la recherche

L'objectif principal de cette thèse est de proposer de nouvelles idées et techniques pour résoudre le problème de suivi visuel d'objet générique dans un environnement non contraint. Plusieurs solutions existantes imposent des contraintes sur l'objet suivi, la nature de son mouvement, et/ou sur la scène où il se trouve. Notre objectif est de développer des techniques permettant de faire face aux difficultés discutées dans la section 1.1, fonctionnant sous un contrôle minimum sur l'objet et son environnement et sans connaissance à priori du modèle d'apparence. Les objectifs spécifiques de ce travail sont les suivants :

1. Modéliser l'apparence de l'objet cible par des caractéristiques robustes aux différents facteurs perturbateurs de l'environnement (p. ex. changement d'illumination, bruit, distracteurs, etc.) et adaptatives aux principaux changements d'apparence de l'objet (p. ex. changement de point de vue, déformation).
2. Prédire avec précision l'état de l'objet cible malgré les déplacements aléatoires de ce dernier, les rotations (coplanaires et tridimensionnelles) et les mouvements de caméra.
3. Gérer efficacement les différentes situations d'occultation partielle et totale.
4. Valider les performances des méthodes implémentées à travers des scénarios concrets de suivi et quantifier les résultats par des métriques standards.

Il est à noter que le problème d'occultation totale peut être traité en imposant des contraintes sur le mouvement de la cible (p. ex. une vitesse constante). Ces contraintes permettent d'interpoler l'état de l'objet, mais sont souvent improbables dans un environnement non contraint. D'autres travaux proposent d'analyser le contexte d'un objet totalement occulté afin de se baser sur les éléments corrélés (en terme de mouvement) du contexte. Cependant, ces algorithmes représentent des mécanismes externes à l'AS principal, appelés seulement en cas d'occultation totale (p. ex. [11]). Nous précisons ici qu'en cas d'occultation totale, notre objectif sera de récupérer le suivi dès que l'objet devient partiellement visible,

et non pas en prédire l'état pendant l'occultation totale. Cet objectif reste le même en cas d'une rotation tridimensionnelle rapide changeant complètement l'apparence de la cible et d'une sortie du champ de vision.

1.3 Contributions

D'une manière générale, un AS comporte deux composantes principales : (1) le modèle d'apparence représentant les caractéristiques de l'objet et (2) l'approche de prédiction d'état qui correspond à la procédure de recherche de la cible. Les contributions de cette thèse se rapportent à chacune de ces deux composantes. Nous proposons des solutions pour modéliser l'apparence des objets et prédire leurs états sur une séquence vidéo, tout en permettant de traiter des situations complexes de suivi sans modèle à priori. Les contributions de ce travail sont documentées dans cinq articles et peuvent être résumées comme suit.

Suivi par combinaison de caractéristiques globales et locales : Ce premier travail [12] présente un algorithme de suivi reposant sur la combinaison séquentielle de deux approches de prédiction : une approche probabiliste (filtre de particules adapté), et une approche déterministe (appariement de points caractéristiques). L'algorithme est conçu initialement pour le suivi de visage, tout en étant généralisable pour d'autres types d'objets. Les principales contributions de ce travail sont :

1. Un modèle d'apparence multi caractéristique qui inclut la distribution globale des couleurs de la cible et les points caractéristiques locales, permettant de tirer avantage des deux types de représentation tout en palliant mutuellement leurs inconvénients ;
2. Un algorithme de recherche d'objet par estimation «grossière puis fine » (en anglais *coarse-to-fine*) permettant de réduire, par une méthode probabiliste, l'espace de recherche où les caractéristiques locales sont détectées et calculées. Cette technique peut être utilisée dans d'autres AS en tant qu'étape de prétraitement, lorsqu'il importe de réduire l'espace à analyser aux régions les plus probables de l'image.

Exploitation de contraintes structurelles pour un suivi efficace des objets : Ce travail présente une méthode de suivi par points caractéristiques préservant la structure de l'objet [13, 14]. Les principales contributions sont :

1. Un nouveau modèle d'apparence enregistrant en ligne les caractéristiques locales de l'objet dans un réservoir dynamique de points caractéristiques. Le réservoir encode à la fois des propriétés structurelles anciennes et des propriétés structurelles récentes de l'objet ;

2. L'exploitation explicite de la disposition spatiale des points caractéristiques pour traiter efficacement l'occultation partielle et améliorer la précision de suivi ;
3. Une nouvelle méthode discriminative pour évaluer la qualité de suivi et déterminer si de nouvelles propriétés de la cible doivent être apprises en ligne ;
4. Un ensemble de séquences vidéos avec les réalités de terrain correspondantes (en anglais *ground-truth*) que nous avons publié en ligne. Ces scénarios concrets de suivi permettront à la communauté de chercheurs travaillant sur le suivi d'objets d'évaluer les performances des AS dans des situations spécifiques.

Suivi collaboratif par prédicteurs locaux saillants : Ce travail porte sur une nouvelle méthode de suivi par parties [15, 16]. L'AS proposé utilise des caractéristiques locales saillantes en tant que prédicteurs locaux qui collaborent pour localiser la cible. Les contributions majeures de ce travail sont :

1. Une nouvelle méthode pour évaluer la saillance des caractéristiques locales afin d'en identifier les plus fiables en se basant sur la persistance, la consistance spatiale, et le pouvoir prédictif du point caractéristique. La méthode proposée est généralisable pour d'autres algorithmes de suivi par caractéristiques locales, notamment pour les méthodes par vote ;
2. L'exploitation explicite des informations de saillance dans les différentes étapes de l'AS : prédiction locale, localisation globale, estimation du changement d'échelle et élimination des caractéristiques non persistantes ;
3. L'exploitation des informations sur l'échelle de détection et l'orientation contenues dans les descripteurs locaux pour traiter les rotations bidimensionnelles et les changements d'échelle de la cible ;
4. Une expérimentation comparative étendue évaluant l'AS face à cinq autres méthodes récentes de la littérature.

1.4 Structure de la thèse

Dans le chapitre 2, nous présentons une revue critique de la littérature en parcourant les travaux les plus pertinents sur le suivi d'objets. Le chapitre 3 survole l'ensemble des approches utilisées dans cette thèse. Le chapitre 4 présente un nouvel algorithme de suivi par estimation «grossière puis fine» de l'état et un modèle d'objet combinant des caractéristiques locales et globales dans un article intitulé *Visual face tracking : a coarse-to-fine target state estimation* publié à *International Conference on Computer and Robot Vision (CRV2013)*. Le chapitre

5 présente une méthode de suivi par caractéristiques locales préservant la structure des objets dans un article intitulé *Exploiting structural constraints for visual object tracking* soumis pour publication à la revue *Image and Vision Computing (IVC)*. Une version antérieure de ce travail a été publiée dans *IEEE Winter Conference on Applications of Computer Vision (WACV2014)* sous le titre *Structure-aware keypoint tracking for partial occlusion handling*. Dans le chapitre 6, nous introduisons une nouvelle approche de suivi d'objets basée sur des prédicteurs locaux saillants pour prédire avec précision l'état de la cible. Ce chapitre est constitué d'un article intitulé *Collaborative part-based tracking using salient local predictors* soumis à la revue *Computer Vision and Image Understanding (CVIU)*. Une première version de cet article, s'intitulant *Part-based tracking via salient collaborating features*, a également été acceptée pour publication dans *IEEE Winter Conference on Applications of Computer Vision (WACV2015)*. Des résultats expérimentaux complémentaires sont fournis dans le chapitre 7. Dans le chapitre 8, nous discutons les résultats de ce travail et le chapitre 9 conclut la thèse en résumant les contributions et en proposant des voies pour des travaux futurs.

CHAPITRE 2

REVUE DE LITTÉRATURE

Durant les dernières décennies, une multitude d'algorithmes de suivi ont été proposées dans littérature. Certaines méthodes sont dédiées au suivi d'objets spécifiques (p. ex. suivi de la tête [17, 18], de la main [19], de piétons [20, 21], de véhicules [22, 23], etc.), tandis que d'autres sont conçues d'une manière plus générique, permettant ainsi de suivre un objet sans aucune connaissance préalable de son type [24, 25, 26]. Il existe plus d'une catégorisation possible des AS dans la littérature. En 2006, Yilmaz *et al.* ont publié une revue de littérature couvrant les travaux majeurs de suivi d'objets, en catégorisant les AS selon la représentation de l'objet cible [27]. Leur classification comprend trois catégories de méthodes : le suivi par points [28, 29], le suivi par noyau [30, 31] et le suivi par silhouette [32, 33]. Récemment, d'autres auteurs (tels que [34] et [35]) distinguent deux principales approches : l'approche générative utilisant un modèle d'apparence qui décrit seulement l'objet d'intérêt, visant à en trouver la meilleure correspondance sur l'image courante [36, 37, 38] et l'approche discriminative qui décrit l'objet par rapport à l'arrière-plan, en transformant le problème de suivi en un problème de classification binaire pour distinguer la cible de l'arrière-plan [39, 40, 41].

Dans ce chapitre, nous présentons les principaux travaux connexes et inspirants de cette recherche en adoptant une catégorisation basée sur le modèle d'apparence. La catégorisation proposée tient compte à la fois des classifications antérieures et des derniers travaux sur la modélisation visuelle des objets d'intérêt. Notre revue de littérature sera axée sur deux grandes classes de modèles d'objets : les modèles holistiques décrivant les caractéristiques globales de l'objet et les modèles par parties décomposant la cible en éléments décrits par des caractéristiques locales. La figure 2.1 présente la taxinomie proposée en illustrant les sous-catégories de chaque classe de modèles. Notons que les catégorisations des modèles mentionnées ci-dessus (incluant la nôtre) concernent les algorithmes orientés cible, par opposition aux AS orientés contexte. Par suivi orienté contexte, on désigne un paradigme récent de suivi où la prédiction de l'état de la cible dépend (partiellement ou entièrement) de son contexte (p. ex. objets ayant un mouvement corrélé avec celui de la cible). Bien que les solutions proposées dans cette thèse se focalisent sur le mouvement de l'objet cible (et sont de ce fait orientés cible), certaines techniques sont inspirées par des idées issues de travaux antérieurs du suivi orienté contexte. Les principaux travaux inspirants de ce paradigme sont présentés dans la dernière section de cette revue de littérature.

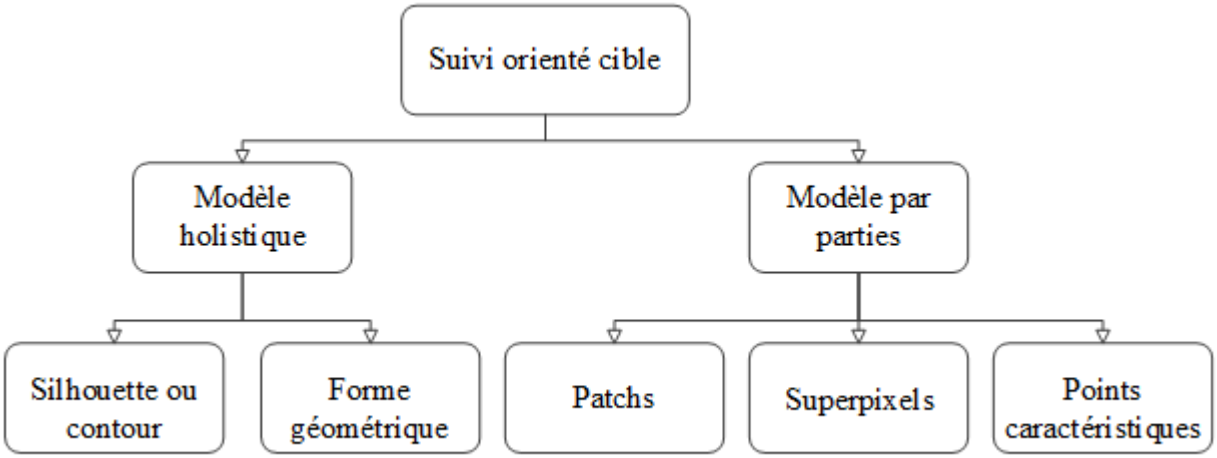


Figure 2.1 Taxinomie des approches de suivi (orienté cible).

2.1 Suivi par modèle holistique

Les modèles holistiques encodent des informations globales sur l'apparence de la cible (p. ex. la distribution globale des couleurs de l'objet, histogramme des orientations du gradient, histogramme d'arêtes, etc.). Nous répartissons les AS de cette catégorie en deux sous-catégories selon que le modèle d'apparence décrit (1) toute la région délimitée par une forme géométrique ou (2) la silhouette ou le contour de l'objet.

2.1.1 Suivi par forme géométrique

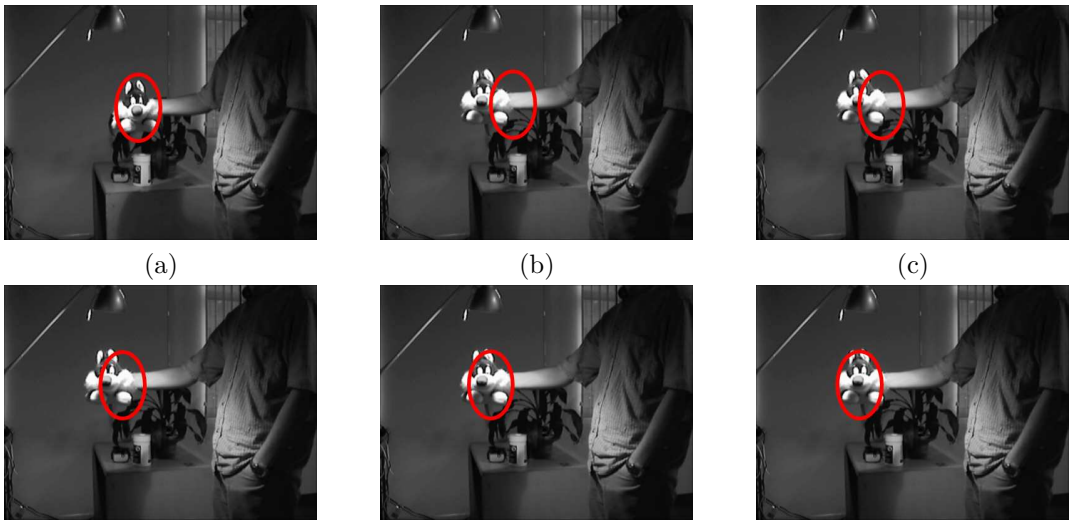
Les méthodes de cette catégorie utilisent des formes géométriques simples telles que les rectangles [42], les ellipses [43, 44] et les cercles [45], pour délimiter la région d'intérêt. L'objet est trouvé d'une trame à l'autre par le biais de modèles de changements paramétriques (translation, rotation, etc.) ou par appariement direct entre un modèle et des régions candidates. Les algorithmes de cette catégorie se différencient par plusieurs aspects, tels que le choix des caractéristiques visuelles et le modèle de mouvement. La procédure de suivi repose généralement sur l'évaluation de ressemblance entre le modèle de référence de la cible et les régions hypothétiques selon une mesure de similarité. Dans le cas d'un modèle discriminatif, les calculs de similarité impliquent des régions de l'arrière-plan et la prédiction peut utiliser une classification binaire pour distinguer la cible de son environnement. D'une manière générale, les calculs de similarité sont effectués en utilisant des propriétés statistiques des régions telles que la distribution des couleurs ou des intensités. Concernant les mesures de similarité, on retrouve des AS utilisant la corrélation croisée [46], des AS utilisant la somme des différences

au carré [47], ou encore des méthodes de suivi basées sur le coefficient de Bhattacharyya [48]. Les AS décrivant le mouvement par des modèles de changement paramétriques utilisent des méthodes statistiques pour la modélisation de mouvement et représentent la forme géométrique suivie par des paramètres tels que la position, la taille, la vélocité, etc. Lors du suivi, l'état de l'objet est estimé par un modèle dynamique de transition, mis à jour et corrigé au cours du suivi en prenant des mesures de l'image. Parmi les principales méthodes statistiques de suivi, nous citons le filtre de Kalman [49] et le filtre de particules [50]. Grâce à sa capacité à modéliser les changements aléatoires des états, le filtre de particules est une méthode largement utilisée, non seulement pour le suivi par forme géométrique [46, 51], mais aussi pour le suivi par contours [52]. Dans [46], les auteurs modélisent le mouvement de l'objet par un filtre de particules, en appliquant la mesure de corrélation croisée normalisée sur des cartes de distances (en anglais *distance transform* ou *distance map*, voir [53]) calculées pour des régions rectangulaires. Dans la méthode de filtre de particules, les états possibles de l'objet à l'instant t sont représentés par N particules $\{s_t^{(n)} : n = 1, \ldots, N\}$ générées aléatoirement. Une particule est définie par (1) les valeurs de son état courant (p. ex. la position (x, y)) et (2) le poids $\pi_t^{(n)}$ indiquant son importance. Dans le but de simplifier la sélection probabiliste des particules, un poids cumulatif $c^{(n)}$ est assigné à chaque couple $(s^{(n)}, \pi^{(n)})$ avec $c^{(N)} = 1$. Chacune des particules à l'instant t est générée à partir des états de l'instant $t-1$ selon la procédure suivante :

- Générer un nombre aléatoire $r \in [0, 1]$;
- Sélectionner la particule $s_{(t-1)}^{(j)}$, avec la plus petite valeur de j vérifiant $c_{(t-1)}^{(j)} \geq r$;
- Générer pour la particule sélectionnée $\hat{s}_t^{(n)}$ une nouvelle particule $s_t^{(n)}$ avec $s_t^{(n)} = f(\hat{s}_t^{(n)}, W_t^{(n)})$ où $W_t^{(n)}$ est une variable aléatoire de distribution normale multivariée ;
- Déterminer le poids $\pi_t^{(n)}$ de la particule $s_t^{(n)}$ en utilisant les caractéristiques mesurées à l'instant t.

Les poids sont enfin normalisés et enregistrés avec la probabilité cumulative $c_t^{(n)}$ pour former un triplet $(s_t^{(n)}, \pi_t^{(n)}, c_t^{(n)})$ pour chaque particule. À la suite de la construction de N particules, il est possible d'estimer la position de la région suivie en calculant, par exemple, la position moyenne pondérée : $\epsilon_t = \sum_{n=1}^{N} \pi_t^{(n)} s_t^{(n)}$.

La procédure Mean-Shift est une autre approche largement utilisée pour le suivi par forme géométrique et caractéristiques globales [54, 55, 56, 48, 57, 58]. D'une manière générale, la procédure Mean-Shift utilise la distribution globale des couleurs ou des niveaux de gris à l'intérieur de la forme géométrique pour représenter l'apparence de l'objet. L'algorithme maximise la similarité en apparence itérativement en comparant l'histogramme de référence Q de l'objet à l'histogramme P calculé dans une fenêtre autour d'une position hypothétique de l'objet. La similarité des histogrammes est définie en terme du coefficient de Bhattacharya,



(a) (b) (c)

(d) (e) (f)

Figure 2.2 Les itérations de suivi Mean-Shift. (2.2a) la position de l'objet estimée à l'instant $t-1$, (2.2b) la trame de l'instant t avec une estimation initiale de position correspondant à l'estimation précédente, (2.2c), (2.2d), (2.2e) mise à jour de la position estimée avec le calcul itératif du vecteur Mean-Shift, (2.2f) estimation finale de la position à la trame t. Les images sont prises de la séquence vidéo *Sylvester* [6]

$\sum_{u=1}^{b} P(u)Q(u)$, où b est le nombre de composantes de l'histogramme. Le vecteur de déplacement (appelé vecteur Mean-Shift) est calculé à chaque itération de manière à améliorer cette similarité. Comme illustré dans la figure 2.2, ce processus est répété généralement de quatre à six fois avant d'atteindre la convergence définie par un déplacement minimum.

Une autre approche commune de suivi par forme géométrique consiste à appliquer des méthodes de détection exhaustivement sur toute l'image (ou sur une sous-image). Cette approche, souvent appelée suivi par détection (en anglais *tracking-by-detection*), est utilisée dans les algorithmes discriminatifs par modèle holistique en appliquant un classifieur binaire. Ce dernier a pour but de distinguer l'objet suivi de l'arrière-plan. L'approche comprend plusieurs variantes qui se distinguent par la manière d'utiliser le résultat de classification :

- Dans [59] la classification est utilisée seule pour déterminer l'état final de la cible ;
- Dans [60] le classifieur valide la prédiction de l'AS principal et exécute une recherche exhaustive si la cible est perdue ;
- Dans la méthode TLD (*Tracking-Learning-Detection*) [40] la prédiction finale combine le résultat d'un AS avec celui du classifieur.

L'utilisation des caractéristiques globales calculées à l'intérieur d'une forme 2D simple est une technique commune offrant une représentation souvent efficace pour modéliser l'objet d'intérêt à une faible complexité. L'un des inconvénients de cette représentation est que

certaines parties de l'objet peuvent rester à l'extérieur de la forme géométrique, tandis que des pixels de l'arrière-plan interviennent dans le calcul des caractéristiques. Cet effet est dû principalement aux formes complexes des objets ne pouvant pas être décrites par des formes géométriques primitives et aux déformations que peuvent subir les objets articulés durant le suivi.

2.1.2 Suivi par silhouette

Les modèles d'apparence basés sur les contours fournissent une représentation appropriée pour les objets de forme complexe en trouvant les frontières exactes de la cible. Le suivi par silhouettes et contours vise à estimer la région de l'objet sur chaque trame en utilisant des mécanismes externes de détection ou à partir des contours générés sur les trames précédentes. Parmi les caractéristiques d'objets utilisées, on retrouve l'histogramme de couleurs calculé à l'intérieur des silhouettes [61] et les arêtes détectées [62]. Dans cette catégorie d'algorithmes, la recherche de la cible peut être effectuée par appariement de formes [61, 63, 32] ou par évolution de contour [64, 65, 66]. Dans les méthodes d'appariement de formes, la translation de la silhouette est calculée d'une trame à l'autre et le modèle de l'objet peut être réinitialisé en conséquence pour l'adapter aux changements d'apparence. Ce modèle repose généralement sur une fonction de densité (histogramme de couleurs ou d'arêtes), sur les frontières de la silhouette, ou sur une combinaison de ces caractéristiques. Dans [61], les auteurs proposent un AS par appariement de formes. Les silhouettes sont modélisées par des histogrammes d'arêtes et de couleurs, calculées à partir de cercles couvrant la silhouette de l'objet suivi, ayant différentes longueurs de rayon. Initialement, l'algorithme construit le plus petit cercle contenant la silhouette détectée. Des points de contrôle P_i sont ensuite choisis uniformément sur le cercle englobant, définissant chacun un ensemble de cercles concentriques à différentes longueurs de

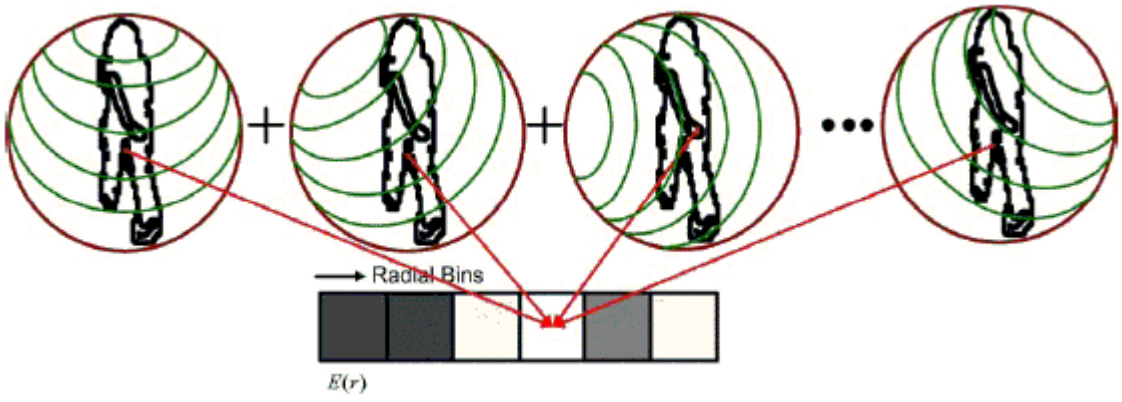


Figure 2.3 Calcul de l'histogramme d'arêtes pour la silhouette détectée [61], © 2004 IEEE.

rayon. Pour un point de contrôle donné, chaque cercle correspond à une composante de l'histogramme global de la silhouette. La figure 2.3 illustre la définition du modèle d'arêtes utilisé avec huit points de contrôle sur le cercle englobant et six cercles concentriques. La valeur de la $j^{ème}$ composante de l'histogramme d'arêtes indique la fréquence des pixels d'arêtes situés dans le $j^{ème}$ cercle selon les huit points de contrôle. La $j^{ème}$ composante de l'histogramme est calculée avec $E(j) = \sum_i E_j(P_i) / max_j(\sum_i E_j(P_i))$ où $E_j(P_i)$ est le nombre de points d'arêtes pour la $j^{ème}$ composante radiale définie par le $i^{ème}$ point de contrôle. D'autre part, un histogramme de couleurs est construit pour la silhouette en utilisant la même configuration des cercles concentriques. Les similarités entre les histogrammes d'arêtes et les histogrammes de couleurs provenant de deux trames consécutives sont basées sur le coefficient de Bhattacharya. La distance globale entre deux silhouettes combine les deux distances élémentaires en une seule mesure obtenue par l'équation : $Sim = 1/(1 + \sqrt{(Dist_{couleur})^2 + (Dist_{forme})^2})$. Ce modèle est invariant à la rotation, car une rotation de la silhouette est équivalente à la permutation des points de contrôle. D'autre part, l'invariance à l'échelle est également garantie par la normalisation dans l'équation de l'histogramme.

À la différence des méthodes par appariement fonctionnant par détection et mise en correspondance de silhouettes, les méthodes par évolution de contours résolvent le problème de suivi en évoluant des contours sur la trame courante à partir des contours de la trame précédente. L'évolution des contours peut être effectuée selon des modèles de l'espace d'états modélisant l'évolution de la forme et du mouvement du contour, tel que le filtre de particules utilisé dans [52] et le filtre de Kalman dans [67]. D'autres méthodes évoluent le contour par la minimisation de son énergie, en utilisant des techniques d'optimisation telles que la descente de gradient [68, 69].

Il est à noter que le suivi par silhouette est le choix approprié lorsque l'application de suivi requiert la région complète et exacte de l'objet. Cette approche de représentation est également adaptée aux objets déformables et de forme complexe. Cependant, qu'il s'agisse de méthodes par appariement ou par évolution de contour, le suivi par silhouette représente des limites dans les environnements non contraints. En effet, le suivi par appariement de formes utilise des mécanismes externes pour la détection des changements (p. ex. soustraction de l'arrière-plan) et n'est de ce fait pas toujours adapté aux arrière-plans dynamiques. D'autre part, l'approche par évolution de contours nécessite qu'une partie de l'objet sur la trame courante soit en recouvrement avec sa région sur la trame précédente. Cette hypothèse n'est souvent pas valide à cause d'éventuels déplacements importants de la cible ou d'un faible taux de trame de la caméra.

D'une manière générale, les modèles d'apparence holistiques présentent l'avantage de simplifier la caractérisation des objets, dès lors que les caractéristiques sont calculées d'une façon

globale. Cependant, les descripteurs globaux sont incapables de représenter efficacement les éventuels changements locaux et partiels que peuvent subir les objets. D'autre part, l'occultation partielle des objets est une autre difficulté limitant les méthodes par modèle holistique. En effet, le camouflage d'une partie de la cible risque de dégrader considérablement le modèle global d'apparence causant la dérive de l'AS. Les modèles d'apparence décomposant la cible en parties représentent une approche relativement récente. Ils reflètent les changements locaux d'apparence tout en permettant de traiter les occultations par des appariements partiels de caractéristiques. Cette catégorie d'algorithmes est présentée dans la section suivante.

2.2 Modèles par parties

Les dernières années ont connu une utilisation croissante des modèles d'apparence par parties dans des applications de détection [70, 71, 72] et de reconnaissance d'objets [73, 74]. Plusieurs travaux de suivi se sont également basés sur des modèles d'apparence par parties [75, 2, 76, 77, 78, 79, 80]. Cette tendance est expliquée principalement par la robustesse des représentations par parties face aux changements d'apparence locaux et à l'efficacité des algorithmes de recherche pour la prédiction de l'état global de la cible à partir d'un sous-ensemble des parties de l'objet.

2.2.1 Représentation par patchs

Il s'agit de la représentation la plus populaire dans la littérature. Parmi les travaux précurseurs de cette approche, on retrouve l'AS d'Adam *et al.* [2] nommé *FragTrack*. Dans ce travail, l'objet suivi est représenté par un ensemble de patchs extraits selon une grille régulière. La région rectangulaire contenant l'objet suivi est divisée en plusieurs patchs rectangulaires représentant chacun une partie de l'objet. La configuration des patchs est choisie d'une manière arbitraire et n'est, de ce fait, pas conçue pour un type d'objets en particulier. Les grilles utilisées pour la décomposition des objets comprennent un nombre total de 36 patchs tel que montré dans la figure 2.4. L'appariement des patchs entre le modèle de l'objet et une région candidate repose sur la comparaison d'histogrammes locaux d'intensités. Ces derniers sont calculés en un temps réduit grâce à l'utilisation des images intégrales [81, 82]. La position et l'échelle de la cible sont ensuite estimées en se basant sur les votes des patchs appariés.

