Computer Vision and Image Understanding 116 (2012) 1149–1158

Contents lists available at SciVerse ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Evolutionary Hough Games for coherent object detection ☆

Peter Kontschieder [a,*], Samuel Rota Bulò [b], Michael Donoser [a], Marcello Pelillo [b], Horst Bischof [a]

[a] *Institute for Computer Graphics and Vision, Graz University of Technology, Austria*
[b] *Dipartimento di Scienze Ambientali, Informatica e Statistica, Università Ca' Foscari Venezia, Italy*

ARTICLE INFO

Article history:
Received 20 March 2012
Accepted 20 August 2012
Available online 29 August 2012

Keywords:
Hough space
Object detection
Evolutionary Game Theory

ABSTRACT

In this paper we propose a novel, game-theoretic approach for finding multiple instances of an object category as sets of mutually coherent votes in a generalized Hough space. Existing Hough-voting based detection systems have to inherently apply parameter-sensitive non-maxima suppression (NMS) or mode detection techniques for finding object center hypotheses. Moreover, the voting origins contributing to a particular maximum are lost and hence mostly bounding boxes are drawn to indicate the object hypotheses. To overcome these problems, we introduce a two-stage method, applicable on top of any Hough-voting based detection framework. First, we define a Hough environment, where the geometric compatibilities of the voting elements are captured in a pairwise fashion. Then we analyze this environment within a game-theoretic setting, where we model the competition between voting elements as a Darwinian process, driven by their mutual geometric compatibilities. In order to find multiple and possibly overlapping objects, we introduce a new enumeration method inspired by tabu search. As a result, we obtain locations and voting element compositions of each object instance while bypassing the task of NMS. We demonstrate the broad applicability of our method on challenging datasets like the extended TUD pedestrian crossing scene.

1. Introduction and related work

The Hough transform [1] in its initial formulation 50 years ago has proven to be a fundamental method for line detection in edge images. Subsequently, Ballard came up with a generalization [2] for detecting arbitrary parametric shapes. The basic idea behind the Hough concept is to abstract the input image into voting elements and transfer the detection process in a voting space, the so-called Hough space. Each voting element is allowed to cast a directional vote for a specific location in the Hough space. The entries in these locations are grouped in a point-wise, accumulative style and are associated to certain hypotheses, indicating the confidences for shape presence. The quality of a hypothesis, i.e. its certainty about shape presence, depends on the associated peak in the Hough domain.

Due to its generality, the Hough concept has recently gained much attention in the field of part-based object detection as e.g. in [3–5]. Typically, such methods construct a codebook from training data for the desired object category, containing local patch descriptors (voting elements) together with their respective center vote information. Given e.g. bounding box annotated training data, the center vote information is simply obtained as offset vector between the voting element within the bounding box and its respective center. Since the Hough space representation is decoupled from the original image domain, there are no restrictions on how the voting elements have to be obtained. For example, some methods use interest points [3], dense sampling [4], boundary fragments [6] or edge groups [7]. In the detection step, the test image is matched to the codebook and the corresponding voting elements are projected in the Hough space. Traditionally, the resulting peaks are analyzed with respect to their confidence values and furthermore treated as possible object location hypotheses. In such a way, analysis of the Hough space should simply boil down to identifying true object locations and discriminating them from wrong ones.

Unfortunately, this maxima detection is complicated by several problems arising in practical Hough-voting detection systems. First, projections of voting elements in the Hough space produce scattered and therefore smeared hypotheses such that there are no unique observations of (local) maxima. Second, once the voting elements are cast to the Hough space, their origin is lost due to the accumulative aggregation. Moreover, this implies that re-associations (i.e. back-projections) between particular maxima and their voting origins are questionable since noisy contributions from non-object votes cannot be explained right away. Third, there is no evidence whether an object is present or not since the number of objects in an image cannot be determined, based on the Hough space alone. Of course, maxima in the Hough space should coincide

☆ This paper has been recommended for acceptance by Walter G. Kropatsch, Ph.D.
* Corresponding author.
E-mail address: kontschieder@icg.tugraz.at (P. Kontschieder).

1077-3142/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.cviu.2012.08.003

with true object centers but in practice very often all peaks are collected that exceed a certain, application-specific threshold. Finally, for overlapping or strongly occluded objects, maxima analysis in the Hough space is largely undefined. Therefore, Hough space analysis should not be performed as vanilla maxima thresholding but instead take all contributing voting elements *and* their mutual agreement on the object centroid hypothesis into account.

To obtain meaningful object center hypotheses from the Hough space, *non-maxima suppression* (NMS) techniques are typically applied to resolve at least the first of the above mentioned problems. For instance, NMS aims for suppressing all hypotheses (or bounding boxes) within a certain distance and quality with respect to each other. Various NMS techniques were introduced but most of them turn out to be application-specific heuristics with significant effort for parameter tuning. Recently, Barinova et al. [8] came up with a probabilistic interpretation of the Hough transform, reformulated into a facility location problem in order to bypass the task of NMS. However, the resulting maximization problem is NP-hard and cannot be solved in reasonable time. Instead, maximization was tackled by iteratively applying a greedy algorithm together with a NMS scheme. Their NMS strategy enforced to pick contributing voting elements only within (a) a pre-defined range and (b) voting quality above a certain threshold. As a result, they provided a bounding box delimited object detection.

In this work, we propose a novel method for finding (multiple) instances of an object category, applicable on top of arbitrary Hough-voting based detection frameworks like the Hough Forest [4] or the Implicit Shape Model (ISM) [3]. More specifically, we detect and describe each object as the set of geometrically coherent voting elements, meaning that each of our final object hypotheses consists of all mutually compatible voting elements. This is in contrast to bounding-box delimited object detections and makes additional NMS unnecessary. Our method mainly consists of the following two steps.

First, we are analyzing the voting element information in a pairwise manner. Therefore, we are introducing a compatibility function that transfers pairs of votes with respect to their hypothesized object center location and their geometrical constellation into a compatibility matrix representation. In such a way, we combine geometric information like orientation and object center agreement between voting elements together in a novel *Hough environment* formulation.

In the second step, we jointly assemble the individual object instances as geometrically coherent sets by introducing a novel, game-theoretic detection approach. Specifically, we model the competition between contributing voting elements as a Darwinian process within the context of *Evolutionary Game Theory* (Egt) [9], where the mutual geometrical compatibilities drive the selection mechanism. This process imitates the well-known principle of survival of the fittest, which in our case corresponds to identifying the subset of voting elements that best possibly assemble the object category to be found. As a result, we obtain precise information about presence, location and composition of an object at reasonable computational time. The advantages of our method over standard Hough space analysis are subsumed as follows:

- We exploit geometric information provided by the individual voting elements rather than analyzing the accumulative Hough space. Enforcing structural coherence prunes away spurious location hypotheses.
- The proposed evolutionary game theoretical formulation identifies sets of mutually compatible, coherent voting elements with respect to arbitrary, learned object categories.
- The nature of the evolutionary process can leave voting elements unassigned, i.e. spurious contributions, noise and outliers are massively suppressed or even ignored.
- The proposed method is also capable of assigning voting elements to multiple objects. This is particularly helpful when objects occlude each other.