Dans un travail ultérieur, Erdem *et al.* ont proposé une amélioration de l'algorithme en associant des degrés de fiabilité aux patchs locaux afin de les différencier [76]. Chaque patch contribue ainsi à la prédiction de l'état de la cible selon son degré de fiabilité. Cette amélioration a permis de surpasser *FragTrack* en terme précision de suivi. Toutefois, à cause de l'utilisation d'une disposition prédéfinie et figée des blocs composant l'objet, les modèles

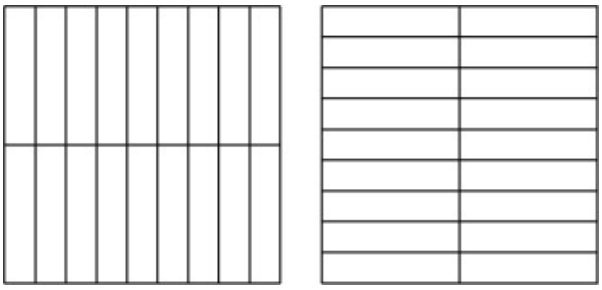


Figure 2.4 Utilisation des patchs pour décomposer la cible dans [2].

utilisant une grille régulière sont inefficaces dans le cas d'un changement important de pose dans le plan, en plus d'être inappropriées aux objets articulés. D'autres travaux proposent des modèles par patch adaptés aux objets articulés en tenant compte des déformations possibles de l'objet [80, 75, 79]. À la différence de [2] et [76], ces méthodes n'utilisent pas de gabarits prédéfinis. Les modèles utilisés décomposent l'objet en régions rectangulaires représentant la structure de la cible. Dans [80], les auteurs suivent les parties de l'objet séparément. La prédiction de l'état de la cible intègre les prédictions locales des différents blocs suivis. Cette méthode détecte les blocs produisant des mesures incohérentes afin de les éliminer dans la procédure d'intégration. Dans [75] la structure de l'objet est encodée par la disposition spatiale d'un nombre réduit de patchs (deux ou trois). Les patchs sont caractérisés par des histogrammes d'intensité calculés en utilisant des images intégrales. L'objet suivi est ensuite recherché exhaustivement sur toute l'image. Les histogrammes locaux sont appariés par la distance de Bhattacharyya et les comparaisons locales sont combinées en une seule mesure de similarité globale pour toute la fenêtre candidate englobant l'objet. Kwon et Lee proposent une approche similaire pour représenter la structure des objets non rigides par un ensemble dynamique de patchs mis à jour durant le suivi [79]. Les modèles d'apparence proposés dans ces trois travaux [80, 75, 79] garantissent une représentation appropriée aux objets articulés, en tenant compte des déformations possibles. Cependant, toutes ces méthodes présentent le désavantage de nécessiter une initialisation manuelle des patchs sur les parties de l'objet. Plus récemment, d'autres auteurs ont introduit des modèles par représentation clairsemée (en anglais *sparse representation*) initialisant automatiquement les positions des patchs locaux [83, 84]. Dans [84], des patchs de taille 16x16 pixels en chevauchement sont échantillonnés sur la boite englobant la cible avec des pas de huit pixels. Cette méthode encode la structure de la cible à travers les relations géométriques entre les patchs et inclut un module de gestion d'occultation permettant de localiser la cible en se basant sur les parties visibles de l'objet. Zhong *et al.* [83] utilisent une représentation similaire où la structure de la cible est encodée par un histogramme global basé sur les relations géométriques entre les patchs. La probabi-

lité d'une région candidate est déterminée en fusionnant (par multiplication) deux mesures : (1) le score d'un classifieur discriminatif attribué à la région candidate et (2) la valeur de similarité entre les histogrammes globaux. Il est à noter que la procédure de mise à jour du modèle d'apparence représente un inconvénient commun pour les deux méthodes. En effet, cette étape manque de flexibilité vu que l'adaptation du dictionnaire de patchs consiste à un simple ajout de nouveaux éléments, sans adapter les éléments existants aux changements locaux de propriétés. Par ailleurs, tous les modèles par patchs présentent un risque d'inclure des pixels de l'arrière-plan aux blocs échantillonnés sur les frontières de l'objet. Nous avons vu dans le cadre du suivi par représentation holistique (section 2.1) que les modèles par forme géométrique souffrent de la même limite. De la même manière que les représentations par silhouette pallient ce problème avec les modèles holistiques, les méthodes de suivi par superpixels permettent de fournir une représentation par parties tout en trouvant les frontières exactes de l'objet.

2.2.2 Représentation par superpixels

Les AS de cette catégorie utilisent des techniques de sur segmentation pour décomposer l'objet en superpixels [85, 86, 87]. Dans [87], les auteurs font appel à des mécanismes externes pour segmenter chaque trame en superpixels en combinant les résultats de deux étapes : (1) la détection des frontières de tous les objets dans l'image et (2) la décomposition de toute l'image en petites régions triangulaires. Le résultat de ce prétraitement est l'ensemble des superpixels représentant les éléments atomiques de l'image. La méthode utilise un modèle d'apparence discriminatif comportant les distributions globales des niveaux d'intensité pour l'objet et l'arrière-plan, en plus des distributions locales d'intensité des superpixels. Le suivi est effectué en appariant les superpixels entre deux trames consécutives. La mise en correspondance est établie par une méthode de programmation linéaire qui minimise la différence en intensité locale et en distance spatiale entre deux éléments appariés. Les distributions globales de couleurs sont utilisées pour évaluer la vraisemblance que le superpixel appartienne à l'objet ou à l'arrière-plan. La méthode proposée a montré de bons résultats sur des séquences vidéos de sport avec des déformations importantes de la cible. L'un des inconvénients constatés est la nécessité de sur segmenter l'image entière et mettre en correspondance tous les superpixels entre deux trames consécutives. D'autre part, le modèle d'apparence ne mémorise que les caractéristiques des superpixels de la trame courante. On risque ainsi de propager les erreurs appariement à travers toute la séquence vidéo, sans que la correction des fausses correspondances soit possible.

Ultérieurement, Wang *et al.* [85] ont utilisé l'algorithme de sur segmentation SLIC (*Simple Linear Iterative Clustering*) [88] pour décomposer l'objet en superpixels qu'ils décrivent en-

suite par leurs histogrammes locaux TSL (Teinte, Saturation, Luminosité). Leur suivi est basé sur une approche discriminative qui évalue les superpixels individuellement, dans le but de distinguer la cible de l'arrière-plan et détecter les déformations et les occultations. Vu que la sur segmentation est effectuée seulement dans une région entourant la dernière position prédite de la cible, cette méthode risque de perdre la cible dans le cas d'un grand déplacement entre deux trames consécutives ou d'un faible taux de trames. De plus l'AS nécessite une phase d'entrainement pour apprendre les caractéristiques des superpixels provenant de l'objet et de l'arrière-plan.

Le choix des éléments les plus importants et les plus informatifs est un problème essentiel dans la conception des modèles d'apparence par parties. Qu'il s'agisse d'une représentation par patchs ou par superpixels, l'objet est décomposé soit en appliquant des motifs particuliers (p. ex. grille, représentation clairsemée), soit selon des caractéristiques d'apparence qui ne tiennent pas compte de la saillance visuelle (p. ex. la sur segmentation par la couleur). Une solution intéressante pour sélectionner des éléments souvent considérés comme plus informatifs et importants que les patchs et les superpixels, consiste à détecter et suivre les points caractéristiques sur l'objet.

2.2.3 Représentation par points caractéristiques

Les méthodes par points caractéristiques (appelés aussi points d'intérêt) représentent la cible par des descripteurs locaux calculés sur une petite région autour du point considéré. Les points sont déterminés sur l'objet par des méthodes de détection telles que SIFT (*Scale-Invariant Feature Transform*) [89], FAST (*Features from Accelerated Segment Test*)[90] et SURF (*Speeded Up Robust Features*) [91]. Le suivi des points repose généralement sur la mesure de similarité entre les descripteurs locaux dans le but d'apparier les points de la trame courante avec les points de la cible détectés précédemment [92, 77, 93].

Dans [92], Yang *et al.* proposent un algorithme basé sur les points SIFT et des patchs aléatoires pour modéliser l'objet suivi. Les auteurs utilisent conjointement deux sous-modèles d'apparence : un sous-modèle de couleur/texture et un sous-modèle de structure. Le sous-modèle de couleur/texture inclut des patchs aléatoires décrits par leurs histogrammes de couleurs locaux RVB (Rouge, Vert, Bleu) et leurs descripteurs LBP (*Local Binary Patterns*). Le sous-modèle de structure consiste à un histogramme spatial global encodant la disposition des points SIFT de l'objet. Notons que cette représentation de structure est fortement inspirée du modèle par cercles concentriques de [61] discuté précédemment dans le cadre du suivi par silhouette (section 2.1.2). Afin d'évaluer une région candidate sur la trame courante, les points caractéristiques sont détectés sur cette région, puis appariés avec ceux de l'objet à la trame précédente. L'ensemble des points appariés est utilisé pour construire l'histogramme spatial

des points de l'objet. La similarité en apparence entre la région candidate et la cible est calculée en intégrant en une seule équation (1) la mesure de similarité des histogrammes spatiaux de points SIFT et (2) la similarité des caractéristiques RVB et LBP locales. L'algorithme de suivi exploite ainsi trois types différents de caractéristiques, à savoir : les distributions locales de couleur, les caractéristiques locales de texture et les propriétés géométriques globales. Cependant le sous-modèle de structure ne retient que les propriétés structurelles récentes vu que l'histogramme spatial ne considère que les points appariés avec ceux de l'objet sur la trame précédente.

Ce dernier inconvénient n'est pas rencontré dans la méthode de Guo *et al.* [77] qui mémorise les points caractéristiques extraits dans un modèle à variétés (en anglais *manifolds*) mis à jour durant le suivi. Le modèle d'apparence contient ainsi plusieurs variétés organisées dans un graphe qui représente la structure de la cible. Une variété est créée pour chaque point caractéristique de l'objet, contenant en plus du descripteur originel du point, d'autres descripteurs synthétiques simulant des variations du point de vue et de l'échelle. La cible est trouvée sur la trame courante en détectant les points caractéristiques et en les appariant avec ceux du modèle de variétés. L'AS proposé a montré une stabilité remarquable dans le suivi d'objets dynamiques. Néanmoins, ces résultats sont obtenus au détriment de calculs complexes d'homographies avec la méthode itérative RANSAC (*RANdom SAmple Consensus*).

2.2.4 Discussion

D'une manière générale, la littérature de suivi d'objets connait une utilisation croissante des représentations par parties durant les dernières années. Ceci est principalement dû à la capacité de ces modèles à encoder la structure de la cible et à tenir compte des changements locaux d'apparence. De plus, les modèles par parties sont adaptés aux situations d'occultation, vu que l'état global de la cible peut être déduit à partir des états locaux des parties visibles de l'objet.

Dans ce travail, nous soutenons que les points caractéristiques offrent une modélisation de structure plus stable et plus robuste que les modèles par patchs ou par superpixels. Cette robustesse découle de la répétabilité des points d'intérêt et leur invariance face aux différents facteurs perturbateurs de suivi, tels que les variations des conditions d'illumination, le changement d'échelle et le changement de pose. À la différence de [92] où des caractéristiques locales sont extraites aléatoirement et [85] où elles sont extraites dans une région d'une taille fixe autour de la dernière position de l'objet, nous utilisons une méthode probabiliste et un modèle par forme géométrique pour réduire la région de détection des points caractéristiques en nous basant sur la distribution globale des couleurs de la cible. Cette étape permet de réduire l'espace de recherche aux zones les plus probables et éviter ainsi de détecter les points

sur toute l'image. La structure de l'objet suivi est ensuite encodée par la disposition spatiale des points caractéristiques. Plutôt que de se limiter aux propriétés structurelles récentes comme dans [92], notre représentation de l'objet inclut à la fois des propriétés récentes et des propriétés anciennes dans un modèle d'apparence dynamique mis à jour durant le suivi.

Avec les modèles d'objets par parties, la procédure de recherche typique consiste à apparier les caractéristiques locales. Si l'appariement implique itérativement deux trames consécutives (comme dans [87] et [92]), un faux appariement propagera l'erreur sur toutes les trames restantes. Dans notre méthode, l'appariement est réalisé entre la trame courante et le modèle de référence de l'objet. Bien qu'un faux appariement puisse se produire au niveau d'un point caractéristique, l'erreur n'affecte pas les appariements subséquents de ce point sur les trames suivantes. Dans notre modèle, les points caractéristiques et leurs propriétés structurelles sont appris en ligne. Tandis que certains AS mettent à jour le modèle d'apparence par un simple ajout de nouveaux éléments [83, 84], nous avons conçu une procédure d'adaptation flexible qui permet l'ajout de nouvelles caractéristiques, la modification des caractéristiques existantes et l'élimination des caractéristiques expirées. Pour décider de la suppression d'un élément local, Wang *et al.* [85] ont utilisé une fenêtre temporelle glissante conservant les caractéristiques locales pour une durée prédéterminée. Dans notre méthode, la durée de conservation dépend seulement de la propriété de persistance de l'élément, ce qui permet de conserver les points caractéristiques qui sont anciens et utiles pour le suivi.

Pour prédire la position de la cible, l'approche proposée utilise un mécanisme de vote où chaque point caractéristique de l'objet exprime une contrainte structurelle. Les votes effectués d'une façon individuelle permettent de préserver la structure de la cible en gérant l'occultation, sans l'emploi d'une structure de graphe complexe et d'un calcul d'homographie tel que dans [77]. Les idées proposées pour la modélisation de la structure et la prédiction de l'état de l'objet sont inspirées de travaux récents relevant d'un nouveau paradigme de suivi d'objet, dit orienté contexte.

2.3 Suivi orienté contexte

Dans les approches classiques de suivi, l'environnement de la cible est considéré comme un agent déstabilisateur. Par conséquent, d'importants efforts ont été déployés pour mettre au point des AS permettant de séparer et distinguer la cible du reste de la scène. Cette séparation peut être réalisée implicitement par un algorithme génératif, ou explicitement avec une méthode de suivi discriminative [1]. Cependant, l'environnement peut être intéressant dans le sens où il peut contenir des éléments corrélés avec la cible. Si, par exemple, il s'agit de

1. Pour un plus de détails sur les deux approches, le lecteur est référé au premier paragraphe de ce chapitre.

suivre un visage dans une foule, il est généralement difficile de concevoir un modèle génératif permettant de trouver le visage suivi, ou un modèle discriminatif qui permet de distinguer le visage d'intérêt parmi les autres. Si la personne porte un chandail ou chapeau d'une couleur particulière, le suivi de ce dernier pourrait aider à localiser la cible principale. De la même façon, si un autre visage accompagne la cible dans son déplacement, il serait également intéressant de considérer la relation géométrique entre les deux visages. Il s'agit des principales idées ayant motivé les travaux sur le suivi orienté contexte.

Les auteurs de ces travaux affirment qu'un objet est souvent dépendant de son contexte. Ainsi, il existerait toujours des éléments ou des objets dont les mouvements sont corrélés avec ceux de la cible (voir la figure 2.5). Selon ce principe, les auteurs dans [95] proposent d'utiliser un «compagnon» pour améliorer le suivi. Ce dernier correspond à une région avoisinant la cible en ayant un mouvement similaire. D'une manière semblable, Yang *et al.* [94] utilisent plusieurs «objets auxiliaires» assistant l'AS. De tels objets doivent avoir deux propriétés essentielles, vérifiées au moins pour un court intervalle de temps, à savoir :

– une cooccurrence fréquente avec la cible ;
– des mouvements corrélés avec ceux de la cible.

Dans [96], les auteurs considèrent les relations géométriques entre la cible et des objets similaires en apparence (distracteurs). L'AS suit ces objets simultanément avec la cible pour éviter de confondre cette dernière avec les distracteurs. En 2010, Grabner *et al.* [11] proposent une approche plus générale pour le suivi orienté contexte. Ils introduisent la notion de «supporteurs» définis comme étant des «caractéristiques utiles pour la prédiction de la position de l'objet cible» [11]. Dans leur AS, les «supporteurs» sont des points caractéristiques détectés sur toute l'image. Le déplacement de ces points est statistiquement lié au mouvement de la cible, sans qu'ils appartiennent à cette dernière. Les «supporteurs» sont initialisés puis mis à jour à chaque trame en se basant sur le résultat de la méthode KLT (Kanade-Lucas-Tomasi)

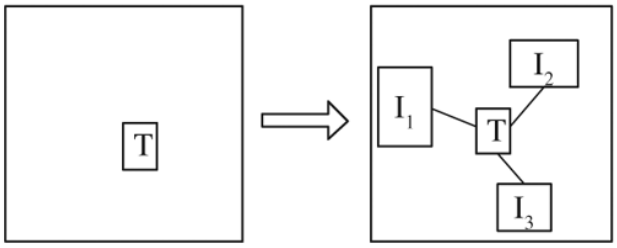


Figure 2.5 Illustration du suivi orienté contexte. T est l'objet suivi et les I_k sont des éléments du contexte de la cible. Les AS orientés cible se focalisent seulement sur l'objet suivi, tandis que les AS orientés contexte considèrent les relations géométriques entre la cible et des éléments corrélés de la scène. Figure de [94], © 2009 IEEE.

[97]. Le suivi est ainsi réalisé par KLT en tant qu' AS principal, en appliquant le modèle de «supporteurs» à chaque fois que l'objet est entièrement occulté. Dans un travail ultérieur, Dinh *et al.* utilisent aussi les «supporteurs» en gérant explicitement les situations où des distracteurs (régions d'apparence similaire à la cible) apparaissent [98].

Les méthodes orientées contexte étendent le modèle d'apparence de la cible en exploitant les informations provenant de toute la scène. Dans plusieurs cas, les relations géométriques entre l'objet et son environnement permettent de résoudre des problèmes difficiles de suivi, tels que l'occultation totale dans [11]. Toutefois, la recherche de corrélation entre l'objet et son entourage demeure une tâche couteuse qui nécessite l'analyse de tous les éléments de la scène. De plus, l'approche orientée contexte ne peut être appliquée lorsque aucune corrélation n'est trouvée. Étant donné leur spécificité à des conditions particulières, les techniques orientées contexte ne sont pas généralisables sur de larges gammes de problèmes de suivi. Ces méthodes ne représentent de ce fait que des mécanismes d'appui et de vérification, ou des procédures de secours appelées par l'algorithme principal dans des situations spécifiques.

D'une manière similaire à l'utilisation de la structure géométrique de la scène dans les méthodes orientées contexte, nous exploitons la disposition spatiale des points caractéristiques de l'objet suivi. Bien que notre modélisation des propriétés structurelles soit inspirée du paradigme orienté contexte, nos idées et motivations diffèrent considérablement de celles décrites ci-dessus. Notre modèle structurel encode ainsi les relations géométriques entre les caractéristiques locales de l'objet suivi et ne considère pas les relations entre les éléments du contexte. Le modèle d'objet proposé permet d'accomplir un suivi précis en gérant efficacement plusieurs problèmes de suivi discutés dans le chapitre 1.

CHAPITRE 3

DÉMARCHE GÉNÉRALE ET SURVOL DES APPROCHES

Cette thèse suit un format par articles pour englober les travaux effectués dans le cadre de mon projet de doctorat. Les solutions proposées aux problèmes discutés dans le chapitre 1 sont présentées dans les trois chapitres suivants. Chacun des trois chapitres correspond à un article publié ou soumis pour évaluation et peut ainsi être lu d'une façon indépendante. Le lecteur retrouvera la cohérence de l'ensemble de la recherche dans la complémentarité de ces travaux et dans l'incrémentalité des idées proposées pour traiter les problèmes de suivi.

Le premier article propose une méthode de suivi de visage généralisable pour d'autres types d'objets. Notre modélisation de l'objet cible utilise la distribution globale des couleurs RVB en tant que modèle holistique et un modèle par parties constitué des descripteurs de points caractéristiques SIFT. Sur chaque trame traitée, l'algorithme de suivi emploie séquentiellement deux types d'approches pour localiser la cible. Dans une première étape, une estimation grossière de l'état de l'objet est obtenue par une méthode probabiliste qui maximise la similarité, en terme de distribution de couleurs, avec le modèle de la cible. Nous appliquons à cet effet un filtre de particules où chaque échantillon est une région candidate de l'objet délimitée par une forme géométrique. Le résultat de cette étape est un espace de recherche réduit où les caractéristiques locales seront extraites. La prédiction finale de l'état de la cible est obtenue en appariant les points caractéristiques détectés sur l'espace de recherche réduit avec ceux du modèle de référence de l'objet. L'AS proposé tire ainsi avantage du modèle holistique (distribution globale de couleurs) pour simplifier le suivi par parties (points caractéristiques). Cette méthode s'appuie sur l'invariance et la distinctivité des points SIFT pour faire face aux problèmes de changements de luminosité, aux changements d'échelle et aux distracteurs. D'autre part, l'appariement partiel des points caractéristiques permet de localiser la cible même si une partie de cette dernière est occultée.

Remarquons que le modèle par parties proposé dans le premier article ne tient pas compte de la disposition spatiale des points. En effet, la recherche de la cible est basée uniquement sur les scores d'appariement des régions candidates sans tenir compte de la structure géométrique des points caractéristiques, résultant en une moins bonne précision de localisation (notamment lors de l'occultation partielle). Dans le deuxième article, nous présentons un nouvel algorithme de suivi nommé SAT (*Structure-Aware Tracker*). La méthode SAT exploite les contraintes structurelles encodées par les points caractéristiques en s'inspirant d'idées issues du suivi orienté contexte (voir la section 2.3). Dans cet article, nous démontrons que

les points caractéristiques sont des éléments stables permettant d'encoder efficacement la structure interne de l'objet. L'utilisation des propriétés structurelles des points SIFT peut améliorer considérablement la précision de suivi, notamment dans les situations d'occultation partielle. Afin de limiter la détection des points caractéristiques aux régions les plus probables de l'image, SAT utilise la méthode probabiliste d'estimation grossière présentée dans le premier article [12]. Les points SIFT sont ensuite détectés dans l'espace de recherche réduit pour être appariés avec le modèle de la cible. Pour déterminer la position précise de l'objet, chaque point caractéristique identifié comme appartenant à la cible exprime une contrainte structurelle en votant pour la position de l'objet suivi. Notons qu'à la différence de la méthode du premier article, où le modèle par parties est constitué uniquement des points caractéristiques extraits sur la dernière trame, SAT utilise un réservoir de caractéristiques qui enregistre les propriétés structurelles de l'objet capturées dès le début du suivi. Un point caractéristique (et les propriétés qui y sont liées) n'est éliminé du réservoir que s'il ne satisfait plus une exigence minimale de persistance.

Grâce à la préservation des propriétés structurelles de l'objet et au dynamisme de son modèle d'apparence, SAT effectue un suivi stable en traitant efficacement l'occultation partielle. Toutefois, l'algorithme proposé ne comprend pas une estimation du changement d'échelle et présente une sensibilité aux rotations importantes dans le plan. Le troisième article introduit une nouvelle méthode de suivi par parties nommée SCFT (*Salient Collaborating Features Tracker*) qui exploite les informations sur l'échelle de détection et sur l'orientation principale contenues dans les descripteurs locaux SIFT pour tenir compte de toutes les transformations locales. La méthode proposée utilise les propriétés structurelles des points caractéristiques d'une manière similaire à SAT, mais en différenciant les caractéristiques les plus fiables de l'objet des autres caractéristiques aberrantes et des mauvais prédicteurs. Dans le but de distinguer les points caractéristiques les plus fiables, nous définissons la notion de saillance d'un point d'intérêt comprenant trois facteurs : la persistance, la consistance spatiale et le pouvoir prédictif. Ces facteurs sont évalués itérativement pour tous les points caractéristiques lors de la mise à jour du modèle d'apparence. Les informations de saillance sont ensuite utilisées dans toutes les étapes de l'AS pour prédire la position de la cible et estimer le changement d'échelle. Une évaluation expérimentale étendue montre que SCFT surpasse cinq méthodes récentes de la littérature dans des scénarios difficiles de suivi.

CHAPITRE 4

ARTICLE 1 : VISUAL FACE TRACKING : A COARSE-TO-FINE TARGET STATE ESTIMATION [1]

Abstract

Keypoint-based methods are used in visual tracking applications. These methods often model the target as a collection of keypoint descriptors. Target localization on subsequent frames is thus a complex task that involves detecting keypoints, computing descriptors, matching features, and checking match consistency to update the reference model adequately and avoid tracker drifts. This work aims to boost keypoint tracking efficiency while reducing complexity by a coarse-to-fine state estimation to track human faces. In this context, we present a novel face tracking algorithm combining color distribution and keypoints to model the target. Our tracking strategy is based on a color model to predict a coarse state where the target search should be performed using keypoints. The fine estimation of the target state is then made by matching candidate keypoints with those of a reference appearance model that evolves during the tracking procedure. Qualitative and quantitative evaluations conducted on a number of challenging video clips demonstrate the validity of the proposed method and its competitiveness with state of the art trackers.

4.1 Introduction

Detecting, tracking and recognizing individuals are key components in automated video surveillance systems. Despite great progress in automated visual tracking, person tracking remains a challenging problem due to numerous real life difficult situations, such as sophisticated object shape, complex motion, and appearance changes caused by pose, illumination, occlusion, etc. Finding the appropriate appearance model is a key problem that attracted much attention in recent years.

In this work, we focus on the problem of tracking a human face with no prior knowledge other than its state in the first video frame. This tracker will be used in a face recognition application were a PTZ camera follows a face until enough information are extracted from it to allow person identification. The designed system should be able to track arbitrary movements

1. W. Bouachir and G.-A. Bilodeau, "Visual face tracking : A coarse-to-fine target state estimation," publié dans la international conference on Computer and Robot Vision (CRV), pp. 45–51, 2013.

of a human face, under different scales, with a variable background (due to camera motion), and under changing illumination conditions. In addition to these requirements, the proposed algorithm should robustly handle partial occlusions. We address the problem of finding an appearance model robust to partial occlusion and finding an efficient target search strategy. Since our algorithm is designed to track human faces, the target is represented by a region using color features as a global descriptor for coarse localization of the target position, in addition to keypoint descriptors for fine localization and occlusion handling. Both appearance models reinforce each other for more robust and more accurate state estimation. It has been shown that an adaptive appearance model, evolving during the tracking procedure is the key to good performance [99, 6]. To ensure model adaptation to the target appearance changes, our appearance model is updated during tracking, under certain conditions to avoid a too large drift.

The contributions of this paper are: 1) the appearance model of the target, and 2) the search strategy for predicting the target location in the current frame. In our search strategy, kernel tracking and point tracking are used in conjunction in order to perform a robust prediction. Note that in our work, and in accordance with the definition in [27], kernel tracking refers to the target representation (not to the iterative localization procedure *mean-shift*). For example, the kernel can be a rectangular or a circular shape with the associated color distribution. In the first step, our algorithm uses a particle filter to track a kernel for finding image region candidates where the keypoint target search should be performed. The fine estimation of the target state is based on keypoint descriptors computed in the region candidates and matched with those of the target model. The advantage of reducing the keypoint search space with a coarse state estimation is twofold: i) by reducing the search space, we reduce the number of possible false keypoint matches and simplify the keypoint matching process, and ii) considering a smaller search space in the image reduces the number of keypoints to compute. Indeed, in our method we limit the search area on the current frame to the overlapping region defined by the best particles as selected from the kernel representation.

4.2 Related Work

Object tracking methods can be divided into three categories according to their appearance model [27]: point tracking, kernel tracking, and silhouette tracking. Silhouette tracking methods use the information encoded inside the object region which is estimated in each frame by either shape matching or contour evolution[61, 100]. The most significant advantage of silhouette tracking methods is their flexibility to handle various object shapes by

extracting the complete object region. Among their important issues, is their capability to address the occlusion problem explicitly [27]. Moreover, contour tracking algorithms require that a part of the object in the current frame overlaps with the object region in the previous frame.

In kernel tracking methods, a kernel of different shape is used depending on the target (e.g. a rectangular template to track the complete human body, circular shape for face tracking, etc.)[47, 101, 41]. Targets are tracked by computing the motion of the kernel in subsequent frames in the form of a parametric transformation, such as translation and rotation. The use of geometric shapes to represent objects is very common due to computational efficiency. One of the limitations of kernel methods is that parts of the objects may lie outside the kernel, while parts of the background may lie inside it. This makes model updating more challenging as including background pixels in the model results in tracker drift.

Point trackers represent targets by points and the association of the points across consecutive frames is based on previous objects states that can include point descriptors and locations [102, 103, 93]. They are naturally suited to handle occlusions as partial matches between points are sufficient for most tracking scenarios. Recent point tracking methods model an object as a set of keypoints detected by an external mechanism (i.e. a keypoints detector) [103, 93]. Once the keypoints are detected in a video frame, and their descriptors are computed, the object localisation can be achieved according to two possible approaches: classification in the case of a discriminative algorithm, and matching in the case of a generative tracker. Matching approaches store keypoint descriptors in a database. The descriptors are designed to be invariant to various perturbation factors (noise, scale, rotation, illumination, etc.) and can be matched with those of the target model in a nearest-neighbour fashion. Classification approaches are used in discriminative algorithms and consider matching as a binary classification problem: each keypoint is classified as a keypoint from the background, or a keypoint from the target model. The initial classifier needs to be learned offline, considering the background and the object observed under various transformations.

Representing a target by a set of keypoints enforces invariance against rotation, scale changes, changes in viewpoint and robustness to partial occlusions [104]. However, detecting keypoints on large image regions, computing the descriptors and matching them is quite costly. On the other hand, the keypoint classification approach requires a prior knowledge of the object appearance and a training stage.

4.3 Tracking method

We propose a novel generative tracking algorithm based on a combination of global and local features of the target, where the search strategy for finding the target combines kernel tracking and point tracking techniques.

4.3.1 Motivation

By using a geometric shape to contain the target, and global features for modeling, kernel trackers perform well while maintaining a low complexity. Nevertheless, this approach is not designed to handle occlusions, unless representing the target by multiple fragments to be matched. Keypoint methods can handle this problem by establishing partial correspondences, but may corrupt the target model in case of mismatches or tracker drift. In this context, the proposed method includes two steps that take advantage of both approaches, while mutually reducing their drawbacks. In more concrete terms, kernel tracking is firstly applied to provide a coarse localisation of the target. Keypoints are then used to improve the prediction by finding a final more accurate position, and by adding distinctiveness and robustness against occlusions. Moreover, we consider that their reliability is confirmed by the kernel tracker since they are located on candidate regions determined in the first step. This assumption offers several advantages:

- keypoints model adaptation becomes easier;
- drift and mismatches are considerably reduced;
- matching keypoints can be performed in a simple way, as the matched keypoints should be consistent with the target model, and no further spatial consistency should be verified.

Details of the different system components and algorithmic steps are presented in the following sections.

4.3.2 Appearance model

We delimit the target using a circular area that contains the tracked face in every video frame. As we will show in the experiments, the method presented here is not limited to faces and can be adapted, or even directly applied to track other types of arbitrary moving objects. The proposed target model describes the image region delimited by the circle that circumscribes the face. This model includes two types of features: 1) the RGB color probability distribution represented by a quantized 3D histogram, and 2) a set of keypoint descriptors computed within the face region. By constructing an m-bin histogram $\hat{\mathbf{q}} = \{\hat{q}\}_{u=1\ldots m}$, with $\sum_{u=1}^{m} \hat{q}_u = 1$, some parts of the background may lie inside the circular kernel. As discussed

in [105], these pixels will affect the color distribution and cause tracking drift. To reduce the effect of these pixels in the distribution calculation, we use a kernel function $k(x)$ that assigns smaller weights to pixels farther from the kernel center. On the other hand, $\hat{\mathbf{q}}$ is normalized to ensure scale invariance. More formally, the RGB histogram is computed for the n pixels inside the circular region according to the equation:

$$\hat{q}_u = \frac{1}{\sum_{i=1}^{n} k(d_i)} \sum_{i=1}^{n} k(d_i)\delta[c_i - u] \tag{4.1}$$

where $d_i \in [0, 1]$ is the normalized distance from the pixel x_i to the kernel center, c_i is the bin index for x_i in the quantized space, δ is the Kronecker delta function, and $k(d_i)$ is the tricube kernel profile defined by:

$$k(d_i) = \frac{70}{81}(1 - d_i^3)^3. \tag{4.2}$$

In this way, the proposed color model is suited to our application requirements. Indeed, the color histogram is: i) quantized to be quite general, reduce noise and light sensitivity and reduce the computation complexity, ii) normalized to enforce scale invariance, and iii) weighted to reduce the effect of the background pixels in the target model, and therefore reduce model drift.

The proposed system should be able to handle many difficult scenarios, such as occlusions and the presence of background regions with colors similar to those of the target. In addition, it has been shown that even for individuals of different races, the skin color distributions are very similar [106]. To ensure a more robust and distinctive feature set, the target reference model also includes SIFT keypoints [89] detected and described in the target region. Our method is not specific to SIFT. Even faster keypoint detector/descriptor combination may be used, although SIFT remains one of the most reliable method under various image transformations [107]. Including keypoints to the target model increases the distinctiveness of the tracking algorithm to distinguish the tracked person from the other individuals who may enter the field of view and ensures robustness against changes in lighting conditions and scale. Moreover, the tracking algorithm will be able to handle efficiently partial occlusions due to the locality propriety of SIFT keypoints that allows partial match.

4.3.3 Coarse target state estimation

To localize the target in the current frame, the search is guided by a probabilistic particle filtering approach [52], where each particle is a circular region characterized by its RGB color distribution as described in section 4.3.2. More specifically, the possible target states in frame t are represented by N randomly generated particles $\{s_t^{(n)} : n = 1, ..., N\}$. Each particle is

defined by:

- the current state values: position (x, y) and radius r;
- the weight $\pi_t^{(n)}$ that reflects the importance of the particle.

To reduce the computational cost, we assign a cumulative weight $c^{(n)}$ to each pair $(s^{(n)}, \pi^{(n)})$ where $c^{(N)} = 1$. When processing a frame t, the new particles are generated from the states of $t-1$ according to the following procedure:

1. generate a random number $R \in [0, 1]$;
2. find the particle $s_{t-1}^{(j)}$ with the smallest value of j verifying $c_{t-1}^{(j)} \geq R$;
3. generate for the selected particle $\hat{s}_t^{(n)}$, a new particle $s_t^{(n)}$ with $s_t^{(n)} = f(\hat{s}_t^{(n)}, W_t^{(n)})$, where $W_t^{(n)}$ represents the error;
4. determine the weight $\pi_t^{(n)}$ for the particle $s_t^{(n)}$ by comparing its RGB color distribution estimated at frame t to the reference model color distribution.

To evaluate the similarity between the reference color model $\hat{\mathbf{q}}$ and the color distribution $\hat{p}_t^{(n)}$ of a generated particle $s_t^{(n)}$, we define the distance between the two distributions as:

$$d(\hat{\mathbf{q}}, \hat{p}_t^{(n)}) = \sqrt{1 - \rho[\hat{q}, \hat{p}_t^{(n)}]} \tag{4.3}$$

where

$$\rho[\hat{q}, \hat{p}_t^{(n)}] = \sum_{u=1}^{m} \sqrt{\hat{q}_u . \hat{p}_{u,t}^{(n)}} \tag{4.4}$$

is the Bhattacharyya coefficient between $\hat{\mathbf{q}}$ and $\hat{p}_t^{(n)}$. The particles weights are finally normalized and saved with the cumulative probability $c_t^{(n)}$ to form a triplet $(s_t^{(n)}, \pi_t^{(n)}, c_t^{(n)})$ for each particle. The area covered by the best particles in the image (i.e. the particles having the highest weights) represents a coarse estimation of the target state, and thus constitutes a reduced search space where keypoints will be detected and matched. The keypoints are detected and described in the overlapping region defined by the best particles as explained in the following section.

4.3.4 Target prediction and model adaptation

4.3.4.1 Fine state estimation

The second step of the tracking procedure relies on keypoints. These keypoints are the centers of salient patches located on the target face region. In our work, we use SIFT as keypoints detector and descriptor. SIFT features are reasonably invariant to changes in illumination, image noise, rotation, scaling, and changes in viewpoint between two consecutive frames [104, 107, 108]. SIFT keypoints are detected and described on the target region to be

included to the initial appearance model. For each subsequent frame, keypoints matching will consider only the image region delimited by the best particles. SIFT keypoints are detected on the overlapping region defined by the N^* best particles. The 128 element descriptors are then computed for each keypoint to summarize the local gradient information.

By reducing the search region to the most important candidate particles, we avoid detecting the keypoints, computing the local descriptors and matching them on the entire image. The descriptors of the overlapping region are then matched with those of the target model based on the Euclidian distance. To do so, we construct a first set of candidate matches by selecting for each keypoint of the target model, the most similar keypoint detected in the search region in the current frame. In [89], it has been shown that the probability that a match is correct can be determined by evaluating the ratio of distance from the closest neighbour to the distance of the second closest. For our algorithm, we keep only the matches in which the distance ratio from the closest neighbour to the distance of the second closest is less than 0.7. Given the final set of matched pairs and their locations, we use a mapping table that indicates the particles where each keypoint in the search region is located to compute N^* matching scores. Finally, the predicted target location is given by the particle having the highest matching score.

4.3.4.2 Appearance model adaptation

After the target model is built, tracking is made in subsequent frames using the procedure described above. To adapt the reference model to the latest appearance of the target, the color distribution and the keypoint descriptors set are updated every time a good prediction is achieved. Our definition of a good prediction is that 50% of the SIFT descriptors in the model are matched with the keypoints of a candidate particle. This avoids too large drifts of the model. Therefore, the reference color model is adapted every time we hypothesize that the prediction is relatively precise. For this, the color distribution of the predicted target $\hat{q}_{new}$ is computed, and the adaptation is made according to the equation:

$$\hat{q} = (1 - \alpha)\hat{q}_{old} + \alpha\hat{q}_{new}. \tag{4.5}$$

The learning factor α is determined automatically based on the quality of the new prediction and is defined as $\alpha = 0.5M$ with $M \in [0.5, 1]$ being the matching rate of the keypoint descriptors in the target model. Furthermore, we update the target keypoint model to include the newly detected features.

4.4 Experiments

To evaluate the effectiveness of the proposed tracking algorithm, we performed two series of tests using two different datasets. The first set of video sequences was captured in our laboratory to evaluate the quality of the proposed algorithm when tracking faces in different scenarios. To compare with the state of the art results and evaluate quantitatively the tracking performance in a more general context with different types of objects, we also tested our method on a publicly available dataset used in the latest visual tracking works.

4.4.1 Qualitative evaluation

The first dataset includes seven scenarios. Seven video sequences were captured using one IP PTZ camera in a laboratory room. The room was cluttered with desks, chairs, and technical video equipment of various shapes and textures in the background. Lighting was uneven in the room. The Sony SNC-RZ50N camera used for capture was mounted on a tripod at a height of approximately 1.8 meters. The video frames are 320x240 pixels and were sent via IP network, at a frame rate of 15 fps, to a 3.4 GHZ Core i7-3770 CPU on which the processing is done.

Figure 4.1 shows tracking results on a few key frames for different scenarios. Each row in this figure corresponds to a video sequence where the selected video frames are numbered. The goal of scenario 1 (331-frame video) is to track the face of a person moving randomly in the room. The walking speed is varying abruptly and the person changes direction and orientation, with distances to the camera varying from 2 to 6m. After manually initializing the system, the face was successfully tracked in practically all the processed frames. We observed a decrease of tracking precision when the subject changes direction or turns its face quickly (frame 69). Nevertheless, the adaptive target model is robust enough to quickly recover a stable track after few frames (frame 80).

In scenario 2 (428-frame video), we test the robustness of our algorithm in the case of two moving persons. Here we track a subject that can cross in front or behind another walking or immobile person. As shown in the second row of figure 4.1, the algorithm can keep correct track of the target face, even though there are partial occlusions by another face. In the case where the target person's face is completely occluded by another face (no keypoint match), the system detects a total occlusion, and thus avoids tracking the occluding person's face, and therefore does not erroneously update the face model. This is mainly due to the target model keypoints that does not match with those of the occluding face.

In scenarios 3 (339-frame video) and 4 (357-frame video), we evaluate the tracking quality in case of change in pose and orientation, and severe changes of viewpoint. Although

these changes can co-occur with fast lateral movements of the target, the tracking results of sequences 3 and 4 show that our tracker can handle such situations very well.

The last scenarios (5-7) test the ability of the system to handle different types of partial occlusion. In scenario 5 (321-frame video), the subject tries to hide behind a structure of the background. During the partial occlusion, the target continues moving laterally and the tracker successfully predicts its position in all the frames. In scenario 6 (304-frame video), the face is occluded by the person's hand, while in scenario 7 (532-frame video) a book is used to partially hide the target from different sides. In both cases, the tracker continues to correctly predict the target position without drifting, even when the face is severely occluded. This observation highlights the advantage of using a keypoint-based model.

4.4.2 Quantitative comparison with state of the art methods

While our proposed tracker was designed specifically to track human faces, a quantitative comparison with several state of the art trackers is presented in this section. We show that it can achieve very competitive results when tracking different types of objects. We tested our tracking system on challenging video sequences used in the latest works in visual tracking [3, 109, 38]. We also used publicly available ground truth data and experimental results provided by [3]. The performance comparison is made with two versions of the Online AdaBoost algorithm (OAB) presented in [110]. The first version (OAB-1) generates one positive example per frame, while the second version (OAB-45) generates 45 image patches comprising one positive bag. To further evaluate our tracker performance, we also compare our results with three other algorithms: the SemiBoost tracker [39], FragTrack [2], and MILTrack [3]. Note that the experimental results for the compared methods are obtained by using the default parameters provided by the authors. In order to quantify several results, we consider the two metrics used in [3]:

- the precision at a fixed distance threshold, which is the percentage of frames where the tracker is within a certain distance of the ground truth;
- the average center location error of the tracker.

Since the proposed tracker is non-deterministic, the results presented in tables 4.1 and 4.2 are the averages over 5 runs. Also note that these video sequences are available in gray scale only. Thus, we adapted our algorithm to use intensity distributions instead of RGB color distribution.

The sequence *Sylvester* shows a moving stuffed animal undergoing pose variations, lighting changes, and scale variations. For this sequence, the OAB-45 tracker fails with only 11% of precision. The other trackers, including our, are able to track the target with a high accuracy. MILTrack has the best results while our tracker achieved the second best performance in terms

Table 4.1 Tracker precisions at a fixed threshold of 30: percentage of frames where the center of the predicted location is within 30 pixels of the ground truth. **Bold red** font indicates best results, *blue italics* font indicates second best.

Video sequence	OAB-1	OAB-45	Semi-Boost	Frag-Track	MIL-Track	Ours
Sylvester	0.71	0.11	0.81	0.87	**0.98**	*0.90*
David indoor	0.36	0.18	0.51	0.50	*0.70*	**0.80**
Occluded face	0.32	0.04	*0.97*	**1**	0.71	0.76
Tiger 1	0.61	0.38	0.46	0.36	*0.89*	**0.93**
Average	0.50	0.18	0.69	0.68	*0.82*	**0.85**

Table 4.2 The average tracking errors: the error is measured using the Euclidian distance from the center of the predicted location to the center of ground truth. **Bold red** font indicates best results, *blue italics* font indicates second best.

Video Sequence	OAB-1	OAB-45	Semi-Boost	Frag-Track	MIL-Track	Ours
Sylvester	25	79	16	**11**	**11**	*14*
David indoor	49	72	39	46	**23**	*26*
Occluded face	43	105	*7*	**6**	27	20
Tiger 1	*35*	57	42	39	**16**	**16**
Average	38	78.25	26	25.50	*19.25*	**19**

of precision and average error.

The sequence *David indoor* shows the robustness of our algorithm when tracking a human face under severe camera motion, background and illumination changes, and large scale variations. Our tracker achieved the best precision of 80% and an average error of 26 pixels.

The results of *Occluded face* sequence show that FragTrack outperforms all the other methods because it is specifically designed to handle occlusions via a part-based model. Our tracker is also designed to handle occlusion, but we did not use yet the position of the keypoints in a particle to improve object localization. Our tracker achieved a good result, outperforming MILTrack, OAB-1, and OAB-45, with a 76% precision.

The sequence *tiger 1* exhibit many challenges, showing a stuffed tiger in many different poses, with frequent occlusion level, fast motion and rotations causing motion blur. With this sequence, our algorithm outperforms significantly the others in precision, having also the best average error. Figure 4.2 presents a few screenshots of tracking results. In general, the proposed tracker performed well for all the sequences. The results of tables 4.1 and 4.2 show that every tracker fails in at least one sequence. Nevertheless, our method outperformed all the other algorithms when averaging the precision and error results over all the experiments.

4.5 Conclusion

We developed a novel face tracking method that learns and updates the target model during the tracking procedure. It is based on coarse-to-fine state estimation that combines kernel and keypoint tracking. Our experiments show the robustness of our algorithm and its competitiveness with the state of the art trackers when tracking human faces, or even other types of targets. As a future work, we aim to apply our tracking algorithm to online person tracking by active IP PTZ camera. In such a system, the camera should be controlled after each target prediction to keep the subject in the field of view. The camera control is closely related to the tracking algorithm. It is essential to reduce the complexity of the tracking algorithm to process the images and control the camera quickly, without losing the target. In our algorithm, the complexity can be controlled by three parameters: 1) the number of particles generated at each iteration, 2) the size of the subset of particles used for keypoint matching, and 3) the keypoint detector/descriptor used.

(a) Video 1, Fr #53 (b) Fr #69 (c) Fr #80

(d) Video 2, Fr #264 (e) Fr #272 (f) Fr #280

(g) Video 3, Fr #32 (h) Fr #91 (i) Fr #102

(j) Video 4, Fr #70 (k) Fr #230 (l) Fr #300

(m) Video 5, Fr #218 (n) Fr #260 (o) Fr #282

(p) Video 6, Fr #49 (q) Fr #209 (r) Fr #233

(s) Video 7, Fr #29 (t) Fr #150 (u) Fr #340

Figure 4.1 Tracking results of the proposed method for different scenarios.

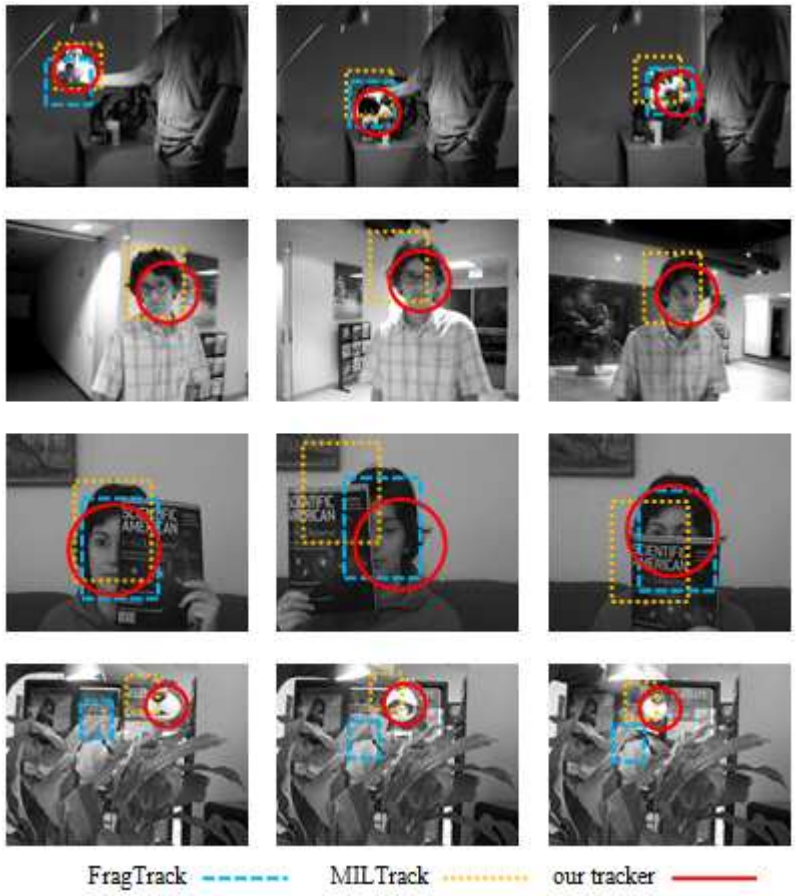


Figure 4.2 Screenshots of tracking results for FragTrack, MILTrack, and the proposed tracker. The rows correspond respectively to the video sequences *Sylvester*, *David indoor*, *Occluded face*, and *tiger 1*.

CHAPITRE 5

ARTICLE 2 : EXPLOITING STRUCTURAL CONSTRAINTS FOR VISUAL OBJECT TRACKING [1]

Abstract

This paper presents a novel structure-aware method for visual tracking. The proposed tracker relies on keypoint regions as salient and stable elements that encode the object structure efficiently. In addition to the object structural properties, the appearance model also includes global color features that we first use in a probabilistic approach to reduce the search space. The second step of our tracking procedure is based on keypoint matching to provide a preliminary prediction of the target state. Final prediction is then achieved by exploiting object structural constraints, where target keypoints vote for the corrected object location. Once the object location is obtained, we update the appearance model and structural properties, allowing to track targets with changing appearance and non-rigid structures. Extensive experiments demonstrate that the proposed Structure-Aware Tracker (SAT) outperforms recent state-of-the-art trackers in challenging scenarios, especially when the target is partly occluded and in moderately crowded scenes.

5.1 Introduction

Model-free visual tracking is one of the most active research areas in computer vision [3, 111, 112]. With a *model-free* tracker, the only available input is the target state annotated in the first video frame. Tracking an object is thus a challenging task due to (1) the lack of sufficient information on object appearance, (2) the inaccuracy in distinguishing the target from the background (which is generally done using a geometric shape), and (3) the object appearance change caused by various perturbation factors (*e.g.* noise, occlusion, motion, illumination, etc.).

This work aims to develop a novel visual tracking method to handle real life difficulties, particularly when tracking an object in a moderately crowded scene in the presence of distracting objects similar to the target, and in the case of severe partial occlusion. The

1. W. Bouachir et G.-A. Bilodeau, "Exploiting structural constraints for visual object tracking," article soumis à la revue Image and Vision Computing (IVC), février 2014 .

robustness of a tracking algorithm in handling these situations is determined by two major aspects: the target representation and the search strategy. The target representation refers to the appearance model that represents the object characteristics while the search strategy deals with how the search of the target is performed on every processed frame. The main contributions and differences of our work from previous works are on both aspects. In the proposed tracker, the target representation includes color features for coarse localization of the target, and keypoints for encoding the object structure while adding distinctiveness and robustness to occlusions. In our search strategy, probabilistic tracking and deterministic keypoint matching are used sequentially to provide a preliminary estimate of the target state. Object internal structural constraints are then applied in a correction step to find an accurate prediction. Our approach for representing the object structure is related to previous works on *context tracking* [95, 94, 113, 11, 114]. The main idea of *context tracking* is to consider the spatial context of the target including neighboring elements whose motion is correlated with the target. While the proposed approach is inspired by the idea of *context tracking*, in our work we exploit the spatial layout of keypoints to encode the internal structure of the target. More specifically, our contributions are:

1. A novel target representation model where local features are stored in a reservoir encoding recent and old structural properties of the target;
2. A new threefold search strategy that reduces the search space, tracks keypoints, and corrects prediction sequentially;
3. A discriminative approach that evaluates tracking quality online to determine if potential new target properties should be learned.

Extensive experiments on challenging video sequences show the validity of the proposed Structure-Aware Tracker (SAT) and its competitiveness with state-of-the-art trackers. A previous version of this work was presented at a conference [14]. This paper extends this previous work with a more complete review of related works, more details and depth in the explanation of the method, and additional experiments analyzing the tracker behavior in several situations.

This paper is organized as follows. In the next section, we review recent works on *keypoint tracking* and *context tracking* which are related to our algorithm. The proposed SAT algorithm is presented in section 5.3. Experimental results are given and discussed in section 5.4. Section 5.5 concludes the paper.

5.2 Related works

5.2.1 Keypoint tracking: from object context to object structure

Many tracking algorithms achieved good performances at a low complexity by using a geometric shape to contain the target, and global features for modeling [47, 41, 101]. Nevertheless, this approach is not designed to handle occlusions, unless representing the target by multiple fragments to be matched. Keypoint methods can handle the occlusion problem by establishing partial correspondences that allow locating the occluded target. Unlike fragment-based methods (where the target image region is divided randomly or according to a regular grid), keypoint locations correspond to salient and stable patches that can be invariantly detected under various perturbation factors. Moreover, their spatial layout naturally encodes structural properties that can enhance the target model.

Due to these characteristics, keypoint-based methods have attracted much attention during the last decade. In this approach, objects are modeled as a set of keypoints detected by an external mechanism (i.e. a keypoint detector) [103, 93, 12]. After computing their descriptors, the object localization can be achieved according to two possible approaches: matching in the case of a generative approach, and classification in the case of a discriminative approach. Generative trackers use a database where keypoint descriptors are stored. The descriptors are designed to be stable and invariant, and can be matched in a nearest-neighbor fashion. Discriminative approaches consider matching as a binary classification problem. Every feature is thus classified as belonging to the background, or to the tracked object. The classifier is built either via online learning, or offline, considering the background and the target observed under various transformations.

Some recent works on object tracking rely on target context to predict its state, which is often referred as *context tracking* [95, 94, 96, 11, 98]. According to this approach, it is necessary to consider target context to ensure the tracker robustness in most real life video surveillance applications. Following this principle, the authors in [95] use a *compagnion* to improve object tracking. This corresponds to image regions around the tracked object with the same movements as those of the target. In [94] the spatial context that can help the tracker includes multiple *auxiliary objects*. These objects have consistent motion correlation with the tracked target and thus help to avoid the drifting problem. In [96], Gu and Tomasi consider the spatial relationship between the target and similar objects and track all of them simultaneously to eliminate target confusion. In a more general approach, Grabner et al. introduced the notion of *supporters* defined as *"useful features for predicting the target object position"* [11]. These features do not belong to the target, but they move in a way that is statistically related to the motion of the target. They developed a method for discovering

these local image features around the target, and demonstrated that motion coupling of *supporters* may allow locating the target even if it is completely occluded. In a later work, Dinh et al. [98] used *supporters* for context tracking, and added the concept of *distracters* which are regions co-occuring with the target while having a similar appearance. Their tracker explicitly handles situations where several objects similar to the target are present.

Context tracking methods expanded the target model by exploiting the motion correlation information in the scene. However, finding motion correlation between objects is a costly task that often requires detecting and analyzing features on the whole image, as in [115] where the authors detect and analyze all local features in the scene, to keep only features which move along with the target object. Furthermore, most of the proposed trackers were tested only on specific scenarios and in constrained environments, where almost all the experiments were limited to proofs of concept. Our idea of using structural constraints in the target appearance model is inspired by *context tracking* methods. However, our motivations differ in an important aspect since our model incorporates the internal structural information of the target, and not the structural layout of different scene elements. In our work, we show that the structural information of the target, encoded by the keypoint spatial layout, allows achieving accurate tracking and handling partial occlusion by inferring the position of the target using the unoccluded features.

5.2.2 Tracking objects by structure

The idea of exploiting object structure for tracking was present, more or less explicitly, in recent works. This is the so called *part-based tracking* that relies on local components for target representation. The most common way to encode object structure is the sparse representation such as in [83] and [84]. In [83], the authors propose to use a histogram-based model that encodes the spatial information of the object patches. In a similar manner, Jia et al. sample a set of overlapped patches on the tracked object [84]. Their strategy includes an occlusion handling module allowing target localization by using only visible image patches.

Another approach for encoding structure consists in using keypoints, since they are more significant than random overlapped patches. In this direction, the authors in [77] model the target by a set of keypoint *manifolds* organized as a graph to explicitly represent the target structure. Each feature *manifolds* includes, in addition to the keypoint descriptor, a set of synthetic descriptors simulating possible variations of the original feature (under viewpoint and scale change). The target location is found by detecting keypoints on the current frame, matching them with those of the target model, and computing a homography for the correspondences. In [92], the authors include both random patches and keypoints in the target model. The random patches are described by their RGB color histograms

and LBP (Local Binary Patterns) descriptors to form an appearance model. Keypoints are characterized by their spatial histograms to be considered as a structural model. Tracking then implies matching detected keypoints in the current frame with those of the object in the previous frame. Matched keypoints are utilized to construct a spatial histogram, which is used jointly with LBP and RGB histograms to locate the target. This approach exploits multiple object characteristics (LBP, color, Keypoints), but the object structural model captures only recent structural properties, as the spatial histogram considers only the keypoints that are matched with those of the target in the last frame.

In our work, we argue and demonstrate through our experiments that keypoint regions are more efficient than random patches in encoding the structure, as they correspond to salient and stable patches invariably detectable under several perturbation factors. Unlike in [92] where random regions are analyzed to extract local features, and [77] where keypoints are extracted from a region with a fixed size (with the assumption of small displacements), we use a probabilistic method to reduce the search space to the most likely image regions, based on the target's global color features. Concerning the target structure, our structural model is not limited, like in [92] to recent properties, which would make it strongly related to the last prediction (and thus may be completely contaminated if the tracker drifts from the target). Instead, our representation includes both recent and old structural constraints in a reservoir of features. The local features and their structural constraints are learned online during tracking. The deletion of a given feature is related to its persistence (not to its moment of occurrence), while the impact of its constraint depends on the persistence as well as the consistence of the feature. Every local feature expresses its structural constraint individually by voting to possible target locations. Thus, our voting-based method preserves the object structure without requiring building and updating complex keypoint graphs, neither calculating homographies such as in [77]. Our method takes into consideration the temporal information of all the target's model components. The target model is thus updated to reflect the object appearance changes including structure changes, which allows tracking objects with non-rigid structures.

5.3 Proposed algorithm

5.3.1 Motivation and overview

The proposed method is illustrated in figure 5.1 where we aim to track a partly occluded face. First, we apply a color-based particle filtering. This allows to reduce the search space and provides a coarse estimation by considering only the best particles. Keypoints are then detected by analyzing the reduced search space as shown in figure 5.1a. The detected key-

(a) (b) (c) (d)

Figure 5.1 Illustration of the SAT algorithm steps when tracking a partly occluded face. 5.1a: Reducing the search space with a probabilistic method, based on color. Local features (red dots) are computed only on the obtained areas. 5.1b: Predicting a preliminary target state based on feature matching. 5.1c: Visible features vote for a new position (yellow star) by applying their structural constraints. 5.1d: The target state is corrected based on the new location

points are matched with those of the target model, which leads to a preliminary estimate of the target location (see figure 5.1b).