A disadvantage of the proposed method (in its general formulation) can be found in the quadratically growing compatibility matrix, holding the pairwise voting information. However, we show ways to incorporate prior knowledge about the object category from training data as well as sampling strategies to overcome this limitation and to keep the matrix at a reasonable size.

The rest of the paper is organized as follows. We start in Section 2 by introducing the Hough Environment, i.e. the space where our pairwise, geometric voting compatibilities live. In Section 3 we explain some basics of Egt to understand what happens in the biologically inspired voting selection process, before we show our definition of *Hough Games* and our novel detection algorithm. Broad applicability and experimental results of our method are demonstrated in Section 4 before we conclude in Section 5.

2. From Hough space to Hough environment

In this section we first introduce the notation that will be used throughout the rest of this paper. Then, we briefly review a probabilistic interpretation of the classical Hough transform, mainly from [8]. Finally, we describe our proposed Hough environment where we include geometric information in a pairwise setting to incorporate a more holistic view in the detection process.

We denote with S the set of N observations (voting elements) from an image. Each observation $i \in S$ has a spatial origin $\mathbf{y}_i$ in the image space, stemming from voting elements and their respective descriptors I_i. We furthermore assume that we are given a classification function $h(i)$ for the class label assignment and a probability score $p(h(i)|I_i)$ for each voting element $i \in S$. In addition, each voting element $i \in S$ obtains a voting vector $\mathbf{d}_i$ after being classified, pointing towards its associated object center (see Fig. 1). All of the above parameters can be obtained from previous works like the ISM [3] or the Hough Forest [4,10]. Most of our experiments in the experimental section are based on the Hough Forest framework, which we briefly summarize for the sake of completeness.

A Hough Forest is an ensemble $\mathcal{T}$ of binary decision trees, similar to the popular concept of randomized decision trees [11–13]. The main difference to traditional random forests is that input samples are mapped to probabilistic votes for an object centroid location hypothesis in the Hough space instead of pure classification [14]

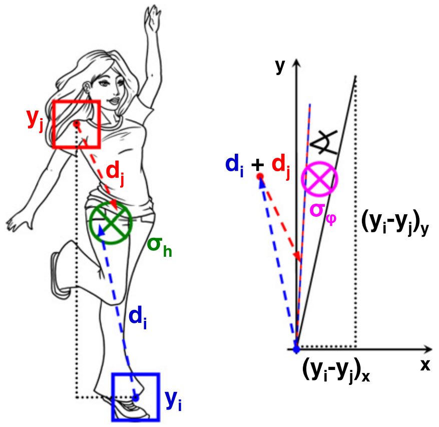

Fig. 1. Sketches of geometrical features used in proposed Hough environment. The left figure shows the geometry used for voting center compatibility estimation while the right one illustrates the features used to compute orientation compatibility.

or regression [15] tasks. More specifically, a sample is routed through a tree $t \in \mathcal{T}$ by applying binary tests in each non-leaf node until a leaf is reached. The training data used to generate a leaf node defines both, the conditional class probability $p(c_i|I_i)$ and a (possibly empty) set of voting vectors $\{\mathbf{d}\}$. Therefore, the trees can be grown recursively, using a set of training data triplets $\{(I_i, \mathbf{d}_i, c_i)\}$ with I_i and $\mathbf{d}_i$ as defined before and c_i being the class label of sample i. The goal of the training process is to partition the training data in a way such that the uncertainty about the class labels and the scatter of the voting vectors is minimized. This can be achieved by selecting proper test parameters for the nodes during training time. For a more detailed description of the Hough Forest we refer the interested reader to [4,10].

Within the generalized Hough transform [2], object detection is formulated by independently considering each voting element $i \in S$ as being generated from some object $h \in \mathcal{H}$, where $\mathcal{H}$ is the Hough space, or from no object at all. Let $\mathcal{H}' = \mathcal{H} \cup \{0\}$ be the Hough space augmented with a special element $\{0\}$ and let $x_i \in \mathcal{H}'$ be a random variable indicating whether the voting element $i \in S$ has been generated by object $h \in \mathcal{H}$ or by no object $\{0\}$. In this way, votes can be considered as pseudo-densities $V(x_i = h|I_i)$ conditioned on their respective descriptor I_i. Summing up the conditional (pseudo)-densities over all observations is then supposed to establish local maxima in the Hough space which can be analyzed for valid hypotheses.

Assuming independence between voting elements is clearly a very rough approximation, especially since neighboring locations are likely to belong to the same object in the image. To bypass this independence assumption, we propose a new interpretation of the Hough space, namely the *Hough environment*. Instead of accumulatively combining center voting information in the Hough space, we aim for a joint combination with additional pairwise constraints derived from respective origins and orientations. This additional, geometric information helps to infer meaningful object locations, even in cases where no dedicated peaks in the associated Hough space are available.

In our proposed Hough environment, we explicitly stress the importance of geometry and spatial relations among intra-object voting elements. Therefore, we are modeling joint densities of pairs of voting elements $i,j \in S$ with respect to their agreements on their mutually hypothesized center location c_{ij} in the Hough domain and their relative angular orientation φ_{ij} and in the image domain. See Fig. 1 for respective illustrations. Please note that our setup is not restricted to the above mentioned features but may also be extended with context or object specific configuration information [16,17].

The pairwise compatibility for the object center certainty is modelled as a function weighting the distance between the hypothesized centers of voting elements $i,j \in S$ according to

$$p_{c_{ij}} = \exp\left(-\frac{\|(\mathbf{y}_i + \mathbf{d}_i) - (\mathbf{y}_j + \mathbf{d}_j)\|^2}{\sigma_h^2}\right), \tag{1}$$

where σ_h is a parameter to control the allowed deviation. This term may also be considered as pairwise breakdown of the original Hough center projection.

The second component in the Hough environment models the orientational similarities between the considered pair of votes and the actual relative orientation between the spatial origins in the image domain. Hence, we define

$$p_{\varphi_{ij}} = \exp\left(-\frac{\sphericalangle(\hat{\mathbf{y}}_{ij}, \hat{\mathbf{d}}_{ij})^2}{\sigma_\varphi^2}\right), \tag{2}$$

where $\sphericalangle(\cdot,\cdot)$ returns the enclosed angle between the normalized vectors $\hat{\mathbf{y}}_{ij} = \frac{\mathbf{y}_i - \mathbf{y}_j}{\|\mathbf{y}_i - \mathbf{y}_j\|}$ and $\hat{\mathbf{d}}_{ij} = \frac{\mathbf{d}_i + \mathbf{d}_j}{\|\mathbf{d}_i + \mathbf{d}_j\|}$, mapped on the interval $[0,\pi]$ (see Fig. 1, right). This orientation feature penalizes differences between the observed geometric configuration in the image and the provided voting information. σ_φ allows to control the influence of the orientation feature.