Note that the preliminary prediction considers only the matching scores of the particles and thus does not guarantee an accurate localization. This is illustrated in figure 5.1b, where the circular shape representing the best particle includes pixels from the background and from the occluding object. Knowing the internal structure of the target, our idea is to perform a correction step by applying internal structural constraints to improve target prediction. In practice, this is carried out by a voting mechanism where available features (unoccluded) determine the exact position of the target (figure 5.1c and 5.1d). Once the target is predicted, the appearance model including keypoints and their structural constraints is updated according to an evaluation criterion (that we define in section 5.3.5). The newly detected keypoints are added to the model while existing keypoints are re-evaluated based on two properties. First, we consider the individual keypoint persistence represented by its weight value. The second property is the spatial consistency of the keypoint that depends on the motion correlation with the target center. If a keypoint of the background is erroneously included in the target model, these two voting parameters will reduce the effect of its vote until its removal from the model when its persistence decreases significantly. Our algorithm steps are explained in details in the following.

5.3.2 Appearance Model

Our appearance model describes the image region delimited by the circle that circumscribes the target. This is a multi-features model including (1) the color probability distribution represented by a weighted histogram, (2) a set of local descriptors computed for the

detected keypoints within the target region, and (3) the target structural properties encoded by the voting parameters of keypoints. By constructing a m-bin histogram $\hat{\mathbf{q}} = \{\hat{q}\}_{u=1...m}$, with $\sum_{u=1}^{m} \hat{q}_u = 1$, some parts of the background may lie inside the circular kernel. As discussed in [105], these pixels will affect the color distribution and may cause tracking drift. To reduce the effect of these pixels, we use a kernel function $k(x)$ that assigns smaller weights to pixels farther from the center. The color histogram is thus computed for the h pixels inside the target region according to the equation:

$$\hat{q}_u = \frac{1}{\sum_{i=1}^{h} k(d_i)} \sum_{i=1}^{h} k(d_i)\delta[c_i - u] \tag{5.1}$$

where $d_i \in [0, 1]$ is the normalized distance from the pixel x_i to the kernel center, c_i is the bin index for x_i in the quantized space, δ is the Kronecker delta function, and $k(d_i)$ is the tricube kernel profile defined by:

$$k(d_i) = \frac{70}{81}(1 - d_i^3)^3. \tag{5.2}$$

Note that in this work, we use a RGB color histogram but any other color space could be used.

The proposed system should be able to handle many difficult scenarios, such as occlusions and the presence of distracting objects. For, example, it has been shown that even for individuals of different races, the skin color distributions are very similar [106]. To ensure a more robust and distinctive feature set, the target reference model also includes SIFT keypoints [89] detected in the target region and stored in a *Reservoir of Features* (*RF*). SIFT features increase the distinctiveness of the tracking algorithm to distinguish the target from other similar objects that may enter the field of view. In fact, SIFT was successfully used for distinguishing between multiple instances of the same object such as in the face recognition problem[116, 117, 118]. In this way, we implicitly handle situations where objects of the same category as the target co-occur (*e.g.* tracking a face in the presence of several faces), and thus we avoid using an additional mechanism to track and distinguish *distracters* as in [98]. Other than the keypoint descriptors, we also exploit the spatial layout of keypoints to encode structural properties of objects. The target structural constraints and the voting method that we use for prediction correction are explained later. We note that our method is not specific to SIFT. Even faster keypoint detector/descriptor combination may be used, although SIFT remains one of the most reliable methods under various image transformations [107].

5.3.3 Reducing the search space

The target search is firstly guided by particle filtering [52]. Each particle is a circular region characterized by its color distribution as explained above. The possible target states at frame t are represented by N weighted particles $\{s_t^{(i)} : i = 1, ..., N\}$ where the weight $\pi_t^{(i)}$ reflects the importance of the particle. The weight of a generated particle $s_t^{(i)}$ depends on the similarity between its color distribution $\hat{p}_t^{(i)}$ and the reference color model $\hat{\mathbf{q}}$. We define the distance between the two distributions as:

$$d(\hat{\mathbf{q}}, \hat{p}_t^{(i)}) = \sqrt{1 - \rho[\hat{q}, \hat{p}_t^{(i)}]} \tag{5.3}$$

where

$$\rho[\hat{q}, \hat{p}_t^{(i)}] = \sum_{u=1}^{m} \sqrt{\hat{q}_u . \hat{p}_{u,t}^{(i)}} \tag{5.4}$$

is the Bhattacharyya coefficient between $\hat{\mathbf{q}}$ and $\hat{p}_t^{(i)}$.

After generating N particles on the current frame, the area covered by the N^* best particles (i.e. the particles having the highest weights) is considered as a coarse estimation of the target state, and thus constitutes a reduced search space where keypoints will be detected and matched. Moreover, we use the N^* states selected at frame t for generating N particles at frame $t + 1$. Note that to simplify computations, we assign a cumulative weight $c^{(n)}$ to each pair $(s^{(n)}, \pi^{(n)})$ where $c^{(N^*)} = 1$. Our space reduction algorithm is summarized in Alg. 1.

5.3.4 Tracking keypoints

Keypoint detection and matching will consider only the reduced search space defined by the N^* best particles. By reducing the search region to the most important candidate particles, we avoid detecting features, computing local descriptors and matching them on the entire image.

The detected descriptors are then matched with those of the target model (features from the reservoir RF) based on the Euclidian distance. Similarly to the criterion used in [89], we determine if a match is correct by evaluating the ratio of distance from the closest neighbor to the distance of the second closest. For our algorithm, we keep only the matches for which the distance ratio is less than $\theta_m = 0.7$. Given the final set of matched pairs, we consider the particle having the highest matching score as a preliminary state of the target (see figure 5.1b). A more formal description of the preliminary prediction is provided in Alg. 2. Since the preliminary prediction considers only matching scores, without guaranteeing an accurate localization of the selected particle, the structural properties of the predicted region will be

Algorithm 1 Reducing the search space at frame t

for $i = 1$ **to** N **do**
  - generate a random number $r_i \in [0, 1]$
  - find the particle $s_{t-1}^{(j)}$ with the smallest j verifying $c_{t-1}^{(j)} \geq r_i$
  - generate for the selected particle $\hat{s}_t^{(j)}$, a new particle $s_t^{(i)}$ with $s_t^{(i)} = f(\hat{s}_t^{(j)})$
  - evaluate similarity between $\hat{p}_t^{(i)}$ and $\hat{\mathbf{q}}$ {Eq. 5.3 and 5.4}
  - compute the weight $\pi_t^{(i)}$ for $s_t^{(i)}$
end for
- select the N^* best particles {This is the main output; the following operations are preparation for the next frame.}
- normalize weights $\pi_t^{(n)}$, for all $n \leq N^*$
- compute cumulative probabilities $c_t^{(n)}$

analyzed in a correction step to provide an accurate estimation of the target location.

5.3.5 Applying structural constraints

In this step, we aim to correct the preliminary prediction by applying a learned structural model of the target. The model is learned from reliable measurements (*i.e.* when a good tracking is achieved), and the internal structural properties are considered as a part of the object appearance model.

Internal structural model. The target keypoints extracted on the target region at different times of its lifecycle are stored in the reservoir of features RF. Instead of automatically eliminating old keypoints, we only remove those that become "non-persistent". RF is thus formed by recent and old keypoints, representing both old and recent object properties. Other than its descriptor summarizing the local gradient information, every keypoint is characterized by a *voting profile* (μ, w, Σ) where:

- $\mu = [\Delta_x, \Delta_y]$ is the average offset vector that describes the keypoint's location with respect to the target region center;
- w is the keypoint's weight considered as a persistence indicator to reflect the feature co-occurence with the target, and to allow eliminating "bad" keypoints;
- Σ is the covariance matrix used as a spatial consistency indicator, depending on the motion correlation with the target center.

Voting. Every matched keypoint f that is located on the preliminary target region votes for the potential object position $\mathbf{x}$ by $P(\mathbf{x}|f)$. Note that we accumulate the votes for all the pixel positions inside the reduced search space. Given the *voting profile* of the feature f, we

Algorithm 2 Preliminary prediction at frame t

- detect features on the reduced search space
for all detected_features $f^{(i)}$ **do**
  - compute Euclidian distance with features from RF
  - compute $dist\_ratio = \frac{dist(f^{(i)}, closest\_neighbor)}{dist(f^{(i)}, 2^{nd}\_closest\_neighbor)}$
  if $dist\_ratio \leq \theta_m$ **then**
    - match $f^{(i)}$ with $closest\_neighbor$
    - update matching scores for the particles containing $f^{(i)}$
  end if
end for
- $preliminary\_prediction_t$= the particle having the highest score

estimate the voting of f with the Gaussian probability density function:

$$P(\mathbf{x}|f) \propto \frac{1}{\sqrt{2\pi|\Sigma|}}\ exp\left(-0.5\left(\mathbf{x}_f - \mu\right)^\top \Sigma^{-1}(\mathbf{x}_f - \mu)\right), \tag{5.5}$$

where $\mathbf{x}_f$ is the relative location of $\mathbf{x}$ with respect to the keypoint coordinates. The probability of a given pixel in the voting space is estimated by accumulating the votes of keypoints weighted by their persistence indicators w. The probability for a given pixel position $\mathbf{x}$ in the voting space at time t is estimated by:

$$P_t(\mathbf{x}) \propto \sum_{i=1}^{|RF|} w_t^{(i)} P_t(\mathbf{x}|f^{(i)}) \mathbb{1}_{\{f^{(i)} \in F_t\}}, \tag{5.6}$$

where $\mathbb{1}_{\{f^{(i)} \in F_t\}}$ is the indicator function defined on the set RF (reservoir of features), indicating if the considered feature $f^{(i)}$ is among the matched target features set F_t at frame t. The target position is then found by analyzing the voting space and selecting its peak to obtain the corrected target state as shown in figure 5.1c.

Update. It has been previously shown that an adaptive target model, evolving during the tracking, is the key to good performance [99]. In our algorithm, the target model (including color, keypoints, and structural constraints) is updated every time we achieve a good tracking using a discriminative approach. Our definition of a good tracking is inspired by the Bayesian evaluation method used in [45], referred as *histogram filtering*. Using the target histogram $\hat{\mathbf{q}}$ (calculated for the target region annotated in the first frame), and the background histogram $\hat{\mathbf{q}}_{bg}$ (calculated for the area outside the reduced search space), we compute a filtered histogram $\hat{\mathbf{q}}_{filt} = \hat{\mathbf{q}}/\hat{\mathbf{q}}_{bg}$ in every iteration. The latter represents the likelihood ratios of pixels belonging to the target. The likelihood ratios are used to calculate a backprojection map on the target

region. Quality evaluation is done by analyzing the backprojection map and thresholding it to determine the percentage of pixels belonging to the target. Every time the evaluation procedure shows sufficient tracking quality, the target model is updated at frame t with a learning factor α as follows:

$$\hat{q}_t = (1-\alpha)\hat{q}_{t-1} + \alpha\hat{q}_{new} \tag{5.7}$$

$$\hat{q}_{bg,t} = (1-\alpha)\hat{q}_{bg,t-1} + \alpha\hat{q}_{bg,new} \tag{5.8}$$

$$w_t^{(i)} = (1-\alpha)w_{t-1}^{(i)} + \alpha\mathbb{1}_{\{f^{(i)}\in F_t\}} \tag{5.9}$$

$$\Delta_{x,t}^{(i)} = (1-\alpha)\Delta_{x,t-1}^{(i)} + \alpha\Delta_{x,new}^{(i)} \tag{5.10}$$

$$\Delta_{y,t}^{(i)} = (1-\alpha)\Delta_{y,t-1}^{(i)} + \alpha\Delta_{y,new}^{(i)} \tag{5.11}$$

where $\mu_{new}^{(i)} = [\Delta_{x,new}^{(i)}, \Delta_{y,new}^{(i)}]$ is the current estimate of the voting vector for the feature $f^{(i)}$. After updating the feature weights, we remove from RF all the features whose the persistence indicators become less than the persistence threshold θ_p (*i.e.* $w_t^{(i)} \leq \theta_p$) regardless if they are recent or old, and we add the newly detected features with initial weight w_0. Further, we update the covariance matrix to determine the spatial consistency of the feature by applying:

$$\Sigma_t^{(i)} = (1-\alpha)\Sigma_{t-1}^{(i)} + \alpha\Sigma_{new}^{(i)}, \tag{5.12}$$

where the new correlation estimate is:

$$\Sigma_{new}^{(i)} = (\mu_{new}^{(i)} - \mu_t^{(i)})(\mu_{new}^{(i)} - \mu_t^{(i)})^\top, \tag{5.13}$$

with $\mu_t^{(i)} = [\Delta_{x,t}^{(i)}, \Delta_{y,t}^{(i)}]$. Note that for the newly detected features, the preliminary persistence indicator is initialized to the covariance matrix $\Sigma = \sigma_0^2 I_2$, where I_2 is a 2 x 2 identity matrix. For consistent features, Σ decreases during the tracking, and thus their votes become more concentrated in the voting space. The overall algorithm is presented in Alg. 3.

5.4 Experiments

5.4.1 Experimental setup

We evaluated our SAT tracker by comparing it with four recent state-of-the-art methods on 11 challenging video sequences. Seven sequences of the dataset are publicly available and

Algorithm 3 Predicting the target location

- initialize RF, $\hat{q}$, $\hat{q}_{bg}$
for all $frames$ **do**
  - reduce the search space: Alg. 1
  - predict a preliminary state: Alg. 2
  for all $voting\_space\_positions$ **x do**
    for all $matched\_features$ ($f^{(i)} \in F_t$) **do**
      - estimate $P(\mathbf{x}|f^{(i)})$: (Eq. 5.5)
    end for
    - estimate location probability $P(\mathbf{x})$: (Eq. 5.6)
  end for
  - $target\_location = select\_peak(voting\_space\_positions)$ {tracker's output for the current frame}
  if ($update\_condition == true$) **then**
    -update $\hat{q}_t$ and $\hat{q}_{bg,t}$: (Eq. 5.7 & 5.8)
    for all $matched\_features$ ($f^{(i)} \in F_t$) **do**
      - update $\mu_t^{(i)}$ (Eq. 5.10 & 5.11)
      - update $\Sigma_t^{(i)}$ (Eq. 5.12)
    end for
    - update $w_t^{(i)}$ (Eq. 5.9) for the entire reservoir
    - remove non-persistent features ($i.e.\, w_t^{(i)} \leq \theta_p$)
    for all $newly\_detected\_features$ $f^{(i)}$ **do**
      - add $f^{(i)}$ to RF
      - $\mu_t^{(i)} = [\Delta_{x,new}^{(i)}, \Delta_{x,new}^{(i)}]$; $\Sigma_t^{(i)} = \sigma_0^2 I_2$; $w_t^{(i)} = w_0$
    end for
  end if
end for

commonly used in the literature, while four are our own sequences [2]. The *Tiger 1*, *Tiger2* and *Cliff bar* are provided in [3] and the *David indoor* and *Sylvester* are from [6]. The *Girl* and *occluded face 1* video sequences are respectively from [18] and [2]. The sequences *jp1*, *jp2*, *wdesk*, and *wbook* (with 608, 229, 709, and 581 frames respectively) were captured in our laboratory using a Sony SNC-RZ50N camera. The video frames are 320x240 pixels captured at a frame rate of 15 fps. For quantitative evaluation, we manually labeled the ground truth of our four sequences. Some of the sequences are available only in grayscale format (*Tiger 1*, *Tiger2* , *Sylvester*, and *Cliff bar*). For these videos, we slightly adapted our algorithm (especially the color model) to use grayscale information instead of RGB color information.

The four methods that we used for our comparison are the SuperPixel Tracker (SPT)

2. Our sequences are available at http://www.polymtl.ca/litiv/en/vid/.

[85], the Sparsity-Based Collaborative Tracker (SBCT) [83], the Adaptive Structural Tracker (AST) [84], and the Online Multiple Support Instance Tracker (OMSIT) [119]. The source codes of these trackers are available on the authors' respective websites. The authors also provide various parameter combinations. For fairness, we tuned the parameters of their methods so that for every video sequence, we always use the best combination among the ones that they proposed.

We quantitatively evaluated the performance of the trackers using the success rate and the average location error. To measure the success rate, we calculate for each frame the Overlap Ratio $OR = \frac{area(P_r \cap G_r)}{area(P_r \cup G_r)}$, where P_r is the predicted target region and G_r is the ground truth target region. Tracking is considered as a success for a given frame, if OR is larger than 0.5. The evaluation of the Center Location Error (CLE) is based on the relative position errors between the center of the tracking result and that of the ground truth. Table 5.1 presents the success rates and the average center location errors for the compared methods. In order to analyze in depth the compared methods on several video sequences, we also prepared two plots for every video sequence: 1) the center location error versus the frame number presented in figure 5.6, and 2) the overlap ratio versus the frame number presented in figure 5.7. These plots are useful for understanding more in details the behavior of the trackers since the success rate and the average location error just summarize the performance of the tracker on a given sequence. Note that we averaged the results over five runs in all our experiments.

5.4.2 Experimental results

5.4.2.1 Long-time occlusion

Figure 5.2 demonstrates the performance of the compared trackers when tracking faces under long-time partial occlusions. In the *Occluded face 1* and the *wbook* sequences, the target faces remain partially occluded for several seconds while they barely move. The corresponding plots in figures 5.6 and 5.7 show that some trackers drift away from the target face, to track the occluding object (*e.g.* between frames 200 and 400 in *Occluded face 1*). Because it is specifically designed to handle partial occlusions via its structure-based model, our tracker was able to track the faces successfully in practically all the frames. SBCT has also achieved a good performance with a slightly lower average location error. In fact, SBCT is also designed to handle occlusions using a scheme that considers only the patches that are not occluded. The target face in *Wdesk* undergoes severe partial occlusions many times while moving behind structures of the background. SAT and SBCT track the target correctly until frame 400. At this point the person performs large displacements, and SBCT drifts away

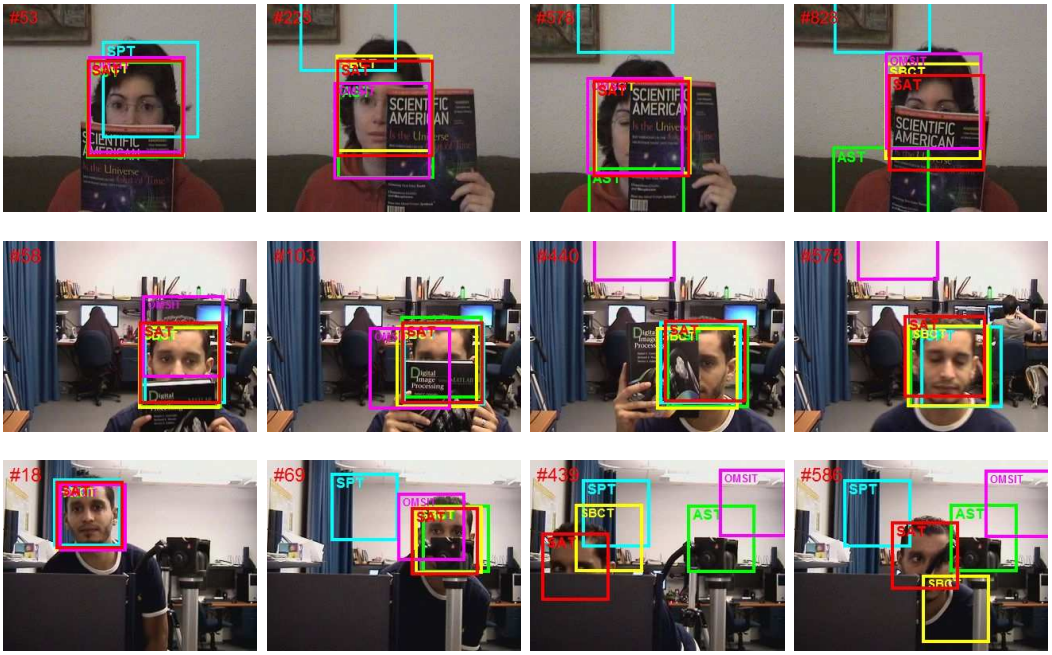


Figure 5.2 Tracking results for video sequences with long-term occlusions: *Occluded face 1, Wbook, Wdesk*. Green, magenta, yellow, cyan, and red rectangles correspond to results from AST, OMSIT, SBCT, SPT, SAT.

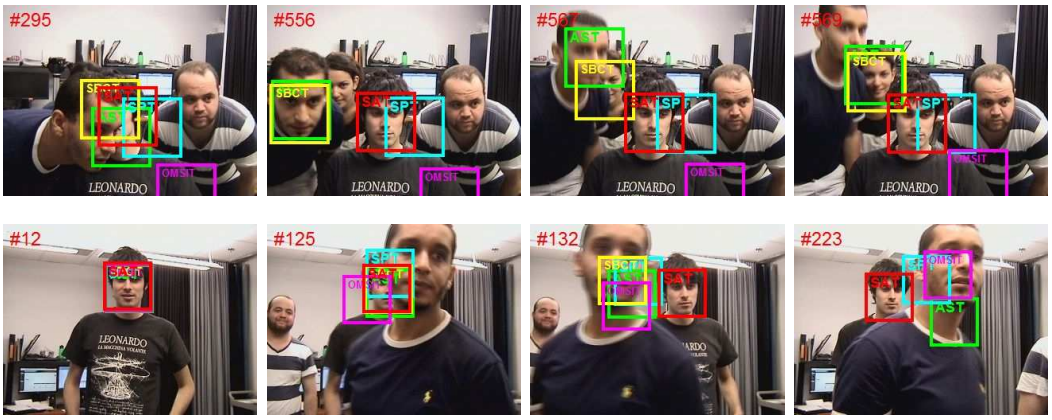


Figure 5.3 Screenshots of face tracking in moderately crowded scenes under short-term occlusions. In the *Jp1* sequence (first row), the tracked face is the one that is in the center of the scene. The same person is tracked while he is walking in the *Jp2* sequence. Green, magenta, yellow, cyan, and red rectangles correspond to results from AST, OMSIT, SBCT, SPT, SAT.

Figure 5.4 Screenshots of tracking results for some of the sequences with illumination change (*david indoor*) and background clutter (*Cliff bar, Tiger1, Tiger2*). Green, magenta, yellow, cyan, and red rectangles correspond to results from AST, OMSIT, SBCT, SPT, SAT.

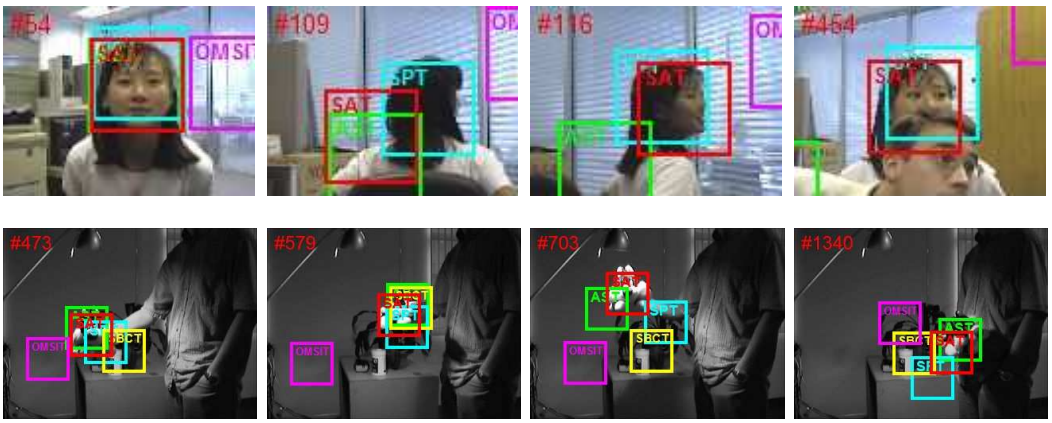


Figure 5.5 Tracking results for video sequences with abrupt motion and/or out of plane rotation: *Girl* and *Sylvester* sequences. Green, magenta, yellow, cyan, and red rectangles correspond to results from AST, OMSIT, SBCT, SPT, SAT.

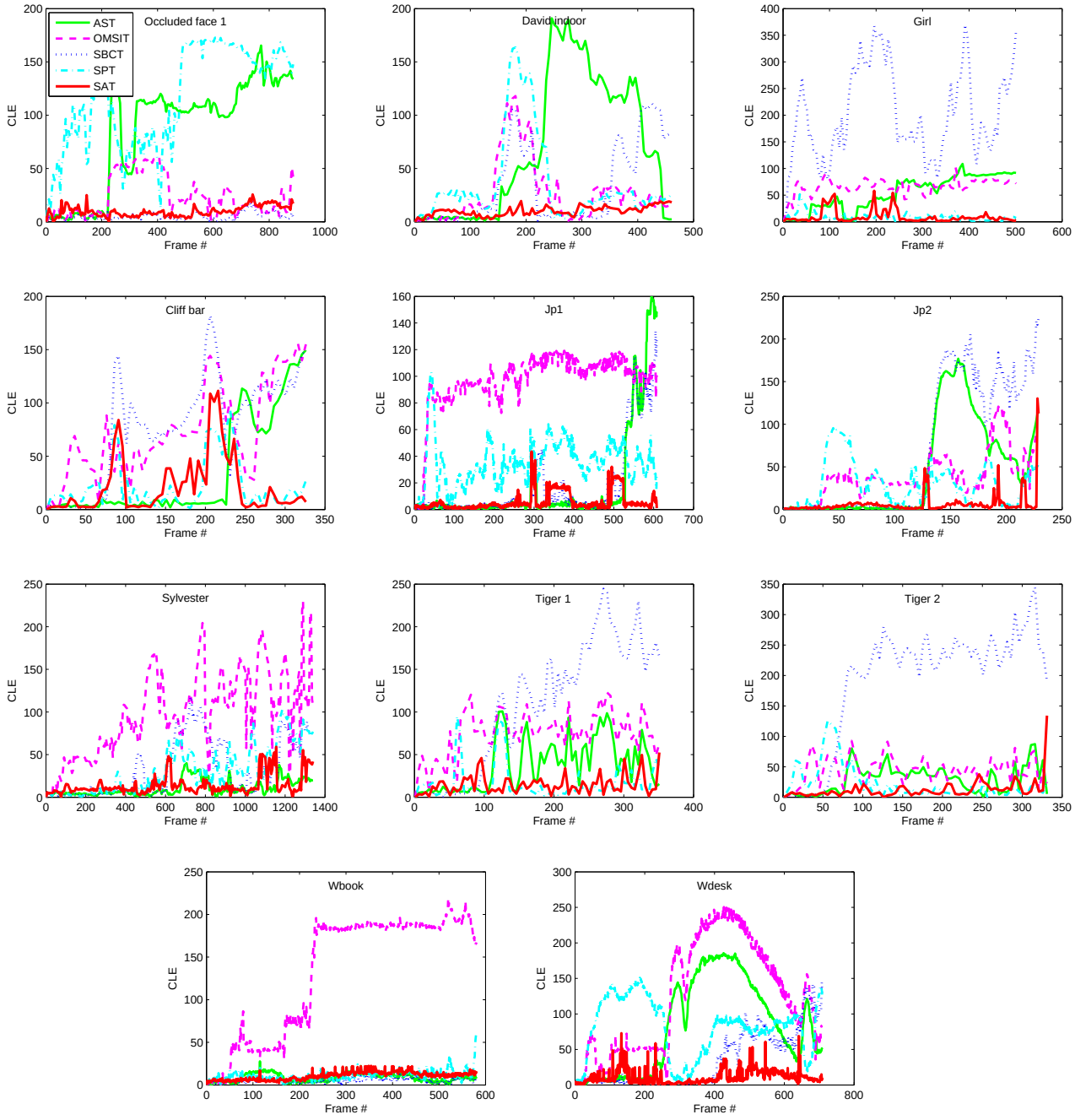


Figure 5.6 Center location error plots.

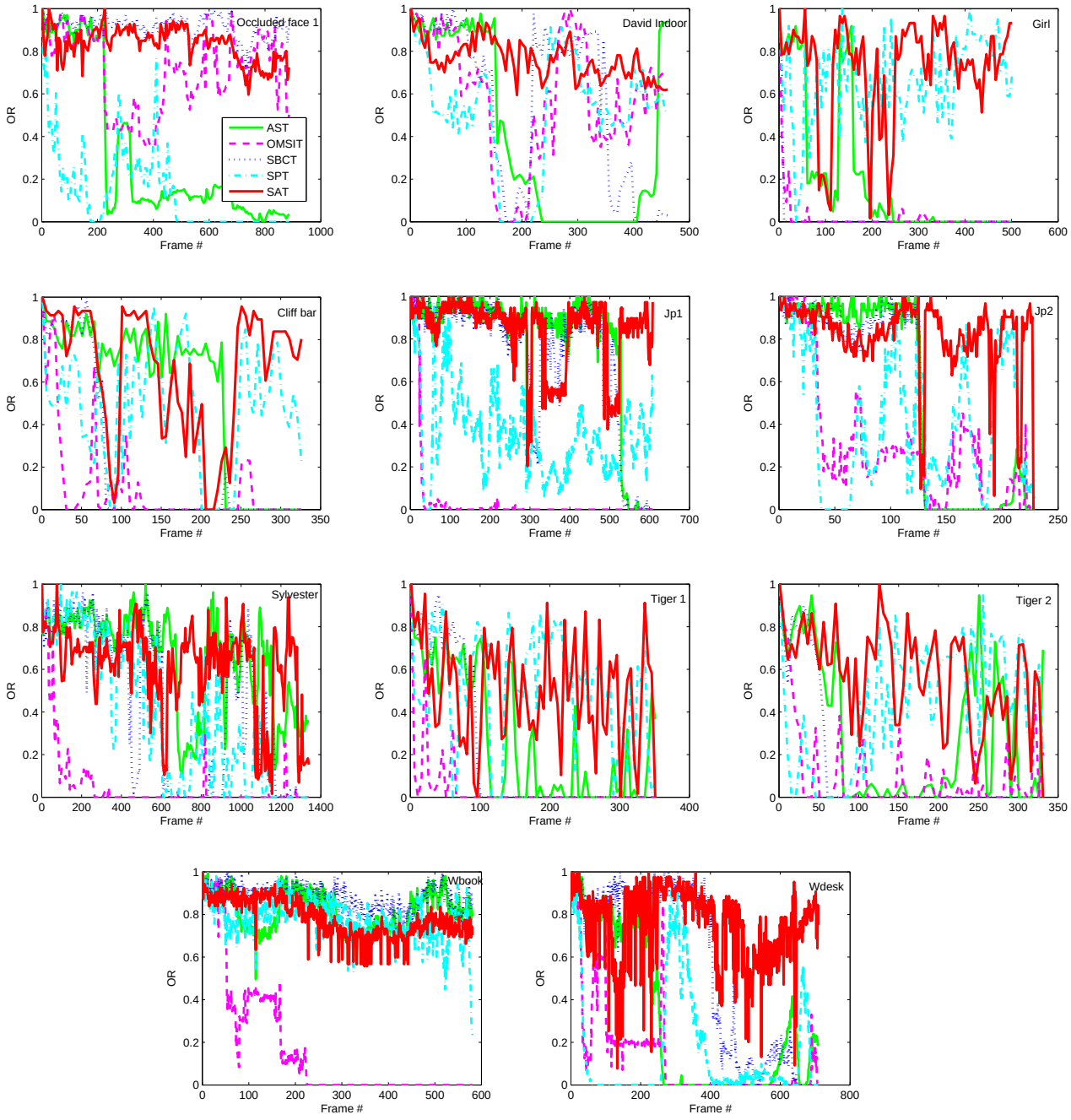


Figure 5.7 Overlap ratio plots.

Table 5.1 Success rate (S) and average location error (E) results for SAT and the four other trackers: **Bold red** font indicates best results, *blue italics* indicates second best.

	SPT		SBCT		AST		OMSIT		SAT	
Sequence	S	E	S	E	S	E	S	E	S	E
David indoor	62	36	60	34	38	69	*63*	*27*	**100**	**10**
girl	*84*	**9**	2	201	18	53	1	66	**85**	*10*
occluded face 1	6	117	**100**	**5**	26	85	81	23	**100**	*14*
tiger 1	**61**	*17*	25	108	31	38	3	75	*51*	**15**
tiger 2	*46*	*23*	16	189	31	29	6	45	**70**	**16**
Sylvester	39	32	49	34	*73*	**10**	3	99	**79**	*14*
Cliff bar	52	**22**	24	77	**70**	35	8	74	*60*	*25*
Jp1	18	35	78	18	*84*	*17*	4	97	**89**	**7**
Jp2	39	*31*	*55*	69	*55*	45	17	39	**94**	**7**
Wdesk	14	80	*57*	*34*	32	81	10	123	**90**	**11**
Wbook	99	11	**100**	**5**	**100**	*9*	9	132	**100**	12
average	47	*38*	*52*	70	51	43	19	73	**84**	**13**

from the face. Nevertheless, our tracker continues the tracking successfully while the tracked person is trying to hide behind structures of the background, achieving a success rate of 90%. The superiority of the proposed method in this experiment highlights the importance of using structural constraints defined by keypoint regions that are more invariant than the patches used in SBCT when such a situation occurs.

5.4.2.2 Moderately crowded scenes

Figure 5.3 presents the results of face tracking in a moderately crowded scene (four persons). In the *Jp1* video, we aim to track a target face in presence of other faces that may partially occlude the target. Although the success rates of 84% and 78% respectively for AST and SBCT indicate good performance in general, the two trackers drift twice, first at frame 530, and a second time at frame 570, to track other faces occluding or neighboring the target face. However, our tracker is not affected by the presence of similar objects around the target, even if partial occlusion occurs. This is mainly due to the distinctiveness of SIFT features compared to the local patches used in AST and SBCT to characterize the target. In this manner, SIFT features allow our tracker to handle situations where multiple instances of the same target object co-occur. In the *jp2* sequence, we track a walking person in a moderately crowded scene with four randomly moving persons. Here, we track a person's face that crosses in front or behind another walking person that may completely occlude the target for a short time. Except the proposed method, none of the trackers is able to relocate

the target after full occlusion by another person. For example, SBCT confused the target with the occluding face like in the video sequence *Jp1*. In this situation, SAT detects a total occlusion (since no features are matched). Our tracker continues searching the target based on color similarity without updating the appearance model. Tracking is finally recovered as soon as a small part of the target face becomes visible and feature matching becomes possible again.

5.4.2.3 Illumination change

In the *David indoor* video, the illumination changes gradually as the person moves from a dark room to an illuminated area (see figure 5.4). While most of the trackers were able to keep track of the person in more than 60% of the frames, SAT was the only tracker to achieve a success rate of 100%. In addition, SAT had the best performance on the *Sylvester* sequence in which the target object appearance changes drastically due to abrupt illumination change. These two experiments show the superiority of our appearance model, which is the only one among the five models, to include keypoints that are robust against lighting variations. Note that every time we update the reservoir of features, we replace the descriptors of all matched keypoints by their latest version computed on the current frame. This technique helps also to reflect appearance changes of keypoint regions (caused by illumination, viewpoint change, etc.), which facilitates matching features.

5.4.2.4 Background clutters

In the *Cliff bar* video, the background (the book) and the target have similar textures. Figure 5.4 shows that SBCT and OMSIT drift away from the target in most video frames. AST, SPT, and the proposed tracker were able to achieve a better performance despite the difficulty of this sequence. In fact, the target undergoes drastic appearance changes due to high motion blur. This caused drifts for all trackers several times (*e.g.* see the corresponding CLE and OR plots at frame 80). In the *Tiger 1* and *Tiger 2* sequences, the tracked object exhibits fast movements in a cluttered background with frequent and various occlusion level. Owing to our voting mechanism that predicts the exact position of the target from the visible keypoints, our SAT tracker overcomes the frequent occlusion problem outperforming the other methods. All the other methods fail to locate the stuffed animal, except SPT that achieved better results due to its discriminative appearance model that facilitates the distinction between the object and the background based on superpixel over-segmentation. Note that our method also presents a discriminative aspect, since it uses information on the background color distribution to evaluate the tracking quality (see the update subsection

under section 5.3.5).

5.4.2.5 Abrupt motion and out of plane rotation

The target object in *Sylvester* undergoes out of plane rotation and sudden movements during more than 1300 frames. Most of the trackers, except AST and ours do not perform well. In the *girl* video, the tracked face undergoes both pose change and 360 degrees rotations abruptly. Our method had the highest success rate and was significantly more robust and accurate than most of the methods as we can see in figure 5.5. SAT handled efficiently pose change and partial occlusion and our tracking was successful as long as the girl's face was at least partly visible. The target was lost only during the frames where it is completely turned away from the camera (see the OR plot, frames 87-116 and 187-250), but tracking is recovered as soon as the face reappears.

5.5 Conclusion

In this paper, we proposed a robust tracking algorithm named SAT (Structure Aware Tracker). Our core idea is to exploit the structural properties of the target, in a voting-based method, to provide accurate location prediction. The target is described by color distribution, keypoints, and their geometrical constraints encoding the object internal structure. This multi-features appearance model is learned during tracking and thus incorporates new structural properties in an online manner. Numerous experiments in a comparison with four state-of-the-art trackers, on eleven challenging video sequences, demonstrate the superiority of the proposed method in handling multiple tracking perturbation factors. Our results also highlight the importance of encoding the object structure via keypoint regions, that are more invariant and stable than other types of patches (*e.g.* the local patches encoding the object spatial information in AST and SBCT).

CHAPITRE 6

ARTICLE 3 : COLLABORATIVE PART-BASED TRACKING USING SALIENT LOCAL PREDICTORS [1]

Abstract

This work proposes a novel part-based method for visual object tracking. In our model, keypoints are considered as elementary predictors localizing the target in a collaborative search strategy. While numerous methods have been proposed in the model-free tracking literature, finding the most relevant features to track remains a challenging problem. To distinguish reliable features from outliers and bad predictors, we evaluate feature saliency comprising three factors: the *persistence*, the *spatial consistency*, and the *predictive power* of a local feature. Saliency information is learned during tracking to be exploited in several algorithm components: local prediction, global localization, model update, and scale change estimation. By encoding the object structure via the spatial layout of the most salient features, the proposed method is able to accomplish successful tracking in difficult real life situations such as long-term occlusion, presence of distractors, and background clutter. The proposed method shows its robustness on challenging public video sequences, outperforming significantly recent state-of-the-art trackers. Our Salient Collaborating Features Tracker (SCFT) also demonstrated a high accuracy even if a few local features are available.

6.1 Introduction

Visual object tracking is a fundamental problem in computer vision with a wide range of applications including automated video monitoring systems [120, 121], traffic monitoring [122, 123], human action recognition [124], robot perception [125], etc. While significant progress has been made in designing sophisticated appearance models and effective target search methods, *model-free* tracking remains a difficult problem receiving a great interest. With *model-free* trackers, the only information available on the target appearance is the bounding box region in the first video frame. Tracking is thus a challenging task due to (1) the insufficient amount of information on object appearance, (2) the inaccuracy in distinguishing

1. W. Bouachir et G.-A. Bilodeau, "Collaborative part-based tracking using salient local predictors," article soumis à la revue Computer Vision and Image Understanding (CVIU), août 2014.

the target from the background, and (3) the target appearance change during tracking.

In this paper, we present a novel part-based tracker handling the aforementioned difficulties, including the lack of information on object appearance and features. This work demonstrates that an efficient way to maximize the knowledge on object appearance is to evaluate the tracked features. To achieve robust tracking in unconstrained environments, our Salient Collaborating Features Tracker (**SCFT**) discovers the most salient local features in an online manner. Every tracked local feature is considered as an elementary predictor having an individual reliability in encoding an object structural constraint, and collaborating with other features to predict the target state. To assess the reliability of a given feature, we define feature saliency as comprising three factors: *persistence*, *spatial consistency*, and *predictive power*. Thereby, the global target state prediction arises from the aggregation of all the local predictions considering individual feature saliency properties. Furthermore, the appearance change problem (which is a major issue causing drift [99]) is handled through a dynamic target model that continuously incorporates new structural properties while removing non-persistent features.

Generally, a tracking algorithm includes two main aspects: the target representation including the object characteristics, and the search strategy for object localization. The contributions of our work relate to both aspects. For target representation, our part-based model includes keypoint patches encoding object structural constraints with different levels of reliability. Part-based representations are proven to be robust to local appearance changes and partial occlusions [84, 126, 83]. Moreover, keypoint regions are more salient and stable than other types of patches (*e.g.* regular grid, random patches), increasing the distinctiveness of the appearance model [104, 27]. Regarding the search strategy, the target state estimation is carried out via local features collaboration. Every detected local feature casts a local prediction expressing a constraint on the target structure according to the spatial layout, saliency information, detection scale, and dominant orientation of the feature. In this manner, feature collaboration preserves the object structure while handling pose and scale change without requiring to analyze the relationship between keypoints like in [126], neither calculating homographies such as in most keypoint matching works [29, 77, 93].

More specifically, the main contributions of this paper are:

1. A novel method for evaluating feature saliency to identify the most reliable features based on their *persistence*, *spatial consistency*, and *predictive power*;
2. The explicit exploitation of feature saliency information in several algorithmic steps: (1) local predictions, (2) feature collaboration for global localization, (3) scale change estimation, and (4) for local feature removal from the target model;
3. A dynamic appearance model where persistent local features are stored in a pool, to

encode both recent and old structural properties of the target.

4. Extensive experimentation to evaluate the tracker performance against five recent state-of-the-art methods. The experimental work conducted on challenging videos shows the validity of the proposed tracker, outperforming the compared methods significantly.

The rest of this paper is organized as follows. In the next section, we review related part-based tracking works. Algorithm steps are presented in details in section 6.3. Experimental results are provided and analyzed in section 6.4, and section 9 concludes the paper.

6.2 Related works

Among various visual tracking algorithms, part-based trackers have attracted a great interest during the last decade. This is mainly due to the robustness of part-based models in handling partial changes, and to the efficiency of prediction methods in finding the whole target region given a subset of object parts. The fragment-based tracker of Adam *et al.* [2] is one of the pioneering methods in this trend. In their tracker, target parts correspond to arbitrary patches voting for object positions and scales in a competitive manner. The object patches are extracted according to a regular grid, and thus are inappropriate for articulated objects and significant in-plane rotations. Further, Erdem *et al.* demonstrated that the winning patch might not always provide reliable predictions [76]. This issue is addressed in [76] by differentiating the object patches based on their reliability. Therefore, every patch contributes to the target state prediction according to its reliability, allowing to achieve a better accuracy. Many other methods have been proposed for locating the object through parts tracking. The authors in [80] track object parts separately and predict the target state as a combination of multiple measurements. This method identifies inconsistent measurements in order to eliminate the false ones in the integration process. The method in [75] represents the shape of an articulated object with a small number of rectangular regions, while the appearance is represented by the corresponding intensity histograms. Tracking is then performed by matching local intensity histograms and by adjusting the locations of the blocks. Note that these last two trackers present the disadvantage of requiring manual initialization of object parts.

In [83], the appearance model includes a combination between holistic and local representations to increase the model distinctiveness. In this model, the spatial information of the object patches is encoded by a histogram representing the object structure. Similarly, Jia *et al.* sample a set of overlapped patches on the tracked object [84]. Their tracker includes an occlusion handling module allowing to locate the object using only visible patches. Kwon *et al.* [79] also used a set of local patches, updated during tracking, for target representation. The

common shortcoming of the last three trackers is the model adaptation mechanism in which the dictionary is updated simply by adding new elements, without adapting existing items. Another approach for creating part-based representations is the superpixel over-segmentation [85, 86]. In [85], Wang *et al.* use a discriminative method evaluating superpixels individually, in order to distinguish the target from the background and detect shape deformation and occlusion. Their tracker is limited to small displacements between consecutive frames, since over-segmentation is performed only for a region surrounding the target location in the last frame. Moreover, this method requires a training phase to learn superpixel features from the object and the background.

One of the major concerns in part-based tracking is to select the most significant and informative components for the appearance model. An interesting approach for defining informative components consists in using keypoint regions. Local keypoint regions (*e.g.* SIFT [89] and BRISK [127]) are more efficient than other types of patches in encoding object structure, as they correspond to salient and stable regions invariably detectable under various perturbation factors [107, 27]. Based on this, Yang *et al.* model the target with a combination of random patches and keypoints [92]. Keypoints layout is used to encode the structure while random patches model other appearance properties via their LBP features and RGB histograms. The target is thus tracked by exploiting multiple object characteristics, but the structural model captures only recent properties, as the keypoint model contains only those detected on the last frame. In a later work, Guo *et al.* [77] used a set of keypoint manifolds organized as a graph to represent the target structure. Every manifold contains a set of synthetic keypoint descriptors simulating possible variations of the original feature under viewpoint and scale change. The target is found by detecting keypoints on the current frame and matching them with those of the manifold model. This tracker achieved stable tracking of dynamic objects, at the cost of calculating homographies with RANSAC, which may be inappropriate for non-planar objects as shown in [126].

Generalized Hough Transform (GHT)-based approaches have been recently presented as an alternative to homography calculation methods. GHT was initially used in context tracking [11], where the target position is predicted by analyzing the whole scene (context) and identifying features (not belonging to the target) that move in a way that is statistically related to the target's motion. In later works, this technique has been applied to object features in order to reflect structural constraints of the target and cope with partial occlusion problems. Nebehay *et al.* [126] propose to combine votes of keypoints to predict the target center. Although every keypoint votes in an individual manner, the geometrical relationship is analyzed between each pair of keypoints in order to rotate and scale votes accordingly. Furthermore, the keypoint model is not adapted to object appearance changes, arising only

from the first observation of the target. In [14], the authors used an adaptive feature reservoir updated online to learn keypoint properties during tracking. The tracker achieved robust tracking in situations of occlusion and against illumination and appearance changes. However, this method does not handle scale changes and suffers from sensitivity to large in-plane rotations. In this paper we propose a novel tracking algorithm that exploits the geometric constraints of salient local features in a way to handle perturbation factors related to the target movement (*e.g.* scale change, in-plane and out-of-plane rotations), as well as those originating from its environment (*i.e.* occlusion, background clutter, distractors).

6.3 Proposed method

6.3.1 Motivation and overview

In our part-based model, object parts correspond to keypoint patches detected during tracking and stored in a feature pool. The pool is initialized with the features detected on the bounding box region defined in the first video frame, and updated dynamically by including and/or removing features to reflect appearance changes. Instead of detecting local features in a region with a fixed size around the target location (like in [85, 77]), we eliminate the restriction of small displacements by using the probabilistic search space reduction method proposed in [14]. This allows us to avoid computing local features on the entire image by limiting their extraction to most likely regions based on the target color distribution.

When performing target search on a given frame, features from the pool are matched with those detected on the reduced search space. Following the matching process, the geometrical constraints (of the matched features) are adapted to local scale and pose changes as explained in section 6.3.3.1. Then all the matched features collaborate in a voting-based method (section 6.3.3.2), to achieve global localization (section 6.3.3.3) and estimate the global scale change (section 6.3.3.4). Thus, the global prediction result corresponds to the aggregation of individual votes (elementary predictions). This method preserves the object structure and handles pose and scale changes, without requiring homography calculations such as in [77], neither analyzing the geometrical relationship between keypoints like in [126].

In order to keep the most relevant elements in the feature pool and exploit appropriately the most reliable predictors, each tracking iteration is followed by a saliency evaluation step. Our definition of feature saliency includes three factors: feature *persistence*, *spatial consistency*, and *predictive power*.

– The *persistence* value ω of a given feature is used to evaluate the degree of co-occurrence between the target and the keypoint, and to determine if the feature should be removed from the pool.

- The *spatial consistency* matrix Σ reflects the motion correlation between the feature and the target center in the local prediction function.
- The *predictive power* ψ indicates the accuracy of the past local predictions by comparison to the past global predictions. This value is used to weight the contribution of a local feature in the global localization function.

Note that both the *spatial consistency* and the *predictive power* are designed to assess the feature quality. On the other hand, the *persistence value* is related to the occurrence level, disregarding the usefulness of the feature. Figure 6.1 illustrates situations where non-salient features can be identified through saliency evaluation. Non-salient features may correspond to outliers included erroneously to the object model in the initialization step or when updating it. Such a feature may originate from the background as seen in figure 6.1a or belong to an occluding object (figure 6.1b) causing incorrect prediction. Once a keypoint is considered as non-salient, the corresponding local prediction (vote) will not be significant in the voting space, and/or its contribution will be reduced in the global localization procedure. Moreover the feature is likely to be removed from the pool as soon as it becomes *non-persistent*.

It should be noted that inconsistent features belonging to the tracked object may remain in the object model if they co-occur frequently with the target. An example is illustrated in figure 6.1c. However, their local predictions hardly affect the overall localization, since their quality indicators (Σ and ψ) will be reduced. While bad predictors are penalized and/or removed from the model, target global localization is carried out via a collaboration mechanism, exploiting the local predictions of the most salient features. The proposed tracking algorithm is presented in figure 6.2 and detailed in the next sections.

6.3.2 Part-based appearance model

In our tracker, the target is represented by a set of keypoint patches stored in a feature pool $\mathcal{P}$. The proposed method could use any type of scale/rotation invariant keypoint detector/descriptor. We used SIFT [89] as a keypoint detector/descriptor for its proven robustness [107]. We denote by f a feature from the pool $\mathcal{P}$. All the detected features are then stored under the form

$$f = [d, \theta, \sigma, V, Sal], \tag{6.1}$$

where:

- d is the SIFT keypoint descriptor comprising 128 elements to describe the gradient information around the keypoint position;
- θ is the detection angle corresponding to the main orientation of the keypoint;
- σ is the detection scale of the keypoint;

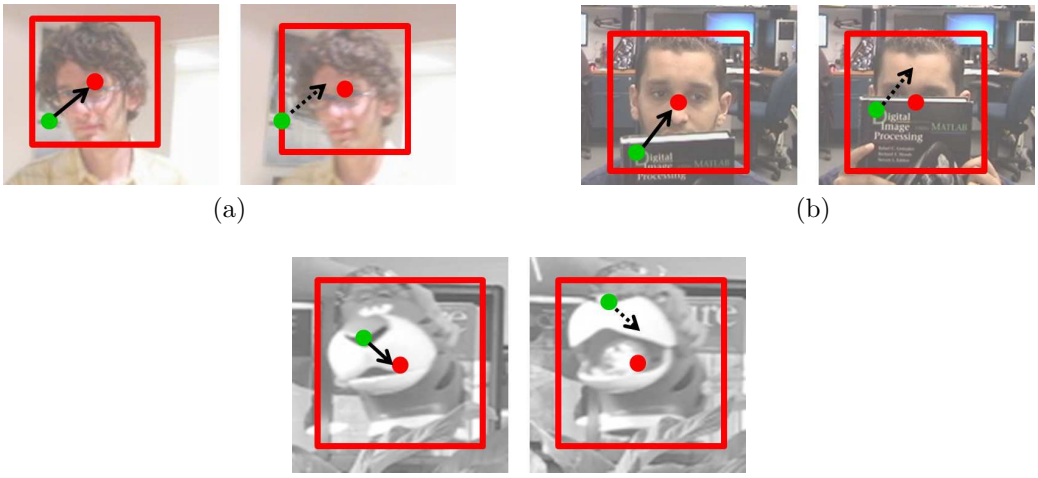


Figure 6.1 Typical situations showing that saliency evaluation allows identifying bad predictors. Red and green dots represent, respectively, the target center and the tracked feature. Continuous arrows represent the feature prediction initialization, while dotted arrows show inconsistent votes after a certain number of frames.

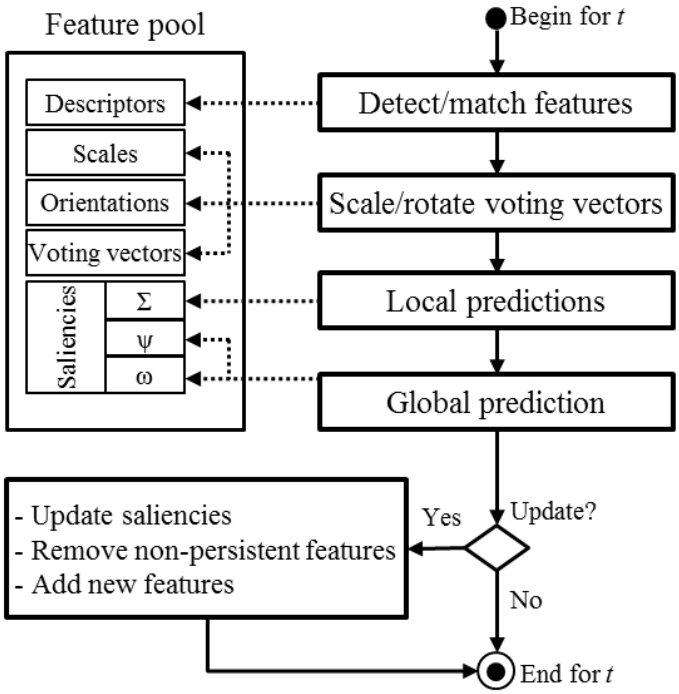


Figure 6.2 Diagram of the algorithm steps for a given frame at time t. Continuous arrows correspond to transitions between steps while dotted arrows show algorithm steps utilizing components from the appearance model.

- $V = [\delta_x, \delta_y]$ is a voting vector describing the target center location with respect to the keypoint location (see figure 6.3);
- $Sal = [\omega, \Sigma, \psi]$ is the saliency information including *persistence*, *spatial consistency*, and *predictive power* indicators.

Note that all the detection properties (*i.e.* d, θ, σ, and V) are defined permanently the first time the feature is detected, whereas saliency information (*i.e.* ω, Σ, and ψ) is updated every time features are evaluated.

6.3.3 Global collaboration of local predictors

In order to limit keypoint detection at time t to the most likely image area, we apply the search space reduction method that we previously proposed in [14]. Detected features from the reduced search space are then matched with those in the target model $\mathcal{P}$ in a nearest neighbor fashion. For matching a pair of features, we require that the ratio of the Euclidian distance from the closest neighbor to the distance of the second closest is less than an upper limit λ. The resulting subset $\mathcal{F}_t \subseteq \mathcal{P}$ contains the matched target features at time t. After the matching process, the voting vectors (of the matched features) are adapted to local scale and pose changes as explained in the following.

6.3.3.1 Voting vectors adaptation

Each feature $f \in \mathcal{F}_t$ encodes a structural property expressed through its voting vector. Before applying the structural constraint of f, the corresponding voting vector V should be scaled and rotated according to the current detection scale σ_t and dominant orientation θ_t at time t as shown in figure 6.3. This adaptation process produces the current voting vector $V_t = [\delta_{x,t}, \delta_{y,t}]$, with

$$\delta_{x,t} = \|V\|\rho_t \cos(\Delta_{\theta,t} + \text{sign}(\delta_y) \arccos \frac{\delta_x}{\|V\|}), \tag{6.2}$$

$$\delta_{y,t} = \|V\|\rho_t \sin(\Delta_{\theta,t} + \text{sign}(\delta_y) \arccos \frac{\delta_x}{\|V\|}), \tag{6.3}$$

where $\Delta_{\theta,t}$ and ρ_t are respectively the orientation angle difference and the scale ratio between the first and the current detection of f:

$$\Delta_{\theta,t} = \theta_t - \theta, \tag{6.4}$$

$$\rho_t = {}^{\sigma_t}/_{\sigma}. \tag{6.5}$$

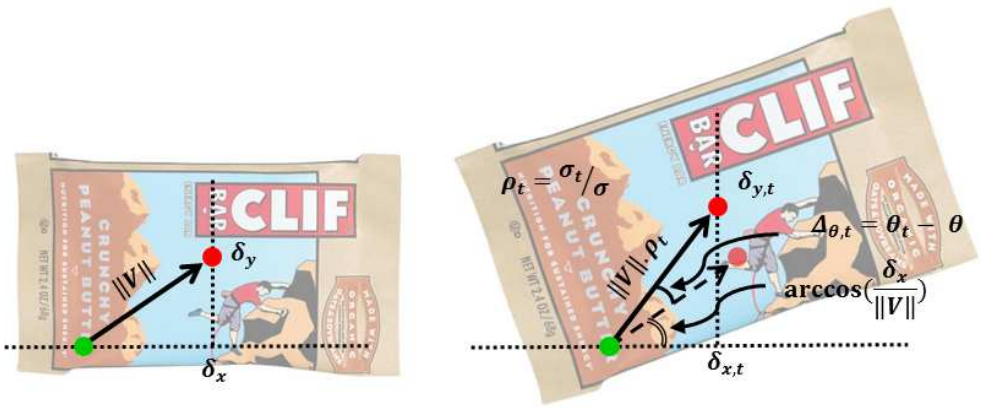


Figure 6.3 Adapting the voting vector to scale and orientation changes between the first detection frame of the feature (left) and the current frame (right). The red and green dots represent, respectively, the target center and the local feature.

6.3.3.2 Local predictions

After adapting the voting vectors to the last local changes, we base local predictions on GHT to build a local likelihood (or prediction) map $\mathcal{M}_l$ for every feature in $\mathcal{F}_t$. For f, the local likelihood map is built in the reduced search space for all the potential object positions $\mathbf{x}$ using their relative positions $\mathbf{x}_f$ with respect to the keypoint location. The local likelihood map is defined using a 2D Gaussian probability density function as

$$\mathcal{M}_l(\mathbf{x}) = \frac{1}{\sqrt{2\pi|\Sigma|}} \, exp\left(-0.5\,(\mathbf{x}_f - V_t)^\top \Sigma^{-1}(\mathbf{x}_f - V_t)\right). \tag{6.6}$$

6.3.3.3 Global localization

To achieve global prediction of the target position, features in $\mathcal{F}_t$ collaborate according to their saliency properties (*persistence* and *predictive power*). The global localization map $\mathcal{M}_g$ is thus created at time t to represent the target center likelihood considering all the detected features. Concretely, the global map is computed by aggregating local maps according to the equation

$$\mathcal{M}_{g,t}(\mathbf{x}) = \sum_{f^{(i)} \in \mathcal{F}_t}^{i} \omega_t^{(i)} \psi_t^{(i)} \mathcal{M}_{l,t}^{(i)}(\mathbf{x}). \tag{6.7}$$

The final target location $\mathbf{x}_t^*$ is then found as

$$\mathbf{x}_t^* = \arg\max_{\mathbf{x}} \mathcal{M}_{g,t}(\mathbf{x}). \tag{6.8}$$

6.3.3.4 Estimating the scale

We also exploit saliency information to determine the target size S_t at time t. Scale change estimation is carried out by using the scale ratios of the most persistent keypoints. We denote by $\mathcal{F}_t^* \subset \mathcal{F}_t$ the subset including 50% of the elements in $\mathcal{F}_t$, having the highest value of ω_t. Then we compute

$$S_t = \frac{1}{|\mathcal{F}_t^*|} \sum_{f^{(j)} \in \mathcal{F}_t^*}^{j} \rho_t^{(j)} S^{(j)} \tag{6.9}$$

to estimate the current target size, taking into account the object size $S^{(j)}$ when the j^{th} feature was detected the first time.

6.3.4 Model update

The saliency information is updated with the object model when a good tracking is achieved. Our definition of a good tracking at time t is that the matching rate τ_t in the target region exceeds the minimum rate τ_{min}. In this case saliency indicators are adapted and $\mathcal{P}$ is updated by adding/removing features.