By combining the terms in Eqs. (1) and (2), we construct a *compatibility function* $C{:}S \times S \to [0,1]$ defined as follows:

$$C(i,j) = p(h(i)|I_i)p(h(j)|I_j)p_{c_{ij}}p_{\varphi_{ij}}. \tag{3}$$

Please note that a voting pair (i,j) has to satisfy not only the geometrical constraints formulated in Eqs. (1) and (2) but also needs to be classified as part of the object in order to receive a non-zero compatibility value.

3. Non-cooperative Hough Games

Objects within the Hough environment are identified as sets of voting elements exhibiting high mutual compatibilities. However, the characterization and the retrieval of those sets is not a trivial task. Indeed, two objects may have, in general, some voting elements in common (i.e., they may overlap) and, moreover, many voting elements are noisy, i.e., they do not contribute to any object, and should thus be avoided.

Despite this challenging scenario, it turns out that game theory can provide an intriguing and effective solution to our problem. Indeed, we will show in the next section how mutually compatible sets of voting elements can be characterized in terms of equilibria of what we call a Hough game. We will then also cope with the algorithmic problem of extracting those equilibria in order to successfully detect multiple objects from a scene.

3.1. Game-theoretic background

Inspired by Torsello et al. [18], we define a *Hough game*, i.e., a non-cooperative two-player symmetric game $\Gamma = (S,A)$ [9], where S is a finite set of *pure strategies* (in the language of game-theory) available to the players, and $A : S \times S \to \mathbb{R}$ is a payoff (or utility) function, where $a_{ij} = A(i,j)$ gives the payoff that a player gains when playing strategy $i \in S$ against an opponent playing strategy $j \in S$. Note that in the sequel we will treat the payoff function A as a matrix which is indexable using elements in S. Within the context of Hough environments, the strategy set S coincides with the set of voting elements, whereas the payoff matrix $A = (a_{ij})$ is defined as follows:

$$a_{ij} = \begin{cases} C(i,j) & \text{if } i \neq j, \\ -\alpha & \text{if } i = j, \end{cases}$$

where C is defined in (3) and $\alpha \geqslant 0$ is a penalization constant, whose role will be clarified later.

Two players with complete information about the Hough game play by simultaneously selecting a voting element from the strategy set and, after disclosing their choices, each player receives a reward proportional to the compatibility of the selected element with respect to the one played by the opponent. Since it is in each players' interest to maximize his own payoff and since no player has prior knowledge about what the opponent is going to choose, the best strategy for a player is the selection of voting elements belonging to a common object. Those elements, indeed, exhibit high mutual compatibilities and by selecting them the chance of earning a higher payoff is increased for both players. Hence, a set of voting elements belonging to a common object arises from what is called an equilibrium of the Hough game. Note that the penalization constant α in the payoff matrix has the important role of preventing trivial solutions where the players select exactly the same voting element. Indeed, we are searching for the configuration of voting elements which yields the maximal coherency with respect to a common object centroid, i.e. an object hypothesis in the Hough environment. In what follows, we provide a formal characterization of game-theoretic equilibria and how they serve our purpose for object detection.

A *mixed strategy* (or randomized strategy) $\mathbf{x} \in \Delta_S$ is a probability distribution over the set of pure strategies S, i.e., an element of the *standard simplex* Δ_S (or simply Δ, if not ambiguous), which is defined as

$$\Delta_S = \left\{ \mathbf{x} : S \to [0,1] : \sum_{i \in S} x_i = 1 \text{ and } x_i \geqslant 0, \forall i \in S \right\}.$$

This models a stochastic playing strategy for a player, where $x_i = x(i)$ denotes the probability that the player will select the voting element $i \in S$. Note that as for the payoff function, we will treat $\mathbf{x} \in \Delta_S$ as a (column) vector, which is indexable by elements in S. The support of a mixed strategy $\mathbf{x} \in \Delta$, denoted by $\sigma(\mathbf{x})$, is the set of elements with non-zero probability, i.e., $\sigma(\mathbf{x}) = \{i \in S : x_i > 0\}$. The expected payoff received by a player playing mixed strategy $\mathbf{y} \in \Delta$ against an opponent playing mixed strategy $\mathbf{x} \in \Delta$ is $\mathbf{y}^\top A\mathbf{x} = \sum_{i,j \in S} a_{ij} y_i x_j$.

An important notion in game theory is that of an equilibrium [9]. A mixed strategy $\mathbf{x} \in \Delta$ is a *Nash equilibrium* if it is best reply to itself, i.e., for all $\mathbf{x} \in \Delta$, $\mathbf{y}^\top A\mathbf{x} \leqslant \mathbf{x}^\top A\mathbf{x}$. Intuitively, if $\mathbf{x} \in \Delta$ is a Nash equilibrium then there is no incentive for a player to play differently from $\mathbf{x}$. A refinement of the Nash equilibrium pertaining to Egt, which plays a pivotal role in this work, is that of an *Evolutionary Stable Strategy* (Ess), which is a Nash equilibrium robust to evolutionary pressure in an exact sense [9]. A mixed strategy $\mathbf{x}$ is an Ess if $\mathbf{x}$ is a Nash equilibrium such that for all $\mathbf{y} \in \Delta \backslash \{\mathbf{x}\}$, $\mathbf{y}^\top A\mathbf{x} = \mathbf{x}^\top A\mathbf{x} \Rightarrow \mathbf{y}^\top A\mathbf{x} > \mathbf{y}^\top A\mathbf{y}$.

3.2. Relations to graph theory and optimization theory

Note that the concept of Ess has an interesting graph-theoretic characterization. Indeed, it coincides with the idea of a Dominant Set, which has been recently introduced in [19,18] and generalizes the notion of maximal cliques to edge-weighted graphs. Another interpretation can be performed from an optimization theoretic viewpoint. Indeed, the detection problem as formulated in our paper can be considered as a quadratic optimization problem (QAP). In [19], the authors established a one-to-one correspondence between the equilibriums of a non-cooperative, two-player game as used in our work and the optima of a QAP. However, the interpretation in terms of an optimization problem holds only true if the payoff matrix fulfills certain properties. The game-theoretic interpretation does not impose such constraints, but instead allows arbitrary compatibilities between the pairs. For example, the advantages of using a game-theoretic formulation for computer vision applications have been recently demonstrated in [20] for matching, shape retrieval [21] and common visual pattern discovery [22]. In this line, we motivate the use of Ess for identifying sets of mutually highly coherent voting elements in the Hough space for performing object detection.

3.3. Dynamics for coherent object detection

We will focus now on the computational aspects of our framework, i.e., the extraction of the Ess of a Hough game $\Gamma = (S,A)$. To this end, we undertake an evolutionary setting. Consider a large population of non-rational players, each of which plays a pre-assigned strategy from S, and let $\mathbf{x}(t) \in \Delta$ be the distribution of strategies in the population at time t. Randomly, pairs of players are drawn from the population to play the game Γ and a selection mechanism, where the reproductive success is driven by the payoff gathered by the players, changes the distribution of strategies $\mathbf{x}(t)$ in the population over time. This Darwinian process continues until an equilibrium is eventually reached.