6.3.4.1 Persistence update

If the matching rate τ_t shows a good tracking quality, the *persistence* value $\omega_t^{(i)}$ is updated for the next iteration with

$$\omega_{t+1}^{(i)} = (1-\beta)\omega_t^{(i)} + \beta \mathbb{1}_{\{f^{(i)} \in \mathcal{F}_t\}}, \tag{6.10}$$

where β is an adaptation factor and $\mathbb{1}_{\{f^{(i)} \in \mathcal{F}_t\}}$ is an indicator function defined on $\mathcal{P}$ to indicate if $f^{(i)}$ belongs to $\mathcal{F}_t$. Following this update, we remove from $\mathcal{P}$ the elements having a *persistence* value lower than ω_{min}. On the other hand, the newly detected features (in the predicted target region) are added to $\mathcal{P}$ with an initial value ω_{init}.

6.3.4.2 Spatial consistency

The *spatial consistency* Σ is a 2x2 covariance matrix considered as a quality indicator and used in the local prediction function (Eq. 6.6). Σ is initialized to Σ_{init} for a new feature. It is then updated to determine the spatial consistency between $f^{(i)}$ and the target center by applying

$$\Sigma_{t+1}^{(i)} = (1-\beta)\Sigma_t^{(i)} + \beta \Sigma_{cur}^{(i)}, \tag{6.11}$$

where the current estimate of Σ is

$$\Sigma_{cur}^{(i)} = (V_{cur}^{(i)} - V_t^{(i)})(V_{cur}^{(i)} - V_t^{(i)})^\top, \tag{6.12}$$

and $V_{cur}^{(i)}$ is the offset vector measured at time t given the global localization result. As a result, Σ decreases for consistent features, causing the votes to be more concentrated in the local prediction map. By contrast, the more this value increases during tracking (for inconsistent features), the more the votes become scattered.

6.3.4.3 Predictive power

In this step, we evaluate the predictive power of every keypoint contributing to the current localization, considering the maxima of local prediction maps, and the global maximum corresponding to the final target position. This process, that we call *prediction back-evaluation*, aims to assess how good local predictions are. The local prediction for the i^{th} feature is defined as the position

$$\hat{\mathbf{x}}_t^{(i)} = \arg\max_{\mathbf{x}} \mathcal{M}_{l,t}^{(i)}(\mathbf{x}). \tag{6.13}$$

The *predictive power* $\psi_{t+1}^{(i)}$ of $f^{(i)}$ at time $t+1$ depends on the distances between its past predictions and the corresponding global predictions. We calculate $\psi_{t+1}^{(i)}$ with the summation of a fuzzy membership function as

$$\psi_{t+1}^{(i)} = \sum_{k=1}^{t} exp(\frac{-(\hat{\mathbf{x}}_k^{(i)} - \mathbf{x}_k^*)^2}{\epsilon S_k^2})\ \mathbb{1}_{\{f^{(i)} \in \mathcal{F}_k\}} \tag{6.14}$$

where ϵ is a constant set to 0.005. The *predictive power* ψ increases as long as the feature achieves good local predictions. Consequently, the feature is considered as a reliable predictor, and its contribution in the global localization function (Eq. 6.7) becomes more prominent. We note that both Σ and ψ are designed to evaluate the feature quality. However, the former affects local predictions while the latter weights its contribution in the global localization. The overall tracking algorithm steps are presented in Alg. 4.

6.4 Experiments

6.4.1 Experimental setup

6.4.1.1 The compared trackers

We evaluated our Salient Collaborating Features Tracker (**SCFT**) by a comparison to recent state-of-the-art algorithms. Among the compared trackers, four are part-based methods

Algorithm 4 Tracking algorithm

- initialize $\mathcal{P}$
for all $frames$ **do**
  - Apply feature detector
  - Match features to get $\mathcal{F}_t \subseteq \mathcal{P}$
  for all $matched\_features$ ($f^{(i)} \in F_t$) **do**
    - Scale/rotate $V^{(i)}$: (Eq. 6.2 & 6.3)
    - Compute local likelihood map $\mathcal{M}_{l,t}^{(i)}(\mathbf{x})$: (Eq. 6.6)
    - Find local prediction result $\hat{\mathbf{x}}_t^{(i)}$: (Eq. 6.13)
  end for
  - Compute global likelihood map $\mathcal{M}_{g,t}(\mathbf{x})$: (Eq. 6.7)
  - Find global location $\mathbf{x}_t^*$: (Eq. 6.8) {output for frame t}
  - Estimate target size S_t: (Eq. 6.9) {output for frame t}
  if ($\tau_t \geq \tau_{min}$) **then**
    - Update ω_{t+1}: (Eq. 6.10)
    - Remove non-persistent features (*i.e.* $\omega_{t+1} \leq \omega_{min}$)
    for all $matched\_features$ ($f^{(i)} \in F_t$) **do**
      - update $\Sigma_{t+1}^{(i)}$ (Eq. 6.11) and $\psi_{t+1}^{(i)}$ (Eq. 6.14)
    end for
    - Add new features to $\mathcal{P}$
    - Initialize V, ω, Σ, and ψ for new features
  end if
end for

already discussed in section 5.2. These trackers are the SuperPixel Tracker (SPT) [85], the Sparsity-based Collaborative Model Tracker (SCMT) [83], the Adaptive Structural Tracker (AST) [84], and the Structure-Aware Tracker (SAT) [14]. The fifth one is the online Multiple Support Instance Tracker (MSIT) [119] using a holistic appearance model. The corresponding source codes are provided by the authors with several parameter combinations. In order to ensure a fair comparison, we tuned the parameters of their methods so that for every video sequence in our dataset, we always use the best parameter combination among the proposed ones.

6.4.1.2 Dataset

We evaluate the trackers on 10 challenging video sequences. Six of them are from prior works, publicly available with the ground truth and commonly used by the community. Figure 6.4 presents the first frame of each of the sequences. The *tiger1*, *tiger2* and *cliffbar* sequences are provided in [3] and the *David* is from [6]. The *girl* and *faceocc* sequences are respectively

from [18] and [2]. The *jp1*, *jp2*, *wdesk*, and *wbook* were captured in our laboratory room using a Sony SNC-RZ50N camera. The area is cluttered with desks, chairs, and technical video equipment in the background. The video frames are 320x240 pixels recorded at 15 fps. We manually created the corresponding ground truths for *jp1*, *jp2*, *wdesk*, and *wbook* with 608, 229, 709, and 581 frames respectively [2]. In order to better figure out the quantitative results of our tracker, we categorized the video sequences according to the main difficulties that may occur in each sequence. The categorization of the sequences according to seven main properties is presented in table 6.1. This allows us to construct subsets of videos in order to quantitatively evaluate the trackers in several situations. Note that one video sequence may present more than one difficulty.

6.4.1.3 Evaluation methodology

Sucess rate and average location error. In order to summarize a tracker's performance on a video sequence, we use the success rate and the average location error. The success rate is measured by calculating for each frame the Overlap Ratio $OR = \frac{area(P_r \cap G_r)}{area(P_r \cup G_r)}$, where P_r is the predicted target region and G_r is the ground truth target region. For a given frame, tracking is considered as a success if $OR \geq 0.5$. The Center Location Error (CLE) for a given frame consists in the position error between the center of the tracking result and that of the ground truth. The tables 6.2 and 6.3 present respectively the success rates and the average center location errors for the compared methods.

Precision plot. While the average location error is known to be useful to summarize performance by calculating the mean error over the whole video sequence, this metric may fail to correctly reflect the tracker behavior. For example, the average location error for a tracker that tracks an object accurately for almost all the sequence before losing it on the last frames could be substantially affected by large CLEs on the last few frames. To address this issue, we adopt the precision plot used in [3] and [109]. This graphic shows the percentage of frames (precision) where the predicted target center is within the given threshold distance from the ground truth center.

Success plot. By analogy to the precision plot that shows percentages of frames corresponding to several threshold distances of the ground truth, the authors in [128] argue that using one success rate value at an overlap ratio of 0.5 may not be representative. As suggested in [128], we use the success plot showing the percentages of successful frames at the ORs varied from 0 to 1.

CLE and OR plots. Two other types of plots are used in our experiments to analyze in depth the compared methods : 1) the center location error versus the frame number presented

2. Our sequences are available at http://www.polymtl.ca/litiv/en/vid/.

Figure 6.4 The annotated first frames of the video sequences used for experiments. From left to right, top to bottom: *tiger1*, *tiger2*, *cliffbar*, *David* , *girl*, *faceocc*, *jp1*, *jp2*, *wdesk*, *wbook*.

Table 6.1 Main difficulties characterizing the test sequences. LTocc: Long-Term Occlusion, Distr: presence of Distractors, BClut: Background Clutter, OPR: Out-of-Plane Rotation, Illum: Illumination change, CamMo: Camera Motion, ArtObj: Articulated Object.

video	LTOcc	Distr	BClut	OPR	Illum	CamMo	ArtObj
David					✓	✓	
girl		✓		✓			
faceocc	✓					✓	
tiger1			✓				✓
tiger2			✓				✓
cliffbar			✓				
jp1		✓					
jp2		✓					
wdesk	✓						
wbook	✓						

in figure 6.5, and 2) the overlap ratio versus the frame number presented in figure 6.6. These plots are useful for monitoring and comparing the behaviors of several trackers over time for a given video sequence. We finally note that we averaged the results over five runs in all our experiments.

6.4.2 Experimental result

6.4.2.1 Overall performance

The overall performance for several trackers is summarized by the average values in the tables 6.2 and 6.3 (last rows), as well as the average precision and success plots for the whole dataset (figure 6.7). All the metrics used for overall performance evaluation demonstrate that our proposed method outperforms all the other trackers, achieving an average success rate of 90.94% and an average localization error lower than 10 pixels. A major advantage of using success and precision plots is to allow choosing the appropriate tracker for a specific situation given the application requirements (*e.g.* high, medium, or low accuracy). In our experiments, the success and precision curves show the robustness of **SCFT** for all application requirements. **SCFT** is also the only tracker to reach 80% in precision for an error threshold of 15 pixels, and to produce a success rate exceeding 60% when the required OR is 80%. Except for SAT that realized the second best overall performance, and MSIT that had the last rank, the rankings of the other trackers are different depending on the considered metric. In the following subsections, the experimental results are organized and analyzed by categories according to the table 6.1.

Table 6.2 Percentage of correctly tracked frames (success rate) for **SCFT** and the five other trackers. **Bold red** font indicates best results, *blue italics* font indicates second best.

video	SPT	SCMT	AST	MSIT	SAT	**SCFT**
David	62.37	60.22	37.63	*63.44*	**100**	**100**
girl	84.16	1.98	17.82	0.99	*84.95*	**85.94**
faceocc	5.62	**100**	25.84	80.90	99.55	*99.89*
tiger1	*60.56*	25.35	30.99	2.82	50.99	**80.28**
tiger2	46.27	16.42	31.34	5.97	*70.15*	**75.74**
cliffbar	51.52	24.24	*69.70*	7.58	60.30	**77.27**
jp1	18.09	78.13	84.38	3.78	*89.14*	**99.41**
jp2	39.30	55.02	55.02	16.59	*93.80*	**97.03**
wdesk	13.68	57.26	32.30	10.01	*90.47*	**93.96**
wbook	98.80	**100**	99.83	8.95	99.86	*99.90*
average	48.04	51.86	48.48	20.10	*83.92*	**90.94**

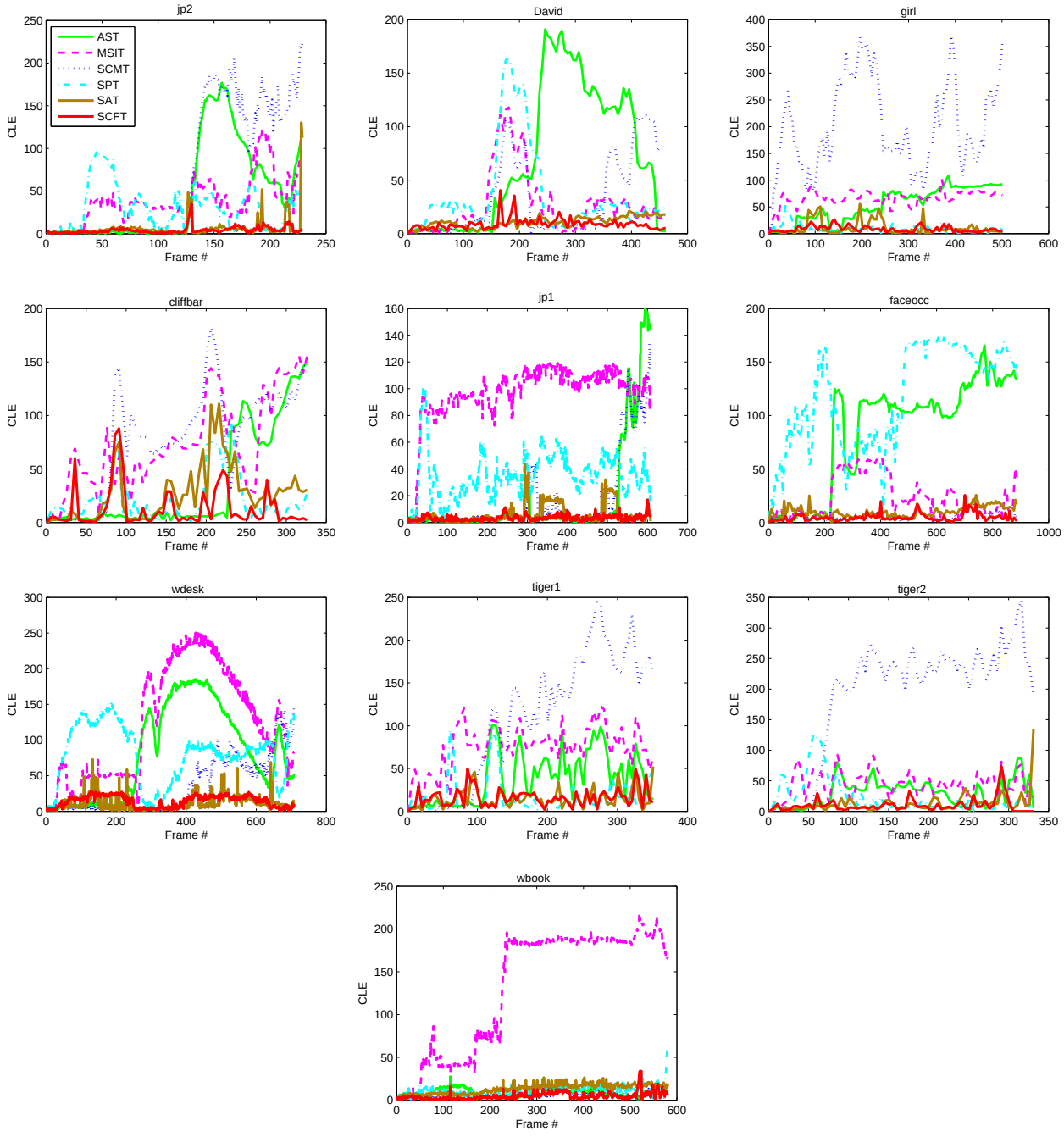


Figure 6.5 Center location error plots.

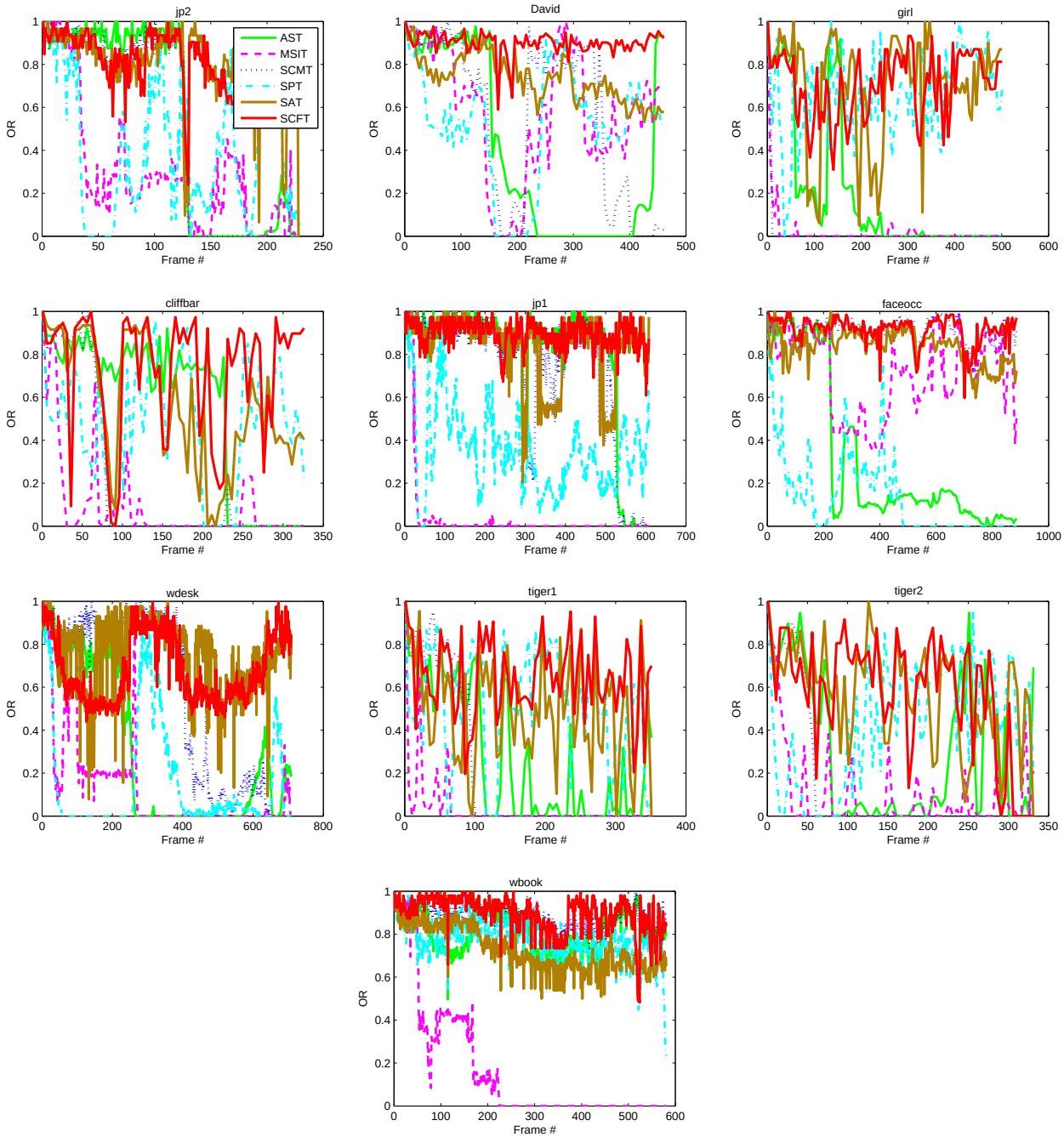


Figure 6.6 Overlap ratio plots.

Table 6.3 Average location errors in pixels for **SCFT** and the five other trackers. **Bold red** font indicates best results, *blue italics* font indicates second best.

video	SPT	SCMT	AST	MSIT	SAT	**SCFT**
David	36.09	33.81	68.57	26.71	*10.48*	**9.96**
girl	**8.97**	201.27	53.42	66.15	10.01	*9.29*
faceocc	116.84	**5.07**	85.43	23.36	14.26	*5.58*
tiger1	17.14	107.74	38.06	74.86	**14.91**	*15.65*
tiger2	22.81	189.50	29.15	44.58	*16.13*	**10.25**
cliffbar	*22.11*	77.31	35.35	73.72	25.33	**13.67**
jp1	35.21	17.74	16.66	97.08	*7.03*	**4.75**
jp2	30.58	69.44	45.15	39.47	*7.25*	**4.21**
wdesk	79.92	34.17	80.97	122.62	**11.12**	*14.31*
wbook	11.27	**5.09**	8.68	131.57	11.87	*5.91*
average	38.09	74.11	46.14	70.01	*12.84*	**9.36**

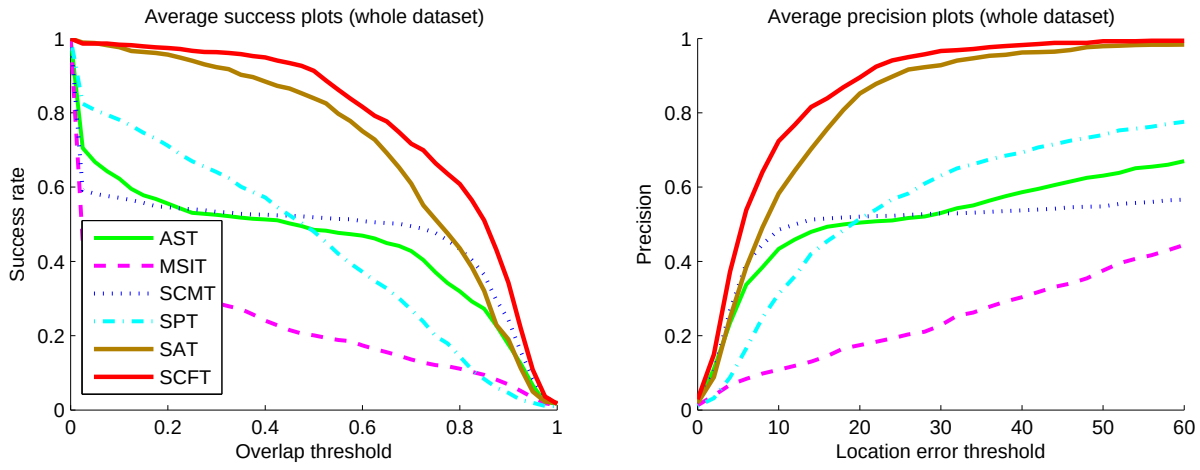


Figure 6.7 Average success and average precision plots for all the sequences.

6.4.2.2 Long-term occlusion

We evaluated the six methods in face tracking under long-term partial occlusion (up to 250 consecutive frames). In the *faceocc* and *wbook*, the tracked face remains partially occluded by an object several times for a long period. Some trackers drift away from the target to track the occluding object, which is mainly due to appearance model contamination by features belonging to the occluding object. Our method was able to track the faces successfully in almost all the frames under severe occlusion. The local predictions of a few detected features were sufficient for **SCFT** to achieve an accurate global prediction. Our target model may erroneously include features from the occluding object, but since we evaluate their motion consistency and predictive power, the corresponding local predictions will be scattered in the voting space and have small weights in the global localization function. The error plots for *faceocc* shows that SCMT and SAT also achieved good performances when the target was occluded (*e.g.* between frames 200 and 400). In fact, SCMT and SAT are also designed to handle occlusions, respectively through a scheme considering unoccluded patches, and a voting-based method that predicts the target center.

In the *wdesk* sequence, the tracked face undergoes severe partial occlusions while moving behind a desk. **SCFT**, SAT and SCMT track the target correctly until frame #400 where the person performs large displacements causing SCMT to drift away from the face. Both **SCFT** and SAT continue the tracking successfully while the tracked person hides behind a desk, and our method achieved the best success rate of 93.96%.

The success plots of long-term occlusion videos for **SCFT** and SAT show that both trackers can achieve almost 100% success rate as long as the required overlap ratio is lower than 0.5. Both trackers also had the two best precision curves, but **SCFT** performed significantly better under high requirement in accuracy (*i.e.* location error threshold lower than 15 pixels). As expected, the success and precision curves of MSIT are located below the others, since the holistic appearance model is not effective for a target undergoing severe occlusions.

6.4.2.3 Presence of distractors

The third and fourth rows of figure 6.9 present results of face tracking in moderately crowded scenes (four persons). In this experiments, our goal is to test the distinctiveness of the trackers. The success and precision plots for this category clearly show that **SCFT** and SAT are ranked respectively first and second regardless of the application requirements. This is mainly explained by the use of SIFT features that are proven to be effective in distinguishing a target face among a large number of other faces [117, 116, 118].

In the *jp1* video, we aim to track a face in presence of three other distracting faces,

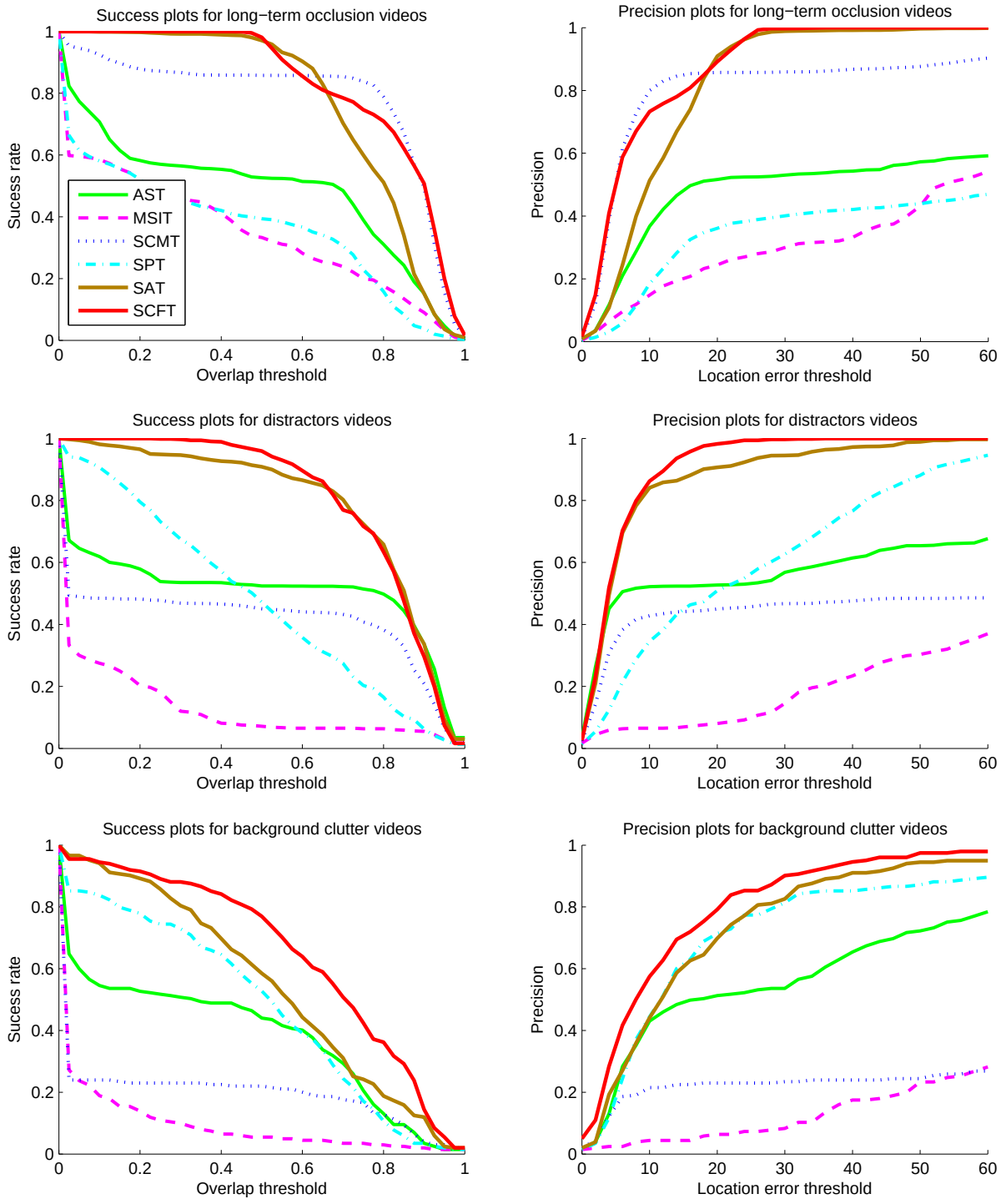


Figure 6.8 Success and precision plots for long-term occlusion, distractors, and background clutter videos.

moving around the target and partially occluding it several times. The corresponding OR and CLE plots show that the proposed **SCFT** method produces the most stable tracking at the lowest error during almost all the 608 video frames. Although the success rates of 89.14%, 84.38%, and 78.13% respectively for SAT, AST, and SCMT indicate good performances, the last two trackers drift twice (first at frame #530 and a second time at frame #570) to track distracting faces occluding or neighboring the target. We can also see in the OR and CLE plots that SAT drifts considerably three times, especially between frames #341 and #397 when the tracked face region (person with a black t-shirt in the middle of the scene) is mostly occluded. However, neither the presence of similar objects near the target nor partial occlusion situations affected our **SCFT** tracker. The high performance of the proposed method in these situations is due to the distinctiveness of SIFT keypoints, in addition to the reliance on local predictions of the most salient features, even if outliers (from the background, neighboring or occluding faces) can be present in the feature pool.

In the *jp2* video, we track a walking person in the presence of four other randomly moving persons. The target crosses in front or behind distractors that may occlude it completely for a short period. All the five other methods confused the target with an occluding face, at least for a few frames after full occlusion. Nevertheless, **SCFT** is able to recover tracking correctly as soon as a small part of the target becomes visible. For both distractors sequences *jp1* and *jp2*, **SCFT** produced simultaneously the highest success rate and the lowest average error.

6.4.2.4 Illumination change, camera motion

The video sequence *David* is recorded using a moving camera, following a walking person. The scene illumination conditions change gradually as the person moves from a dark room to an illuminated area. The face also undergoes significant pose change during movement. All the trackers, except AST, were able to track the face successfully in more than 60% of the frames. Once again, **SCFT** achieved the best success rate and the lowest average error. This experiment shows the efficiency of our appearance model, allowing the tracker deal robustly with illumination variation. Our method is also not affected by large and continuous camera motion since features are detected wherever the space reduction method shows a significant likelihood of finding the target. On the other hand, in-plane rotations are handled efficiently in the global prediction function since we exploit the information on keypoint local orientation changes.

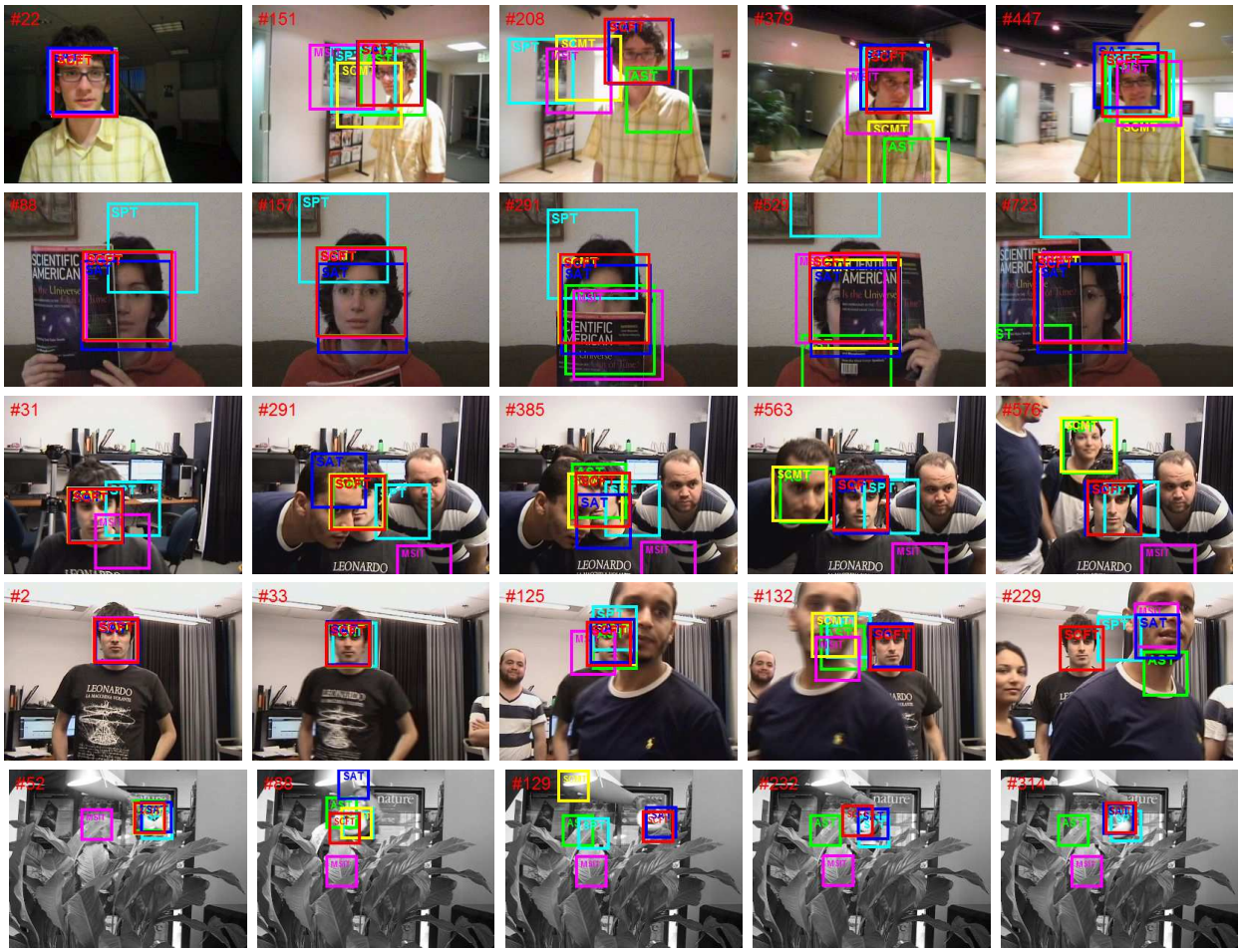


Figure 6.9 Tracking results for several trackers on the video sequences *David*, *faceocc*, *jp1*, *jp2*, and *tiger1* (from top to bottom).

6.4.2.5 Out-of-plane rotation

The target person's face in the *girl* video, exhibits pose change and out-of-plane rotations abruptly. SPT, SAT, and **SCFT** were able to track the face correctly in more than 80% of the frames. **SCFT** achieved the best success rate, handling efficiently pose change and partial occlusion. Our tracking was accurate as long as the girl's face was at least partly visible. We lost the target when the face was turned away from the camera, but we were able to recover tracking quickly as soon as it partially reappeared.

6.4.2.6 Background clutter, articulated object

The main difficulty with the *cliffbar*, *tiger1*, and *tiger2* videos is the cluttered background whose the appearance may disrupt the tracker. For this category, the success and precision curves of **SCFT** are located above the others, showing the advantage of our method for all the tested thresholds of OR and CLE. Always based on the success and precision plots, we can see that SAT and SPT performed well in this category. The precision results of SAT and SPT are almost identical when the location error threshold is lower than 30 pixels. However, SAT performs slightly better for other requirements of overlap and location error threshold. It is noteworthy that both methods include discriminative aspects facilitating tracking under such conditions. In fact, SPT uses a discriminative appearance model based on superpixel segmentation while SAT utilizes information on the background color distribution to evaluate the tracking quality.

In the *Cliffbar* sequence, a book is used as a background having a similar texture to that of the target. **SCFT** outperformed significantly all the competing methods in both success rate and average location error. AST, SAT, and SPT also performed relatively well, taking into account the difficulty of the sequence. Indeed, the target undergoes abrupt in-plane rotations and drastic appearance change because of high motion blur. The proposed tracker is hardly affected by these difficulties since it continues adapting the appearance model by including/removing keypoints, and handling pose change through keypoint orientations.

In the *tiger1* and *tiger2* sequences, the target exhibits fast movements in a cluttered background with frequent occlusions. Owing to partial predictions that localize the target center using a few visible keypoints, **SCFT** had the highest percentages of correct tracks for both videos. SAT also overcomes the frequent occlusion problem via its voting mechanism that predicts the target position from available features. The other methods fail to locate the stuffed animal, but SPT had relatively better results due to its discriminative model facilitating the distinction between target superpixels and background superpixels. Note that the tracked object in *tiger1* and *tiger2* is a deformable stuffed animal. The predictions of

features located on articulated parts are consequently inconsistent with the overall consensus, but this issue is effeciently handled by the use of *spatial consistency* and *predictive power* that reflect the predictors' reliability. These features may remain in $\mathcal{P}$ and continue predicting the target position without affecting the global result (because of low *predictive power* and *spatial consistency*). Our feature pool may also erroneously include outliers from the background, identified as non-persistent to be removed from the model.

6.4.2.7 Sensitivity to the number of features

One of the most challenging situations encountered in our dataset is the partial occlusion. The target faces in the *faceocc*, *wdesk*, and *wbook* videos undergo severe long-term occlusions causing the number of detected features to decrease drastically. Since local features detection represents a critical component for part-based trackers, we propose to study the impact of the number of features on SCFT's performance. We considered the long-term occlusion videos and analyzed the number of detected features on every video frame. We computed the average CLE value for each subset of frames having their numbers of collaborating features within the same interval (spanning 10 values). This allows us to create a scatter plot representing the average CLE versus the number of collaborating features (figure 6.10). To investigate the relationship between the number of features and the CLE, we model the plot by fitting a fourth degree predictor function and a linear function. The plot shows that the smallest numbers of features produce an average CLE not exceeding nine pixels. After that, the fitted fourth degree function decreases before stabilizing around the mean value of four pixels when more than 30 features are detected. Regarding the linear function ($y = ax + b$), it is obvious to expect that the coefficient a would be negative since the CLE becomes lower when the number of features increases. However, a high absolute value for a would suggest that the algorithm requires a large number of features to achieve accurate tracking. In our case, the linear coefficients estimation ($a = -0.0064$; $b = 5.1107$) demonstrate that the error barely increases when the number of collaborating features diminishes from the maximum (*i.e.* 345 features) to one feature. This ascertainment confirms that the collaboration of a few number of unoccluded features is sufficient for our tracker to ensure accurate tracking.

6.5 Conclusion

This paper proposes a novel and effective part-based tracking algorithm, based on the collaboration of salient local features. Feature collaboration is carried out through a voting method where keypoint patches impose local geometrical constraints, preserving the target structure while handling pose and scale changes. The proposed algorithm uses saliency evalu-

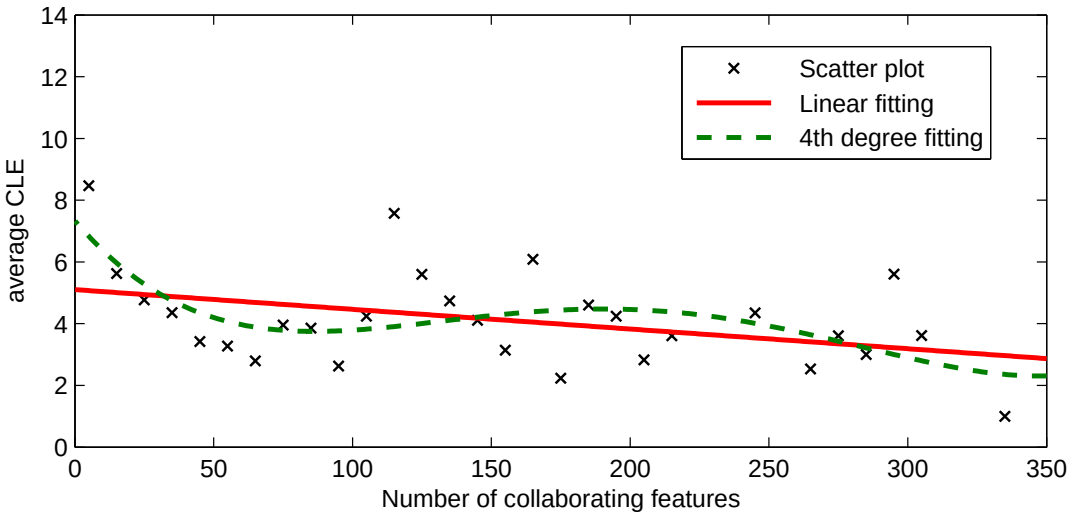


Figure 6.10 Sensitivity of SCFT's localization error (in pixels) to the number of collaborating features (long-term occlusion videos). Data points from the scatter plot correspond to interval centers.

ation as a key technique for identifying the most reliable and useful features. Our conception of feature saliency includes three elements: *persistence*, *spatial consistency*, and *predictive power*. The *persistence* indicator allows to eliminate outliers (*e.g.* from the background, or an occluding object) and expired features from the target model, while the *spatial consistency* and the *predictive power* indicators penalize predictors that do not agree with past consensus. The experiments on publicly available videos from standard benchmarks show that SCFT outperforms state-of-the-art trackers significantly. Moreover, our tracker is insensitive to the number of tracked features, achieving accurate and robust tracking even if most of the local predictors are undetectable.

CHAPITRE 7

RÉSULTATS COMPLÉMENTAIRES

Ce chapitre complète le chapitre 5 par des résultats expérimentaux publiés dans [14]. Dans le but d'évaluer l'efficacité du modèle structurel utilisé dans SAT, nous avons réalisé des expériences comparant la méthode proposée à une version intermédiaire nommée no-SAT, n'utilisant pas de contraintes structurelles. Identiquement à SAT, l'algorithme no-SAT procède avec la réduction de l'espace de recherche par le même filtre de particules. Le modèle d'apparence par parties consiste au même réservoir adaptatif de points caractéristiques de SAT, mais sans inclure les propriétés structurelles des points. Suite à l'appariement entre les points du réservoir et ceux détectés dans l'espace de recherche réduit, no-SAT prédit la position de la cible comme étant la particule avec le plus grand nombre de points appariés. Les expériences ont été réalisées sur les séquences *jp1*, *jp2*, *wdesk* et *wbook* en utilisant les mesures de précision (P) et d'erreur de localisation du centre (E) telles que définies dans les chapitres 5 et 6.

Chaque ligne de la figure 7.1 montre les résultats de suivi sur trois trames clés d'une séquence vidéo. Sur la séquence *jp1*, la principale difficulté consiste à suivre le visage cible en le distinguant des autres visages pouvant l'occulter. Cette expérience montre que les deux méthodes ne confondent pas la cible avec les distracteurs grâce à la distinctivité des points SIFT. Nous avons aussi remarqué que l'erreur de localisation de no-SAT augmente avec l'occultation partielle de la cible (deuxième et troisième image de la première ligne). Toutefois, SAT continue à prédire la position exacte du visage en appliquant les contraintes structurelles des points SIFT détectés sur la partie visible du visage.

La séquence vidéo *jp2* permet d'évaluer les AS dans le suivi du visage d'une personne se déplaçant dans une scène moyennement chargée (quatre personnes). Durant son déplacement, l'individu suivi croise les trajectoires des autres personnes qui l'occultent pour une courte durée. Tel que montré dans la figure 7.1, les deux AS suivent la cible avec succès tant qu'elle est visible. L'occultation totale est détectée dès que le visage suivi est complètement caché, car les points caractéristiques du modèle de référence ne peuvent être appariés sur la trame courante. Dans ce cas, les deux AS continuent la recherche en se basant sur les caractéristiques globales de couleurs, sans mettre à jour le modèle d'apparence. Le suivi peut être récupéré aussitôt qu'une partie du visage devient visible et que l'appariement redevient possible.

Dans la séquence vidéo *wdesk*, nous testons la capacité de l'AS à gérer l'occultation partielle d'une cible en mouvement. La visage suivi effectue des mouvements latéraux tout

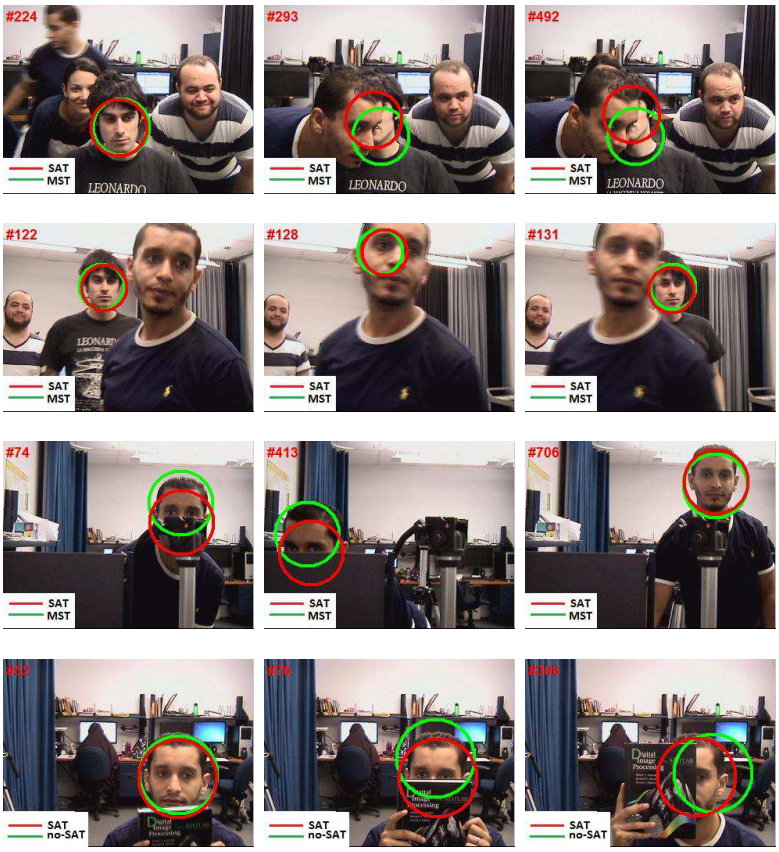


Figure 7.1 Aperçus des résultats de suivi en appliquant SAT (rouge) et no-SAT (vert) sur les séquences *jp1, jp2, wdesk* et *wbook*.

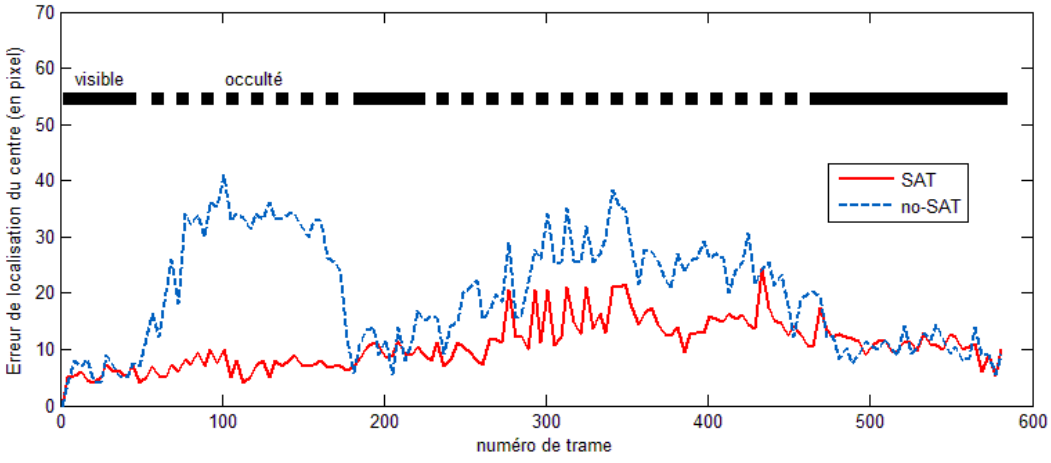


Figure 7.2 Comparaison des erreurs de position du centre entre SAT et no-SAT sur la séquence *wbook* : le suivi devient plus précis en appliquant les contraintes structurelles, notamment lorsque la cible est partiellement occultée.

Tableau 7.1 Mesures de la Precision P (en pourcentage) et de l'Erreur moyenne E (en pixel) des versions no-SAT et SAT.

	jp1		*jp2*		*wdesk*		*wbook*	
	P	E	P	E	P	E	P	E
no-SAT	85	9	94	10	70	14	71	20
SAT	**89**	**8**	**97**	**5**	**83**	**10**	**98**	**11**
amélioration	4	1	3	5	7	4	27	9

en étant partiellement caché derrière des structures de l'arrière-plan. Malgré l'occultation, nous observons que SAT continue à prédire la position exacte du visage, pendant que no-SAT dérive partiellement. Cette expérience souligne le rôle du modèle structurel de SAT dans la correction de la localisation lorsque l'occultation dure plusieurs trames.

Dans la séquence *wbook*, la personne cible cache son visage partiellement par un livre. Cette expérience démontre la haute précision de SAT qui surpasse considérablement no-SAT dans la gestion des occultations partielles de longue durée. La figure 7.2 détaille le résultat des deux algorithmes pour la séquence *wbook*. Sur ce graphique nous remarquons que l'erreur de localisation de no-SAT augmente largement durant l'occultation partielle (entre les intervalles de trames de 50 à 180 et de 230 à 470). Cependant, les occultations de longue durée n'affectent pas la stabilité de SAT qui réalise une précision de 98% (contre 71% pour no-SAT). Le tableau 7.1 présente les résultats complets des mesures de précision et d'erreur moyenne de localisation sur les quatre séquences vidéos.

CHAPITRE 8

DISCUSSIONS GÉNÉRALES

8.1 Modèle holistique ou modèle par parties ?

Les modèles d'apparence holistiques permettent de calculer les caractéristiques visuelles de l'objet d'une manière globale dans la région d'intérêt. Le principal avantage de ces modèles réside dans la complexité de calcul réduite par rapport aux méthodes décomposant l'objet en parties. Toutefois, les modèles holistiques sont inefficaces dans les situations d'occultation partielle et peu robustes face aux distracteurs. Dans le premier algorithme proposé dans cette thèse (chapitre 4), nous avons limité l'utilisation du modèle holistique à la première étape de la prédiction. À cet effet, nous avons utilisé un filtre de particules basé sur la distribution de couleur de l'objet pour déterminer grossièrement la région d'intérêt. Grâce à l'estimation grossière, les points caractéristiques SIFT sont détectés dans un espace de recherche réduit, sans avoir à analyser l'image entière. La prédiction finale repose ensuite sur l'appariement des points caractéristiques constituant le modèle par parties. L'appariement des points SIFT permet une estimation plus fine et plus précise de la cible en se basant sur les détails locaux de l'objet. La représentation par parties augmente la robustesse du modèle en évitant le problème de dérive de l'AS dans des scénarios difficiles :

- Dans le cas d'une occultation partielle, le modèle par parties permet de réaliser des appariements partiels afin de localiser la cible sans dériver sur l'objet occultant ou l'arrière-plan ;
- Lors d'une occultation totale ou d'une sortie du champ de vision, l'invisibilité totale de l'objet est détectée, vu l'impossibilité d'effectuer des appariements locaux, ce qui évite (1) la dérive de l'AS sur une région similaire à la cible (en terme de caractéristiques globales) et (2) la mise à jour erronée du modèle d'apparence ;
- En présence de distracteurs, la distinctivité des points caractéristiques évite la dérive sur les régions dont l'apparence globale est similaire à celle de la cible.

D'une manière générale, les modèles par points caractéristiques fournissent une représentation distinctive, adaptative et robuste face aux occultations. L'algorithme de suivi nécessite cependant un mécanisme externe de détection, une méthode de calcul de descripteurs locaux et une procédure d'appariement souvent complexe. Nous avons adopté un modèle holistique dans une approche probabiliste (filtre de particules) pour limiter considérablement la région de détection (et par conséquent le nombre de descripteurs locaux à calculer) en simplifiant la

procédure d'appariement des caractéristiques locales. Nous concluons que l'utilisation d'un modèle holistique pour la réduction de la région d'extraction de caractéristiques locales est une alternative intéressante aux techniques les plus courantes telles que la recherche exhaustive et la considération d'une sous-fenêtre de recherche arbitraire autour de la dernière position de l'objet. Les algorithmes de suivi présentés dans les chapitres 5 et 6 utilisent le même modèle holistique de couleurs dans une étape de prétraitement pour limiter efficacement la région de recherche de l'objet.

8.2 Mise à jour du modèle d'apparence

Les changements d'apparence de la cible peuvent surgir fréquemment durant le suivi. Parmi les facteurs qui sont à l'origine de ces changements, on retrouve le changement du point de vue de la caméra, la rotation tridimensionnelle de l'objet, la déformation d'un objet articulé et les variations des conditions d'éclairage. Afin de tenir compte des changements d'apparence, il est nécessaire de prévoir une procédure de mise à jour du modèle en répondant à deux questions lors de la conception de l'AS :

1. quand doit-on mettre à jour le modèle ?
2. comment faut-il le mettre à jour ?

La réponse à la première question nous amène à mettre au point une procédure d'évaluation de la qualité de suivi. De cette manière, le modèle d'apparence n'est adapté que lorsque la procédure d'évaluation montre une qualité de suivi acceptable. L'évaluation de qualité évite ainsi de corrompre le modèle par des caractéristiques en provenance de l'arrière-plan ou d'un objet occultant. Chacun des algorithmes de suivi proposés dans cette thèse est muni d'une procédure d'évaluation appropriée :

1. L'AS présenté dans le chapitre 4 utilise le taux d'appariement des points caractéristiques du modèle en tant que critère d'évaluation ;
2. SAT (chapitre 5) utilise une approche bayésienne basée sur un histogramme filtré qu'on rétro projette sur la région d'intérêt prédite ;
3. SCFT (chapitre 6) évalue le taux d'appariement des points caractéristiques détectés dans la région prédite à la trame courante.

La première procédure d'évaluation correspond à un simple calcul du rapport entre le nombre de points SIFT appariés et le nombre total des points caractéristiques du modèle. Ce rapport ne peut être adopté dans SAT car ce dernier utilise un réservoir dynamique de caractéristiques, constitué d'un nombre variable d'éléments. Nous avons, par conséquent, conçu la méthode bayésienne qui a permis une évaluation fiable de la qualité de suivi tout en étant

indépendante du réservoir de caractéristiques. Finalement, nous avons simplifié la procédure dans SCFT en basant l'évaluation sur le rapport entre le nombre de caractéristiques appariées et le nombre total des caractéristiques détectées sur la région prédite. Cette méthode est considérablement moins complexe que la procédure (2) de SAT (qui revient à classifier tous les pixels de la cible individuellement) tout en étant autant efficace. Notons que le rapport utilisé dans SCFT est différent de celui de la procédure (1), étant donné qu'il utilise le nombre de points SIFT détectés sur la région d'intérêt prédite au lieu du nombre d'éléments total du modèle.

La deuxième question concerne la procédure de mise à jour proprement dite et notamment les éléments à inclure au modèle. Certains modèles d'apparence tiennent compte seulement des observations récentes de la cible, ce qui peut causer la dérive de l'AS suite à une mise à jour erronée. Nous avons rencontré ce problème dans la première partie de ce travail (chapitre 4) où le modèle est constitué seulement des points SIFT de la dernière trame. D'autres méthodes apprennent les changements à partir d'exemplaires anciens de l'objet [41, 129], ce qui n'est pas assez adaptatif aux changements rapides d'apparence. Afin d'établir le compromis «stabilité/flexibilité» du modèle, les deux méthodes proposées SAT et SCFT utilisent des modèles adaptatifs qui incluent à la fois des caractéristiques anciennes et récentes de la cible. Les points SIFT et leurs propriétés structurelles sont extraits à différents moments du cycle de vie de l'objet. Le compromis «stabilité/flexibilité» est garanti par les indicateurs de persistance qui gèrent l'élimination des caractéristiques non persistantes.

8.3 Suivi orienté contexte : de la structure de la scène à la structure de l'objet

Les méthodes de suivi orientées contexte détectent les éléments auxiliaires de la scène ayant des mouvements corrélés avec ceux de la cible. Comme discuté dans la section 2.3, il s'agit d'une tâche complexe qui n'est efficace que dans certaines situations spécifiques. Dans ce travail, nous nous sommes inspirés de l'approche orientée contexte pour concevoir des méthodes orientées cible encodant la structure de l'objet. D'une manière similaire à l'exploitation des relations géométriques entre l'objet et les éléments auxiliaires (ou les supporteurs) dans le paradigme orienté contexte, nous avons exploité les relations spatiales entre les points SIFT de l'objet et le centre de ce dernier. La structure interne de l'objet est ainsi encodée à travers les position des points SIFT par rapport au centre de la cible. Notre idée de représenter les propriétés structurelles des points caractéristiques individuellement émane de deux motivations principales :

1. L'état de la cible peut ainsi être estimé en accumulant les votes individuels des caractéristiques locales par la transformée généralisée de Hough (GHT), en évitant des

méthodes plus complexes, tels que l'analyse des relations géométriques entre les paires de points [126] ou le calcul d'homographies par la méthode itérative RANSAC [77] ;

2. La mise à jour du modèle peut s'effectuer d'une manière très simple, en ajoutant les nouveaux points SIFT, en supprimant les points expirés ou en modifiant les propriétés de points existants, sans avoir à mettre à jour une structure de graphe qui représente les relations spatiales entre les caractéristiques locales comme dans [130] et [77].

8.4 Gestion des distracteurs

Pour éviter de confondre la cible avec les objets similaires, certains auteurs proposent de gérer les distracteurs explicitement en les traquant en même temps que l'objet [96]. Dans notre travail, nous avons géré les distracteurs implicitement, à travers un modèle d'apparence distinctif, sans transformer le problème de suivi d'un seul objet en celui du suivi multi objet. Les algorithmes proposés ont été testés dans différents scénarios de suivi de visage en présence de plusieurs personnes dans la scène. Tandis que tous les autres AS testés ont dérivé, au moins une fois, sur un distracteur ou sur l'arrière-plan, les méthodes proposées (incluant la version intermédiaire no-SAT) étaient capables de distinguer le visage cible parmi les distracteurs durant tous les tests. Cette constatation confirme la supériorité des caractéristiques SIFT par rapport aux modèles par patchs en terme de distinctivité.

8.5 Saillance des caractéristiques locales

Dans la littérature de la vision par ordinateur, les points caractéristiques sont souvent définis comme étant les régions les plus saillantes et les plus importantes de l'objet. Notre thèse soutient que dans une application de suivi d'objets, certains points caractéristiques peuvent être plus saillants que d'autres. En effet, notre évaluation de la saillance d'un point caractéristique dépend de sa répétabilité et de sa capacité à prédire avec précision l'état de la cible. Ce travail présente ainsi la saillance (dans le contexte du suivi d'objets) comme une notion composée de trois facteurs : la persistance du point, sa consistance spatiale et son pouvoir prédictif.

Les trois facteurs de saillance sont évalués et mis à jour durant le suivi pour toutes les caractéristiques locales. Lors de la prédiction, chaque point caractéristique est exploité selon ses propriétés de saillance. Par exemple, un point ayant un pouvoir prédictif important et une consistance spatiale élevée aurait une contribution éminente dans la localisation de la cible. En revanche, un point persistant qui se trouve sur une partie articulée en mouvement pourrait contribuer à l'estimation de l'échelle, mais ne participerait pas dans la localisation du centre de l'objet vu son instabilité. Il est à noter que la procédure d'évaluation de saillance

proposée dans ce travail peut être adaptée ou appliquée directement pour une large gamme d'algorithmes de suivi basés sur le vote des caractéristiques locales.