Different dynamics modeling this process have been proposed in Egt and may potentially serve our need of finding Ess of the Hough game. The most famous one is the so-called *Replicator Dynamics* [9]. Fig. 2 shows the result of running the Replicator Dynamics on a Hough game. Initially, the population is uniformly distributed over the set of voting elements (patch samples) and, as time passes, the evolutionary pressure drives the distribution mass on the target object, namely the car, which in turn represents an equilibrium of the game. In this work, however, we will make use of a new class of dynamics called *Infection and Immunization Dynamics* (InImDyn), which has recently been developed [23,24], and overcomes some limitations of the Replicator Dynamics. The rest of this subsection provides a brief review of InImDyn and we refer to the original papers for further details.

The InImDyn dynamics finds an equilibrium of a Hough game $\Gamma = (S,A)$ by iteratively refining an initial mixed strategy $\mathbf{x} \in \Delta$. The procedure is summarized in Algorithm 1. The refinement loop (lines 2–6) continues until $\mathbf{x}$ is close to being a Nash equilibrium according to a tolerance τ, i.e. until $\epsilon(\mathbf{x}) < \tau$, where

$$\epsilon(\mathbf{x}) = \sum_{i \in S} \min\{x_i, (A\mathbf{x})_i - \mathbf{x}^\top A\mathbf{x}\}^2$$

measures the deviation of $\mathbf{x}$ from being a Nash equilibrium and it yields zero if and only if $\mathbf{x}$ is a Nash equilibrium.

Algorithm 1: InImDyn Algorithm for equilibrium selection

```
  Input: A game $\Gamma = (S, A)$, an initial mixed strategy $\mathbf{x} \in \Delta$ and a
         tolerance $\tau$
  Output: A Nash equilibrium of $\Gamma$
1 while $\epsilon(\mathbf{x}) > \tau$ do
2     $\mathbf{y} \leftarrow \mathcal{S}(\mathbf{x})$
3     $\delta \leftarrow 1$
4     if $(\mathbf{y}-\mathbf{x})^\top A(\mathbf{y}-\mathbf{x}) < 0$ then
5         $\delta \leftarrow \min\left\{\frac{(\mathbf{x}-\mathbf{y})^\top A\mathbf{x}}{(\mathbf{y}-\mathbf{x})^\top A(\mathbf{y}-\mathbf{x})}, 1\right\}$
6     $\mathbf{x} \leftarrow \delta\mathbf{y} + (1-\delta)\mathbf{x}$
7 return x
```

If the execution enters the loop then $\mathbf{x}$ is not a Nash equilibrium and, hence, there exist strategies $\mathbf{y} \in \Delta$ such that $(\mathbf{y} - \mathbf{x})^\top A\mathbf{x} > 0$ (called *infective* strategies according to [23]). In particular the following two infective strategies exist [23, Proposition 2]:

$$\mathbf{z}^+ = \mathbf{e}^u \quad \text{and} \quad \mathbf{z}^- = \frac{x_v}{1 - x_v}(\mathbf{x} - \mathbf{e}^v) + \mathbf{x},$$

where $u \in \arg\max_{j \in S}(A\mathbf{x})_j$ and $v \in \arg\min_{j \in \sigma(\mathbf{x})}(A\mathbf{x})_j$. Intuitively, in $\mathbf{z}^+$ only the best performing pure strategy (according to a myopic decision) is present in the mixed strategy, while in $\mathbf{z}^-$ the worst one is extinguished. The function $\mathcal{S}(\mathbf{x})$ at line 2 returns the strategy between $\mathbf{z}^+$ and $\mathbf{z}^-$ yielding the largest expected payoff against $\mathbf{x}$, i.e.

$$\mathbf{y} = \mathcal{S}(\mathbf{x}) \in \arg\max_{\mathbf{z} \in \{\mathbf{z}^+, \mathbf{z}^-\}} \mathbf{z}^\top A\mathbf{x}.$$

Finally, the original strategy $\mathbf{x}$ and the new one $\mathbf{y}$ are linearly combined at line 6 in a way to guarantee a maximum increase in the expected payoff. Indeed, under symmetry assumption of A, the parameter δ computed at lines 3–5 can be regarded to as the solution of

$$\delta \in \arg\max_{\alpha \in [0,1]} \left\{ \mathbf{z}_\alpha^\top A \mathbf{z}_\alpha : \mathbf{z}_\alpha \in \Delta \right\},$$

where $\mathbf{z}_\alpha = (1 - \alpha)\mathbf{x} + \alpha\mathbf{y}$.

The InImDyn dynamics exhibits desired features in contrast to the Replicator Dynamics. There is a one-to-one correspondence between fixed points of the dynamics and the set of Nash equilibria

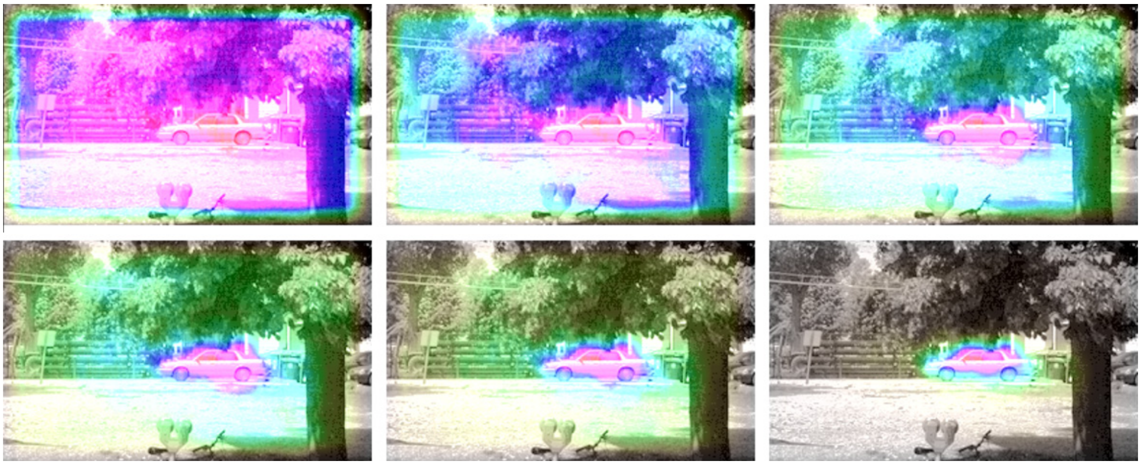

Fig. 2. Proposed Hough game with its evolutionary process applied to car detection. Starting from homogeneous initialization (top left) until convergence (bottom right) where superimposed, reddish regions correspond to fitness of underlying importance with respect to the sought object geometry. Best viewed in color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

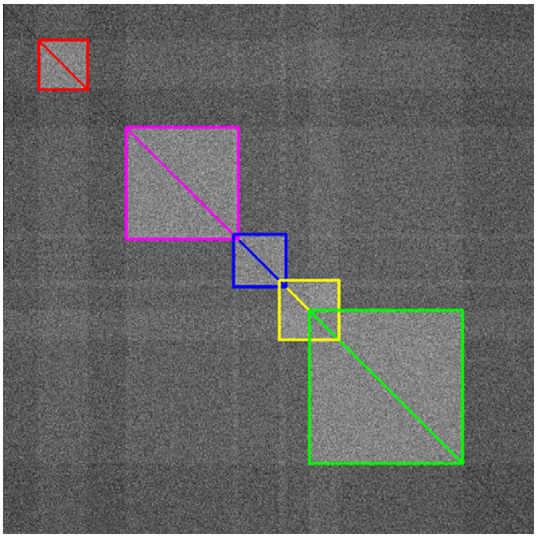

Fig. 3. Visualization of enumeration results on a synthetically generated payoff matrix containing five clusters with varying sizes and overlaps.

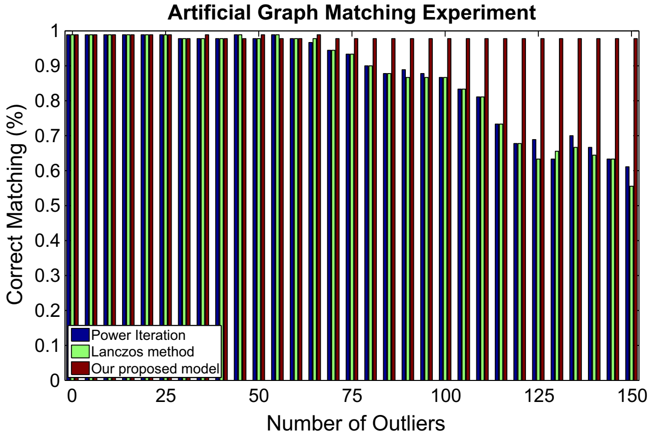

Fig. 4. Matching reference graphs (30 points) to query graphs with an increasing number of randomly generated outliers (up to 150). The proposed method is compared to two widespread spectral methods: Lanczos and power iteration. Cleary, our method handles outliers in a more robust way.

[23, Theorem 1]. Moreover, there are *strong convergence guarantees* under symmetric payoff matrices [23, Theorems 3,4]: the limit points of the dynamics are Nash equilibria and ESS equilibria are all and the only asymptotically stable points. From a computational perspective, iterations are *linear* in the number of strategies [24, Theorem 8] in contrast to the quadratic ones of the Replicator Dynamics and an equilibrium can be extracted after *finitely* many iterations [23, Theorem 7].

3.4. Enumeration of multiple objects

Since an evolutionary dynamics will find at most one equilibrium depending on the initial population state, one problem to solve is how to enumerate multiple Ess in order to detect multiple objects. A simple approach consists in starting multiple times the dynamics from random initial points hoping to converge to different equilibria. This is, however, an inefficient way of exploring the solution space. Another naive method uses a *peeling-off strategy*, i.e., one can iteratively remove the strategies in the support of newly extracted equilibria from the game. This solution, however, cannot find equilibria with overlapping supports and since at each iteration the game is changed, one may potentially introduce equilibria which do not exist in the original game. Due to the lack of a satisfactory solution, we will present a novel heuristic approach for enumerating Ess in the next section, which is built upon the novel InImDyn dynamics.

Our enumeration approach is simple and makes use of both InImDyn as an equilibrium selection algorithm and a *strategy tabu-set* in order to heuristically explore the solution space and have a termination guarantee. The algorithm repeatedly interleaves an exploratory and a validation phase. During the exploratory phase, an equilibrium is found by prohibiting the players from using strategies appearing in the tabu-set. By limiting the strategy set, we basically find an equilibrium of a sub-game and not of the original Hough game, but this allows us to explore a new part of the solution space. A validation phase then follows in order to derive an equilibrium of the original game starting from the solution found in the exploratory phase. At each iteration, the tabu-set is increased and this guarantees the termination.

A pseudo-code of our method is shown in Algorithm 2. It takes a game $\Gamma = (S,A)$ as input and returns a set $\mathcal{X}$ of Nash equilibria of Γ. The first two lines of code initialize the solution set $\mathcal{X}$ and our strategy tabu-set $T \subseteq S$ to empty sets (lines 1–2). The algorithm then iterates over the remaining lines of code until the tabu-set equals S. Consider now a generic iteration k over the lines 4–12 and assume $T^{(k)}$ and $\mathcal{X}^{(k)}$ to be the tabu-set and the solution set at iteration k, respectively. In line 4, we create a sub-game $\widetilde{\Gamma}^{(k)} = (S \setminus T^{(k)}, A)$, which is obtained from Γ by removing the strategies in the tabu-set $T^{(k)}$ (function Sub-Game). In line 5, we initialize a mixed strategy $\mathbf{b} \in \Delta_{S \setminus T^{(k)}}$ to the slightly perturbed barycenter of $\Delta_{S \setminus T^{(k)}}$ (function Barycenter), i.e., $b_i \approx 1/|S \setminus T^{(k)}|$ for all $i \in S \setminus T^{(k)}$. We then run InImDyn on $\widetilde{\Gamma}$ starting the dynamics from $\mathbf{b}$ and obtain a Nash equilibrium $\mathbf{y} \in \Delta_{S \setminus T^{(k)}}$ of $\widetilde{\Gamma}$ (line 6). Note that, differently from a peeling-off strategy, we do not insert $\mathbf{y}$ in our solution set $\mathcal{X}$, because it may not necessarily be an equilibrium of the original game Γ. Instead, we inject $\mathbf{y} \in \Delta_{S \setminus T^{(k)}}$ into Δ_S through the following operator:

$$\pi(\mathbf{y})_i = \begin{cases} y_i & \text{if } i \in S \setminus T^{(k)} \\ 0 & \text{else,} \end{cases} \tag{4}$$

and use the obtained mixed strategy as starting point of another InImDyn execution, but this time on the original game Γ, from which we obtain an equilibrium of Γ (line 7). We can now safely add $\mathbf{z}$ to $\mathcal{X}$ at line 8 with the only remark that $\mathbf{z}$ may already be in $\mathcal{X}$ (we use the set-union to implicitly remove duplicates). The last step is to update the strategy tabu-set. To this end we distinguish two possible scenarios: (1) the support of $\mathbf{z}$ is contained in the tabu-set, i.e., $\sigma(\mathbf{z}) \subseteq T^{(k)}$, or (2) it has at least one element in $S \setminus T^{(k)}$. In the first case, we insert in the tabu-set all strategies in the support of $\mathbf{y}$. In the second case, we insert in T all strategies in the support of $\mathbf{z}$.[1]

```
Algorithm 2: Nash equilibria Enumeration Algorithm
   Input: $\Gamma = (S, A)$
   Output: $\mathcal{X}$ a set of Nash equilibria of $\Gamma$
 1 $\mathcal{X} \leftarrow \emptyset$
 2 $T \leftarrow \emptyset$
 3 while $T \neq S$ do
 4     $\tilde{\Gamma} = $ Sub-Game$(\Gamma, S \setminus T)$
 5     b = Barycenter$(S \setminus T)$
 6     y ← InImDyn$(\tilde{\Gamma}, \mathbf{b})$
 7     z ← InImDyn$(\Gamma, \pi(\mathbf{y}))$
 8     $\mathcal{X} \leftarrow \mathcal{X} \cup \{\mathbf{z}\}$
 9     if $\sigma(\mathbf{z}) \subseteq T$ then
10         $T \leftarrow T \cup \sigma(\mathbf{y})$
11     else
12         $T \leftarrow T \cup \sigma(\mathbf{z})$
13 return $\mathcal{X}$
```

Note that Algorithm 2 will stay in the while-loop for at most $|S|$ iterations as stated by the following proposition. Moreover, at the end, it will return a set of Nash equilibria of Γ.