8.6 Exploitation optimale des descripteurs locaux

Bien que tous les modèles d'apparence par parties proposés dans ce travail reposent sur les points d'intérêt, les algorithmes développés se différencient par leurs degrés d'exploitation des descripteurs locaux des points caractéristiques SIFT. Rappelons qu'un descripteur local SIFT traduit numériquement les propriétés du point caractéristique dans quatre composantes :

1. les informations sur le gradient dans le voisinage du point, résumées par un histogramme d'orientation de gradient de 128 éléments ;
2. les coordonnées cartésiennes (x, y) du point caractéristique dans l'image ;
3. l'orientation dominante du gradient ;
4. le facteur d'échelle caractéristique de détection.

L'AS proposé dans le chapitre 4 utilise uniquement les histogrammes d'orientation de gradient locaux (première composante du descripteur) pour apparier les points SIFT et sélectionner la région candidate ayant le plus haut score d'appariement. Nous avons proposé l'algorithme SAT (chapitre 5) qui exploite, en plus des histogrammes de gradients, la disposition spatiale des points (deuxième composante) pour encoder la structure interne de l'objet. Nous avons finalement présenté la méthode SCFT (chapitre 6) qui exploite toutes les composantes des descripteurs locaux d'une façon plus optimale que les deux méthodes précédentes. Outre l'utilisation des histogrammes de gradient et de la disposition spatiale des points, SCFT considère les orientations dominantes des points SIFT (troisième composante) pour tenir compte des rotations bidimensionnelles et analyse les échelles de détection (quatrième composante) pour estimer un éventuel changement de taille de l'objet.

Remarquons que Nebehay *et al.* ont récemment proposé une nouvelle méthode de suivi par appariement de points caractéristiques [126]. Leur méthode estime les rotations bidimensionnelles en mesurant les changements angulaires entre des paires de points caractéristiques. D'autre part, les changements d'échelle sont calculés en se basant sur les variations des distances entre paires de points. Notre travail démontre que l'exploitation optimale des descripteurs locaux permet un suivi précis qui tient compte des rotations bidimensionnelles et des changements d'échelle, en évitant d'analyser les relations géométriques directes entre les points caractéristiques.

8.7 Limites des méthodes proposées

Les algorithmes proposés s'appuient sur un mécanisme externe de détection pour localiser les points d'intérêt. Bien que dans la section 6.4.2.7 nous avons démontré la faible sensibilité de l'erreur de localisation de SCFT au nombre de caractéristiques locales détectées, la précision de suivi dépend toujours du nombre de points caractéristiques sur l'objet. Les trois méthodes proposées ne peuvent utiliser les points SIFT si l'objet cible n'est pas assez texturé, ou s'il n'est pas suffisamment proche de la caméra pour que ses détails soient visibles. À titre d'exemple, nous avons vérifié expérimentalement que l'application de suivi de visage requiert une distance inférieure ou égale à dix mètres entre la personne suivie et la caméra. En effet, cette distance permet de détecter entre deux et quatre points SIFT dans la plupart des scénarios.

Si le nombre de points caractéristiques est très faible, le risque de dérive de l'AS devient important suite à un faux appariement. Cette contrainte fonctionnelle représente une limite commune pour les méthodes par points caractéristiques. Dans les trois algorithmes présentés dans cette thèse, le suivi par appariement n'est exécuté que si un nombre minimum de trois points clés sont appariés. Dans le cas contraire, les méthodes proposées appliquent le filtre de particules utilisé pour la réduction de l'espace de recherche pour prédire la position de la cible. Il est donc possible de poursuivre le suivi, mais en se basant uniquement sur la distribution globale des couleurs de l'objet.

Une autre limite pourrait découler de l'utilisation d'un nombre réduit de particules pour limiter la région de recherche dans l'étape de prétraitement. En effet, si le déplacement de la cible est très rapide entre deux trames consécutives, ou que le taux de trame est très faible (p. ex. dans le cas de suivi en temps réel avec transmission des trames sur un réseau), il devient possible que l'objet suivi apparaisse loin de sa dernière position prédite, dans une zone ne faisant pas partie de la région de recherche réduite où les points caractéristiques sont détectés. Dans ce cas, la cible sera perdue jusqu'à sa réapparition dans la région de recherche réduite. Ce problème peut être résolu au coût d'un temps de calcul supplémentaire suite à l'augmentation du nombre de particules constituant la région de recherche réduite.

CHAPITRE 9

CONCLUSION

9.1 Synthèse des travaux

Dans cette thèse, nous avons présenté de nouvelles méthodes de suivi visuel d'objets et de personnes sans connaissance à priori de l'apparence de la cible. Nos travaux de recherche ont visé à développer des modèles d'apparence et des stratégies de recherche d'objets permettant de résoudre des problèmes de suivi dans des scénarios du monde réel.

Pour traiter les difficultés causées par la présence de distracteurs et la variation des conditions d'éclairage dans la scène, nous avons conçu des modèles d'objets par parties basés sur les points SIFT qui se sont avérés remarquablement distinctifs et assez invariants aux changements d'illumination. Vu que le suivi par points caractéristiques est une tâche souvent couteuse nécessitant l'extraction des points d'intérêt, le calcul des descripteurs locaux et l'appariement par des méthodes statistiques sophistiquées, nous avons simplifié la procédure en combinant le modèle par parties avec un modèle holistique de couleurs. L'algorithme en question a été présenté et testé avec succès dans le chapitre 4. Les contributions de cette partie de la thèse sont :

1. l'utilisation d'un modèle holistique de couleur pour simplifier le suivi par points caractéristiques ;
2. une méthode de réduction de région de recherche permettant d'éviter la détection des caractéristiques locales sur toute l'image ou sur une sous-fenêtre arbitraire autour de la dernière position de l'objet.

Le chapitre 5 a présenté un nouvel algorithme de suivi d'objets génériques nommé SAT. Dans cet algorithme, nous avons exploité la disposition spatiale des points caractéristiques pour encoder la structure interne de l'objet. Le modèle d'apparence est appris en ligne en incorporant les propriétés structurelles observées durant tout le cycle de vie de la cible. Grâce à son modèle structurel adaptatif, SAT est capable de prédire avec précision la position de l'objet dans différents scénarios de suivi. Dans la méthode proposée, les occultations partielles sont traitées en se basant sur les contraintes structurelles imposées par des points caractéristiques visibles de l'objet. L'évaluation expérimentale de SAT a montré l'importance et l'efficience du modèle structurel défini par les points caractéristiques. Les contributions de cette partie de la thèse sont :

1. un nouveau modèle d'apparence adaptatif mémorisant les propriétés locales de la cible dans un réservoir de caractéristiques ;
2. l'exploitation de la disposition spatiale des points caractéristiques pour encoder la structure de l'objet ;
3. une nouvelle technique pour l'évaluation de la qualité de suivi par rétroprojection d'un histogramme filtré sur la région prédite ;
4. un nouvel ensemble de séquences vidéos annotées, mis en ligne pour la communauté des chercheurs travaillant sur le suivi d'un seul objet.

La méthode de suivi SCFT présentée dans le chapitre 6 constitue la dernière partie de notre travail. L'algorithme proposé évalue la saillance des points d'intérêt afin de prédire l'état de la cible à travers les prédictions locales des caractéristiques les plus fiables. Le modèle d'apparence de SCFT diffère du modèle de SAT par l'utilisation optimale de toutes les informations contenues dans les descripteurs locaux SIFT. Plus spécifiquement, SCFT utilise les orientations principales des points caractéristiques et les échelles de détection pour estimer respectivement les changements de pose et les changements d'échelle de la cible. Une évaluation expérimentale étendue a montré que SCFT surpasse cinq méthodes récentes de la littérature dans plusieurs scénarios de suivi, tout en étant peu affectable par la diminution du nombre de points caractéristiques. Les contributions de cette dernière partie de la thèse se résument comme suit :

1. une méthode pour évaluer la fiabilité des points caractéristiques, généralisable pour d'autres algorithmes de suivi utilisant le vote d'éléments locaux de l'objet ;
2. l'exploitation des informations sur les échelles de détection et les orientations dominantes des points SIFT pour estimer l'échelle et la pose de l'objet ;
3. la comparaison expérimentale de six AS récents de la littérature (incluant SAT et SCFT).

9.2 Travaux futurs

9.2.1 Amélioration

Tous les algorithmes proposés dans cette thèse s'appuient sur les points d'intérêt SIFT en tant que caractéristiques locales stables et robustes face aux facteurs perturbateurs de l'image. Bien que notre utilisation du détecteur/descripteur SIFT offre plusieurs avantages, elle impose des contraintes sur la taille de la région suivie et/ou sa texture (voir la discussion de la section 8.7). Pour surmonter cette limite, nous proposons d'évaluer la convenance d'autres méthodes de détection de points caractéristiques pour générer les points d'intérêt plus densément. Une

voie intéressante serait d'adopter le détecteur BRISK (Binary Robust Invariant Scalable Keypoints) [127] qui permet de détecter un plus grand nombre de points caractéristiques que SIFT (d'après l'évaluation comparative de Heinly *et al.* [107]), tout en étant invariant à la rotation et à l'échelle.

9.2.2 Extension

Dans le cadre des travaux futurs, nous envisageons aussi de développer une application de vidéosurveillance intelligente utilisant l'algorithme SCFT avec une caméra IP PTZ. Le système fonctionnera d'une manière autonome en réorientant dynamiquement la caméra pour suivre la cible dans un large champ de vision. Ce projet inclut la mise au point d'une stratégie de contrôle de la caméra en tenant compte des délais de mouvement de cette dernière et des délais de réseau. D'autre part, nous allons considérer la simplification de la méthode SCFT et/ou le remplacement de certaines composantes pour répondre le mieux aux exigences de traitement en temps réel et aux contraintes liées au délais. À titre d'exemple, l'utilisation du détecteur/descripteur BRISK permettra de réduire d'environ 13 fois le temps de détection des points caractéristiques et de 24 fois le temps de calcul des descripteurs (selon [107]).

RÉFÉRENCES

[1] V. Gouaillier and A. Fleurant, "La vidéosurveillance intelligente : promesses et défis. rapport de veille technologique et commerciale," *Rapport technique, CRIM and Technopôle Défense et Sécurité*, 2009.

[2] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1, pp. 798–805, 2006.

[3] B. Babenko and M.-H. Y. S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pp. 1619–1632, 2011.

[4] T. Zhao and R. Nevatia, "Bayesian human segmentation in crowded situations," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2, pp. II–459–66 vol.2, June 2003.

[5] T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environment," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2, pp. II–406–II–413 Vol.2, June 2004.

[6] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 125–141, 2008.

[7] M. Lalonde, S. Foucher, L. Gagnon, E. Pronovost, M. Derenne, and A. Janelle, "A system to automatically track humans and vehicles with a PTZ camera," in *Defense and Security Symposium*, International Society for Optics and Photonics, 2007.

[8] N. A. Mandellos, I. Keramitsoglou, and C. T. Kiranoudis, "A background subtraction algorithm for detecting and tracking vehicles," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1619 – 1631, 2011.

[9] S. Wachter and H. Nagel, "Tracking of persons in monocular image sequences," in *Nonrigid and Articulated Motion Workshop, 1997. Proceedings., IEEE*, pp. 2–9, Jun 1997.

[10] R. Urtasun, D. Fleet, A. Hertzmann, and P. Fua, "Priors for people tracking from small training sets," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 1, pp. 403–410 Vol. 1, Oct 2005.

[11] H. Grabner, J. Matas, L. Van Gool, and P. Cattin, "Tracking the invisible : Learning where the object might be," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 1285–1292, IEEE, 2010.

[12] W. Bouachir and G.-A. Bilodeau, "Visual face tracking : A coarse-to-fine target state estimation," *2013 International Conference on Computer and Robot Vision*, pp. 45–51, 2013.

[13] W. Bouachir and G.-A. Bilodeau, "Exploiting structural constraints for visual object tracking," *Image and Vision Computing, submitted on February 10, 2014, manuscript number : IMAVIS-D-14-00064*, 2014.

[14] W. Bouachir and G.-A. Bilodeau, "Structure-aware keypoint tracking for partial occlusion handling," *IEEE Winter Conference on Applications of Computer Vision (WACV 2014)*, 2014.

[15] W. Bouachir and G.-A. Bilodeau, "Collaborative part-based tracking using salient local predictors," *Computer Vision and Image Understanding, submitted on August 29, 2014, manuscript number : CVIU-14-439*, 2014.

[16] W. Bouachir and G.-A. Bilodeau, "Part-based tracking via salient collaborating features," *IEEE Winter Conference on Applications of Computer Vision (WACV 2015), accepted on August 27, 2014.*

[17] S. Ba and J. Odobez, "A probabilistic framework for joint head tracking and pose estimation," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 4, pp. 264–267 Vol.4, Aug 2004.

[18] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pp. 232–237, IEEE, 1998.

[19] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun, "Realtime and robust hand tracking from depth," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*, IEEE, 2014.

[20] V. Philomin, R. Duraiswami, and L. Davis, "Pedestrian tracking from a moving vehicle," in *Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE*, pp. 350–355, 2000.

[21] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, and S. Pankanti, "Smart video surveillance : exploring the concept of multiscale spatiotemporal tracking," *Signal Processing Magazine, IEEE*, vol. 22, pp. 38–51, March 2005.

[22] A. Fossati, P. Schönmann, and P. Fua, "Real-time vehicle tracking for driving assistance," *Machine Vision and Applications*, vol. 22, no. 2, pp. 439–448, 2011.

[23] S. Sivaraman and M. Trivedi, "A general active-learning framework for on-road vehicle recognition and tracking," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, pp. 267–276, June 2010.

[24] M. Danelljan, F. Shahbaz Khan, M. Felsberg, and J. Van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*, IEEE, 2014.

[25] X. Lan, A. J. Ma, and P. C. Yuen, "Multi-cue visual tracking using robust feature-level fusion based on joint sparse representation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*, June 2014.

[26] Z. Zhang and K. H. Wong, "Pyramid-based visual tracking using sparsity represented mean transform," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*, June 2014.

[27] A. Yilmaz, O. Javed, and M. Shah, "Object tracking : A survey," *ACM Computing Surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

[28] C. J. Veenman, M. J. Reinders, and E. Backer, "Resolving motion correspondence for densely moving points," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 1, pp. 54–72, 2001.

[29] M. Grabner, H. Grabner, and H. Bischof, "Learning features for tracking," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pp. 1–8, IEEE, 2007.

[30] H. Tao, H. S. Sawhney, and R. Kumar, "Object tracking with bayesian estimation of dynamic layer representations," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 1, pp. 75–89, 2002.

[31] I. Leichter, M. Lindenbaum, and E. Rivlin, "Mean shift tracking with multiple reference color histograms," *Computer Vision and Image Understanding*, vol. 114, no. 3, pp. 400 – 408, 2010.

[32] K. Sato and J. Aggarwal, "Temporal spatio-velocity transform and its application to tracking and interaction," *Computer Vision and Image Understanding*, vol. 96, no. 2, pp. 100 – 128, 2004. Special Issue on Event Detection in Video.

[33] N. Xu and N. Ahuja, "Object contour tracking using graph cuts based active contours," in *Image Processing. 2002. Proceedings. 2002 International Conference on*, vol. 3, pp. III–277–III–280 vol.3, 2002.

[34] S. He, Q. Yang, R. Lau, J. Wang, and M.-H. Yang, "Visual tracking via locality sensitive histograms," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 2427–2434, June 2013.

[35] T. Zhang, K. Jia, C. Xu, Y. Ma, and N. Ahuja, "Partial occlusion handling for visual tracking via robust part matching," June 2014.

[36] L. Sevilla-Lara and E. Learned-Miller, "Distribution fields for tracking," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1910–1917, IEEE, 2012.

[37] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 1269–1276, IEEE, 2010.

[38] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust L1 tracker using accelerated proximal gradient approach," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1830–1837, IEEE, 2012.

[39] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *Computer Vision–ECCV 2008*, pp. 234–247, Springer, 2008.

[40] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 7, pp. 1409–1422, 2012.

[41] S. Hare, A. Saffari, and P. H. Torr, "Struck : Structured output tracking with kernels," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 263–270, IEEE, 2011.

[42] Q. Zhao, Z. Yang, and H. Tao, "Differential earth mover's distance with its applications to visual tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 2, pp. 274–287, 2010.

[43] P. Varcheie and G. A. Bilodeau, "Adaptive fuzzy particle filter tracker for a PTZ camera in an ip surveillance system," *Instrumentation and Measurement, IEEE Transactions on*, vol. 60, pp. 354–371, Feb 2011.

[44] P. Varcheie and G.-A. Bilodeau, "People tracking using a network-based PTZ camera," *Machine Vision and Applications*, vol. 22, no. 4, pp. 671–690, 2011.

[45] K. Bernardin, F. Van De Camp, and R. Stiefelhagen, "Automatic person detection and tracking using fuzzy controlled active cameras," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pp. 1–8, IEEE, 2007.

[46] M. Z. Islam, C.-M. Oh, and C.-W. Lee, "Video based moving object tracking by particle filter," *International Journal of Signal Processing, Image Processing and Pattern*, vol. 2, no. 1, 2009.

[47] G. D. Hager, M. Dewan, and C. V. Stewart, "Multiple kernel tracking with SSD," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1, pp. I–790, IEEE, 2004.

[48] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 5, pp. 564–577, 2003.

[49] T. J. Broida and R. Chellappa, "Estimation of object motion parameters from noisy images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 1, pp. 90–99, 1986.

[50] G. Kitagawa, "Non-gaussian state—space modeling of nonstationary time series," *Journal of the American statistical association*, vol. 82, no. 400, pp. 1032–1041, 1987.

[51] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter," *Image and vision computing*, vol. 21, no. 1, pp. 99–110, 2003.

[52] M. Isard and A. Blake, "Condensation : conditional density propagation for visual tracking," *International journal of computer vision*, vol. 29, no. 1, pp. 5–28, 1998.

[53] D. Gavrila, "Multi-feature hierarchical template matching using distance transforms," in *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, vol. 1, pp. 439–444 vol.1, Aug 1998.

[54] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2, pp. 142–149, IEEE, 2000.

[55] C. Yang, R. Duraiswami, and L. Davis, "Efficient mean-shift tracking via a new similarity measure," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 176–183, IEEE, 2005.

[56] J. Ning, L. Zhang, D. Zhang, and C. Wu, "Robust mean-shift tracking with corrected background-weighted histogram," *Computer Vision, IET*, vol. 6, pp. 62–69, January 2012.

[57] J. G. Allen, R. Y. Xu, and J. S. Jin, "Object tracking using camshift algorithm and multiple quantized feature spaces," in *Proceedings of the Pan-Sydney area workshop on Visual information processing*, pp. 3–7, Australian Computer Society, Inc., 2004.

[58] D. Comaniciu and V. Ramesh, "Robust detection and tracking of human faces with an active camera," in *Visual Surveillance, 2000. Proceedings. Third IEEE International Workshop on*, pp. 11–18, IEEE, 2000.

[59] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 983–990, June 2009.

[60] O. Williams, A. Blake, and R. Cipolla, "Sparse bayesian learning for efficient visual tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, pp. 1292–1304, Aug 2005.

[61] J. Kang, I. Cohen, and G. Medioni, "Object reacquisition using invariant appearance model," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 4, pp. 759–762, IEEE, 2004.

[62] D. P. Huttenlocher, J. J. Noh, and W. J. Rucklidge, "Tracking non-rigid objects in complex scenes," in *Computer Vision, 1993. Proceedings., Fourth International Conference on*, pp. 93–101, IEEE, 1993.

[63] B. Li, R. Chellappa, Q. Zheng, and S. Z. Der, "Model-based temporal object verification using video," *Image Processing, IEEE Transactions on*, vol. 10, no. 6, pp. 897–908, 2001.

[64] A. Yilmaz, X. Li, and M. Shah, "Contour-based object tracking with occlusion handling in video acquired using mobile cameras," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 11, pp. 1531–1536, 2004.

[65] Y. Chen, Y. Rui, and T. Huang, "Jpdaf based hmm for real-time contour tracking," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, pp. I–543–I–550 vol.1, 2001.

[66] D. Cremers, T. Kohlberger, and C. Schnörr, "Nonlinear shape statistics in mumford—shah based segmentation," in *Computer Vision—ECCV 2002*, pp. 93–108, 2002.

[67] D. Terzopoulos and R. Szeliski, "Tracking with kalman snakes," in *Active vision*, pp. 3–20, MIT press, 1993.

[68] M. Bertalmio, G. Sapiro, and G. Randall, "Morphing active contours," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 7, pp. 733–737, 2000.

[69] A.-R. Mansouri, "Region tracking via level set pdes without motion computation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 947–961, 2002.

[70] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 9, pp. 1627–1645, 2010.

[71] L. Zhu, Y. Chen, A. Yuille, and W. Freeman, "Latent hierarchical structural learning for object detection," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 1062–1069, IEEE, 2010.

[72] Z. Lin, G. Hua, and L. S. Davis, "Multiple instance feature for robust part-based object detection," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 405–412, IEEE, 2009.

[73] Y. Amit and A. Trouvé, "Pop : Patchwork of parts models for object recognition," *International Journal of Computer Vision*, vol. 75, no. 2, pp. 267–282, 2007.

[74] P. Schnitzspan, S. Roth, and B. Schiele, "Automatic discovery of meaningful object parts with latent crfs," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 121–128, IEEE, 2010.

[75] S. Shahed Nejhum, J. Ho, and M.-H. Yang, "Visual tracking with histograms and articulating blocks," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8, June 2008.

[76] E. Erdem, S. Dubuisson, and I. Bloch, "Fragments based tracking with adaptive cue integration," *Computer Vision and Image Understanding*, vol. 116, no. 7, pp. 827 – 841, 2012.

[77] Y. Guo, Y. Chen, F. Tang, A. Li, W. Luo, and M. Liu, "Object tracking using learned feature manifolds," *Computer Vision and Image Understanding*, vol. 118, pp. 128–139, 2014.

[78] R. Yao, Q. Shi, C. Shen, Y. Zhang, and A. van den Hengel, "Part-based visual tracking with online latent structural learning," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 2363–2370, IEEE, 2013.

[79] J. Kwon and K. M. Lee, "Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 1208–1215, June 2009.

[80] G. Hua and Y. Wu, "Measurement integration under inconsistency for robust tracking," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1, pp. 650–657, June 2006.

[81] F. Porikli, "Integral histogram : A fast way to extract histograms in cartesian spaces," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 829–836, IEEE, 2005.

[82] M. J. Jones and P. Viola, "Robust real-time object detection," in *Workshop on Statistical and Computational Theories of Vision*, vol. 266, 2001.

[83] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparsity-based collaborative model," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1838–1845, IEEE, 2012.

[84] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1822–1829, IEEE, 2012.

[85] S. Wang, H. Lu, F. Yang, and M.-H. Yang, "Superpixel tracking," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 1323–1330, IEEE, 2011.

[86] W. Wang and R. Nevatia, "Robust object tracking using constellation model with superpixel," in *Computer Vision–ACCV 2012*, pp. 191–204, Springer, 2013.

[87] X. Ren and J. Malik, "Tracking as repeated figure/ground segmentation," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pp. 1–8, IEEE, 2007.

[88] A. Radhakrishna, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC superpixels," *Dept. School Comput. Commun. Sci., EPFL, Lausanne, Switzerland, Tech. Rep*, vol. 149300, 2010.

[89] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[90] E. Rosten, R. Porter, and T. Drummond, "Faster and better : A machine learning approach to corner detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 1, pp. 105–119, 2010.

[91] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF : Speeded up robust features," in *Computer Vision–ECCV 2006*, pp. 404–417, Springer, 2006.

[92] F. Yang, H. Lu, and M.-H. Yang, "Learning structured visual dictionary for object tracking," *Image and Vision Computing*, vol. 31, no. 12, pp. 992–999, 2013.

[93] S. Hare, A. Saffari, and P. H. Torr, "Efficient online structured output learning for keypoint-based object tracking," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1894–1901, IEEE, 2012.

[94] M. Yang, Y. Wu, and G. Hua, "Context-aware visual tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 7, pp. 1195–1209, 2009.

[95] L. Cerman, J. Matas, and V. Hlaváč, "Sputnik tracker : Having a companion improves robustness of the tracker," in *Image Analysis : 16th Scandinavian Conference, Scia 2009, Oslo, Norway, June 15-18, Proceedings*, vol. 5575, p. 291, Springer, 2009.

[96] S. Gu and C. Tomasi, "Branch and track," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1169–1174, IEEE, 2011.

[97] S. Baker and I. Matthews, "Lucas-kanade 20 years on : A unifying framework," *International journal of computer vision*, vol. 56, no. 3, pp. 221–255, 2004.

[98] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker : Exploring supporters and distracters in unconstrained environments," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1177–1184, IEEE, 2011.

[99] L. Matthews, T. Ishikawa, and S. Baker, "The template update problem," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 6, pp. 810–815, 2004.

[100] L. Guan, J.-S. Franco, and M. Pollefeys, "3D occlusion inference from silhouette cues," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pp. 1–8, IEEE, 2007.

[101] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah, "Part-based multiple-person tracking with partial occlusion handling," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1815–1821, IEEE, 2012.

[102] M. Gouiffes, C. Collewet, C. Fernandez-Maloigne, and A. Trémeau, "Feature points tracking : robustness to specular highlights and lighting changes," *Computer Vision–ECCV 2006*, pp. 82–93, 2006.

[103] H. Zhou, Y. Yuan, and C. Shi, "Object tracking using SIFT features and mean shift," *Computer Vision and Image Understanding*, vol. 113, no. 3, pp. 345–352, 2009.

[104] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 10, pp. 1615–1630, 2005.

[105] V. Belagiannis, F. Schubert, N. Navab, and S. Ilic, "Segmentation based particle filtering for real-time 2D object tracking," *Computer Vision–ECCV 2012*, pp. 842–855, 2012.

[106] H.-M. Sun, "Skin detection for single images using dynamic skin color modeling," *Pattern recognition*, vol. 43, no. 4, pp. 1413–1420, 2010.

[107] J. Heinly, E. Dunn, and J.-M. Frahm, "Comparative evaluation of binary features," *Computer Vision–ECCV 2012*, pp. 759–773, 2012.

[108] L. Juan and O. Gwun, "A comparison of SIFT, PCA-SIFT and SURF," *International Journal of Image Processing (IJIP)*, vol. 3, no. 4, pp. 143–152, 2009.

[109] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Computer Vision–ECCV 2012*, pp. 702–715, Springer, 2012.

[110] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting.," in *BMVC*, vol. 1, p. 6, 2006.

[111] Z. Kalal, J. Matas, and K. Mikolajczyk, "Pn learning : Bootstrapping binary classifiers by structural constraints," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 49–56, IEEE, 2010.

[112] L. Zhang and L. van der Maaten, "Structure preserving object tracking," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 1838–1845, 2013.

[113] L. Wen, Z. Cai, Z. Lei, D. Yi, and S. Z. Li, "Online spatio-temporal structural context learning for visual tracking," in *Computer Vision–ECCV 2012*, pp. 716–729, Springer, 2012.

[114] A. Saffari, M. Godec, T. Pock, C. Leistner, and H. Bischof, "Online multi-class LPBoost," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 3570–3577, 2010.

[115] K. M. Yi, H. Jeong, B. Heo, H. J. Chang, and J. Y. Choi, "Initialization-insensitive visual tracking through voting with salient local features," *2013 International Conference on Computer Vision (ICCV)*, 2013.

[116] A. Mian, M. Bennamoun, and R. Owens, "An efficient multimodal 2D-3D hybrid approach to automatic face recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 11, pp. 1927–1943, 2007.

[117] C. Geng and X. Jiang, "Face recognition using SIFT features," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pp. 3313–3316, 2009.

[118] A. Mian, M. Bennamoun, and R. Owens, "Keypoint detection and local feature matching for textured 3D face recognition," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 1–12, 2008.

[119] Q.-H. Zhou, H. Lu, and M.-H. Yang, "Online multiple support instance tracking," in *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pp. 545–552, IEEE, 2011.

[120] O. Javed and M. Shah, "Tracking and object classification for automated surveillance," in *European Conference on Computer Vision (ECCV)*, pp. 343–357, Springer, 2002.

[121] B. Lei and L.-Q. Xu, "Real-time outdoor video surveillance with robust foreground extraction and object tracking via multi-state transition management," *Pattern Recognition Letters*, vol. 27, no. 15, pp. 1816–1825, 2006.

[122] J.-P. Jodoin, G.-A. Bilodeau, and N. Saunier, "Urban tracker : Multiple object tracking in urban mixed traffic," in *Winter Applications of Computer Vision Conference (WACV)*, IEEE, 2014.

[123] M. Keck, L. Galup, and C. Stauffer, "Real-time tracking of low-resolution vehicles for wide-area persistent surveillance," in *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*, pp. 441–448, Jan 2013.

[124] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 3169–3176, June 2011.

[125] W. Choi, C. Pantofaru, and S. Savarese, "Detecting and tracking people using an RGB-D camera via multiple detector fusion," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pp. 1076–1083, Nov 2011.

[126] G. Nebehay and R. Pflugfelder, "Consensus-based matching and tracking of keypoints for object tracking," in *Winter Conference on Applications of Computer Vision*, 2014.

[127] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK : Binary robust invariant scalable keypoints," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2548–2555, IEEE, 2011.

[128] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking : A benchmark," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 2411–2418, IEEE, 2013.

[129] J. S. Supancic III and D. Ramanan, "Self-paced learning for long-term tracking," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 2379–2386, IEEE, 2013.

[130] F. Tang and H. Tao, "Probabilistic object tracking with dynamic attributed relational feature graph," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, pp. 1064–1074, Aug 2008.