Proposition 1. Algorithm 2 *terminates within* $|S|$ *iterations of the while-loop.*

Proof. Since T is initially empty and the while-loop terminates as soon as $T = S$, it suffices to prove that $T^{(k)} \subset T^{(k+1)}$ at each iteration k, where $T^{(k)}$ denotes the content of T at iteration $k \geqslant 0$ of the while-loop. Note that $\sigma(\mathbf{y}) \setminus T^{(k)} = \emptyset$ by construction of the game $\widetilde{\Gamma}$. Hence, if the algorithm executes line 10 then clearly $T^{(k+1)} = T^{(k)}$- $\cup\, \sigma(\mathbf{y}) \supset T^{(k)}$, whereas if it reaches line 12 then $\sigma(\mathbf{z}) \setminus T^{(k)} \neq \emptyset$ and therefore $T^{(k+1)} = T^{(k)} \cup \sigma(\mathbf{z}) \supset T^{(k)}$. □

Proposition 2. Algorithm 2 *is correct, i.e.,* $\mathcal{X}$ *contains Nash equilibria of* Γ.

Proof. The proposition trivially holds by noting that we add to $\mathcal{X}$ only equilibira returned by InImDyn when executed on Γ, and the fixed-points of InImDyn are Nash equilibria [23]. □

4. Experiments

To demonstrate the quality of our proposed method, we conducted several experiments on both, synthetic and real data sets where we compare to widespread spectral clustering methods. In Section 4.1, we provide results for our cluster enumeration algorithm on a synthetically generated data set. Then, we briefly describe how we integrated our Hough game concept in the Hough Forest framework [4] before we show qualitative and quantitative results on real data sets in Section 4.3.

4.1. Synthetic data experiment

First, we demonstrate the capability of robustly enumerating multiple, overlapping clusters under severe amount of noise and outliers. Instead of deriving the payoff matrix as proposed in Eq. (3), we provide an artificially generated matrix simulating five coherent clusters containing different numbers of elements in the range of 140 to 435. The matrix has a size of 1500 × 1500 and four clusters have mutual overlaps in the range of 9–60% of their elements. The internal cluster affinities were generated using 70% noise while affinities to non-cluster elements were randomly set with 30% of noise. In addition, we added 300% of uniformly distributed noise on top of the whole matrix. Fig. 3 shows the detected clusters, forming color-coded[2] block-wise structures. Our method finds the actual clusters with a mean accuracy of 99.33% in terms of F-measure (100, 99.21, 99.67, 99.42, 98.36, per-cluster accuracy in %) while leaving the rest unassigned. Please note that finding such overlapping clusters is of particular interest since e.g. areas of overlapping objects like persons might be assigned to both individuals.

4.2. Artificial graph matching experiment

In our next experiment we assess the quality of our proposed system by interpreting it as a graph matching problem. Indeed, we can consider the voting element pairs used in the Hough environment as nodes in a graph and their edge weights as a measure for geometric compatibility with respect to a learned reference model (see edge compatibility definitions using Eqs. (1) and (2)).

With this experiment we want to stress the independency with respect to the underlying reference model, i.e. our proposed approach can easily be adapted to work with different model generators. Here, the reference graph was constructed from a set of randomly distributed 2D points in a pre-defined region, simulating voting element origins in the image domain. The query graph (i.e. the underlying points we used for the matching task) was formed by independently disturbing the original point locations, followed by a global transformation on the whole point set (random rotation and translation), providing us with ground truth correspondences.

Given the graphs, we define a simple compatibility measure

$$C(ia, jb) = \exp\left(-\frac{D(ia, jb)^2}{2\sigma^2}\right) \text{ using } D(ia, jb) = \frac{|d_{i,j} - d_{a,b}|}{d_{i,j} + d_{a,b}}, \tag{5}$$

[1] Note that adding just one vertex from $\sigma(\mathbf{y})$ for the first case, or $\sigma(\mathbf{z}) \setminus T$ for the second one, would be enough as well. This indeed may lead to a better coverage of the solution space at the cost, however, of a larger number of iterations.

[2] For interpretation of color in Fig. 3, the reader is referred to the web version of this article.

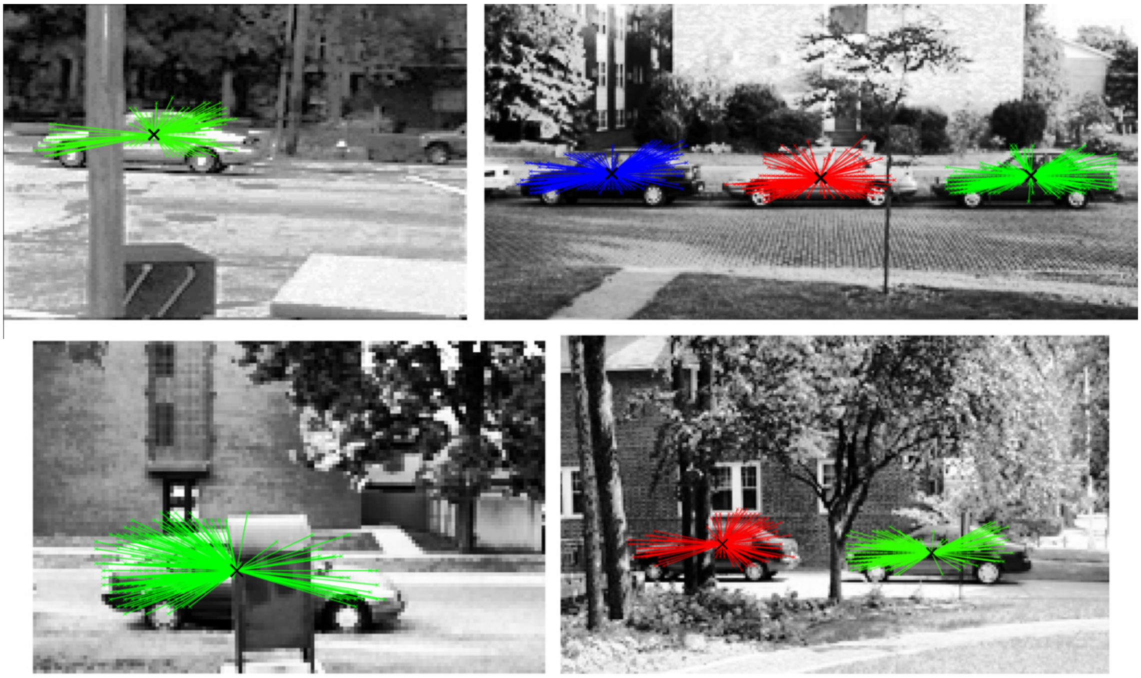

Fig. 5. Car detections and their contributing votes on selected images of UIUC car dataset.

where σ is a normalization parameter and $d_{\cdot,\cdot}$ is the Euclidean distance between point pairs i, j and a, b from the reference and query graphs, respectively.

To demonstrate robustness, we show the impressive noise tolerance of our method by consequently adding outliers to our query graph (drawn uniformly and randomly from the pre-defined region) while analyzing the number of correct matchings. We compare results of our method to two widespread methods for computing the principal eigenvector of C as it is required in spectral methods: power iteration and a variant of the Lanczos method. Each method uses the same difference matrix D as input, analyzing the spatial distances between graph nodes. Please note that it is well-known that the matching quality significantly depends on the normalization of the compatibility matrix and that optimal normalization parameters (here: σ) vary significantly for different optimization approaches. Therefore, we optimized σ for all three methods in an exhaustive search over all matching test cases. We found a high variance in the optimal choice for σ, i.e. 5 for our method, 150 for Lanczos and 200 for power iteration.

Fig. 4 shows average percentage of correct assignments for matching 10 different random reference graphs (each built on 30 points) to query graphs with an increasing number of outliers. As can be seen all methods handle a small amount of outliers in the same reasonable way. However, once the number of outliers becomes significantly higher than the number of points to match, our proposed model shows improved performance, because of its inherent search for an assignment with maximum internal cohesiveness. against outliers.

4.3. Hough Game results

We use the Hough Forest [4] in order to provide the required data to construct the Hough environment as outlined in Section 2 and set $\sigma_h = \sigma_\varphi = 9$. In every Hough tree $t \in \mathcal{T}$, we reduce the set of voting vectors in every leaf node to the median vote vector $\mathbf{d}$. Since the Hough Forest provides multiple trees $\mathcal{T}$ for classification, we update Eq. (3) to $C(i,j) = \sum_{t\in\mathcal{T}} p^t(h(i)|I_i)p^t(h(j)|I_j)p^t_{c_{ij}}p^t_{\varphi_{ij}}$ for computing the compatibilities.

In order to keep the resulting payoff matrices at reasonable size, we constrain the number of considered voting elements to patches with foreground probability $\geqslant 0.5$ and to locations with a gradient magnitude $\geqslant 25$ for pedestrians and $\geqslant 10$ for cars and fibers. Additionally, we consider only pixels lying on a regular lattice with a stride of 2 which massively reduces the amount of data to be processed. The penalization constant has been empirically determined and was fixed to $\alpha = 10$ for all experiments. Please note that sparse implementation techniques are beneficial, since the pairwise interactions are naturally bounded by the largest possible voting vectors obtained from training data.

Unless otherwise stated, we always grow 15 trees with a maximum depth of 12 on 25,000 positive and negative training samples from the referenced data sets. The considered patch size is 16 × 16 and all training samples are resized to a similar scale.

We apply our Hough Game method for localization of cars on the UIUC cars dataset [25] and pedestrians on the extended TUD crossing dataset [8]. In order to demonstrate the broad applicability, we also show qualitative results for challenging task of enumerating paper fibers in microtomy images [26] and mouth detection results on images of the BioID Face Database [27].

4.3.1. UIUC car dataset

In our first experiment we evaluate the proposed method on the single scale UIUC car dataset [25]. The training dataset contains 550 positive and 450 negative images while the test dataset consists of 170 test images showing 210 cars. Although the silhouettes of the cars are mostly rigid, some cars are partially occluded or have low contrast while being located in cluttered background. We achieve a score of 98.5% in terms of equal error rate (EER), hence we are on par with state-of-the-art methods [28,4] while identifying the set of votes that are corresponding to the individual objects. In Fig. 5 we show some sample detections and the groups of votes producing the respective detections. Please note how our method is able to deal with partial occlusions and successfully groups coherent object votes.

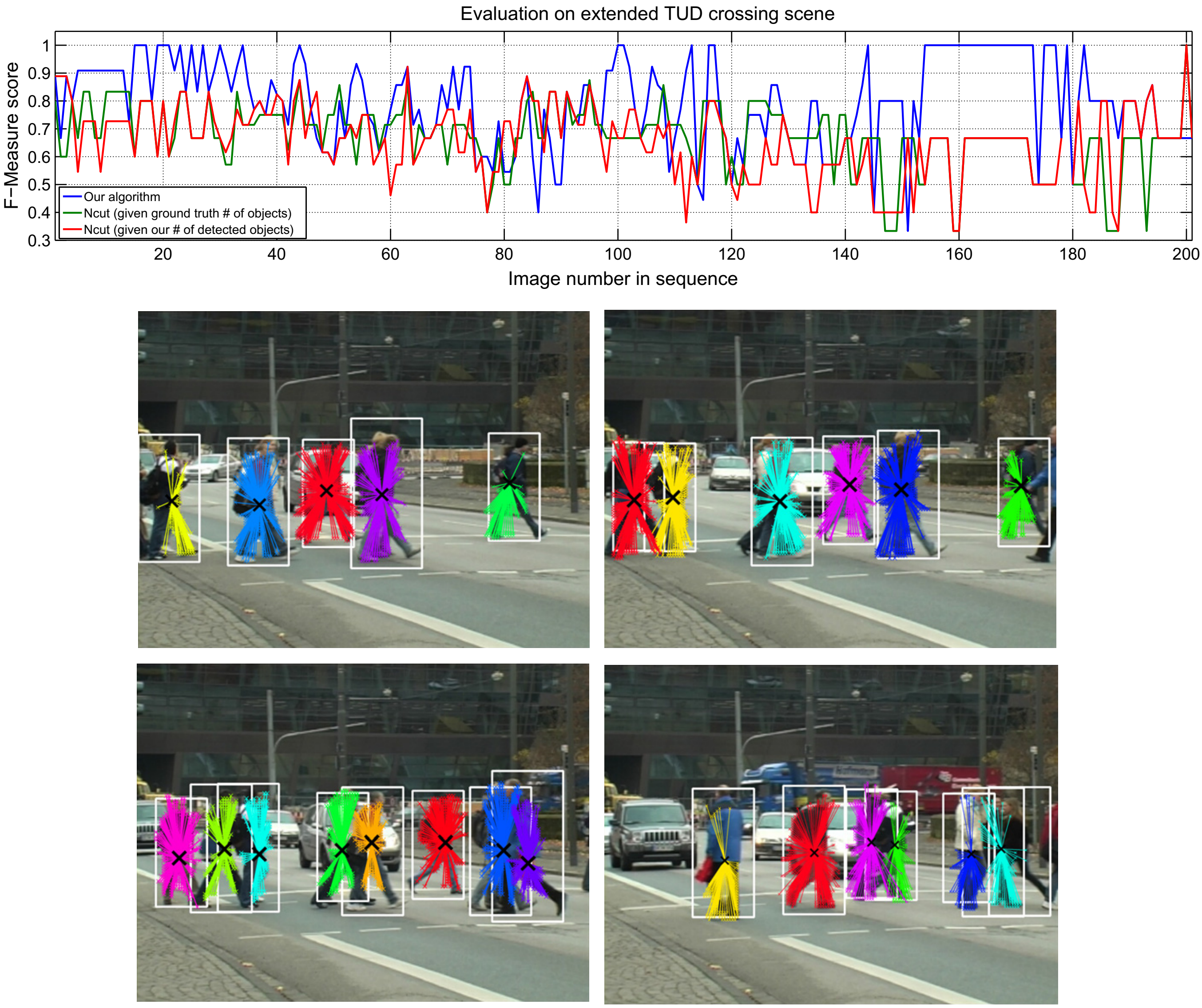

Fig. 6. Top row: Classification results on extended TUD crossing sequence per image using single scale evaluation. We obtain a mean *F*-measure score of 79.88% in comparison to 66.56% and 65.23% for nCut [31] (provided with ground truth # or our detected # of objects, respectively). Second and third rows: Successive and missing (last image) detections of proposed method. White bounding boxes correspond to ground truth annotations. Best viewed in color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.3.2. Extended TUD crossing scene

Next, we evaluated on the extended version of the TUD crossing data base [8], showing several strongly occluded pedestrians walking on a cross-walk. The extended version includes also overlapping pedestrians where head and at least one leg are visible. This results in a very challenging data set consisting of 201 images with 1018 bounding boxes. We used the same training protocol as described in [29]. Since we are not obtaining bounding boxes but rather the sets of contributing voting elements for each person, we decided to evaluate the detection results with the strict criterion introduced in [30]. This criterion accepts detections as correct when the hypothesized center is within 25 pixels of the bounding box centroid on the original scale. In our case, we determined the centroid by taking the median of the reprojected center votes for all detected voting elements. For evaluation, we rescaled the images and the acceptance criterion by a factor of 0.55, such that true positives were counted only within a radius of 13.75 pixels. After constructing the payoff matrices, we played the Hough Game using our novel detection algorithm. To provide a comparison, we handed the same matrices to the widespread normalized cut (nCut) algorithm [31] and illustrate the results (F-measure per test image) in the top row in Fig. 6. Since nCut requires the number of clusters to obtain, we evaluated by giving our number of detections as well as giving the ground truth number of persons. As can be seen, our method outperforms the nCut algorithm, even when the true number of objects is provided. We obtain a mean *F*-measure score of 79.88% compared to 66.56% and 65.23% for nCut provided with ground truth or our detected number of persons, respectively. Since nCut aims for partitioning the whole input into clusters, we tried another setup where we give an additional cluster to the ground truth number and the detected number of persons from our method, respectively. This should allow nCut for partitioning the non-person objects. Before computing the *F*-measure, we removed the detection associated to the lowest eigenvalue. This resulted in *F*-measure scores of 61.94% and 58.79%, hence considerably lower than before and suggesting that nCut does not group noise in an individual cluster but rather incorporates it in the individual detections. The bottom row in Fig. 6 shows color-coded, qualitative results of individual detections of our proposed Hough game. Please note the plausible assemblies of votes from strongly overlapping persons to individual pedestrians, even in the rightmost image, where a person is missed due to assignment of votes to another person in the back. Moreover, it is possible to hypothesize for the person's center by detecting coherent votes of the feet alone (green detection in first image, yellow detection in forth image).

4.3.3. Paper fiber enumeration

In this experiment, we apply our method to detect and enumerate individual paper fibers in microtomy images obtained from serially sectioning a paper specimen [26]. Conventional paper basically consists of a network of thousands of paper fibers, embedded in a filler medium and enclosed by a so-called coating layer. The individual fibers follow no designated pattern when changing their morphology. Moreover, there are fibers sharing their respective fiber walls, making this task an ultimate detection challenge for our enumeration algorithm. In Fig. 7 we show qualitative results of fibers in a cross-section image of eucalyptus paper since no ground truth data is available. Due to a lack of properly labelled data we cannot provide a quantitative evaluation, however, visual inspection of the results reveals that our method can successfully identify many fibers of different shape, despite the adverse imaging conditions.

Mouth localization With this experiment we demonstrate yet another application of our method. The Hough Forest software

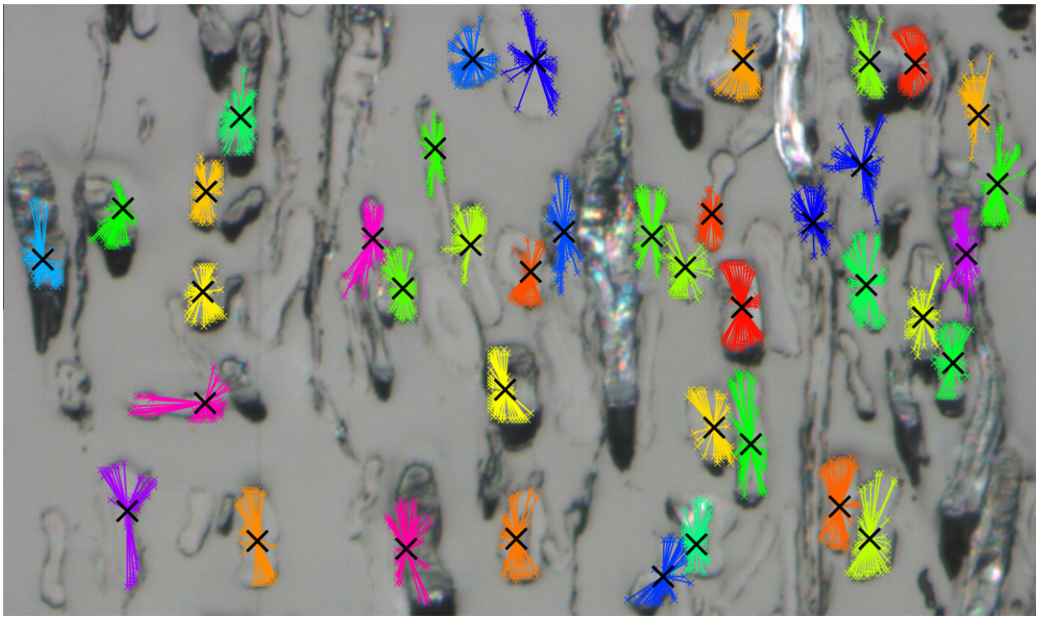

Fig. 7. Enumerating paper fibers in a cross-section image of eucalyptus paper. Differently shaped fibers were successfully assembled using the proposed enumeration scheme.

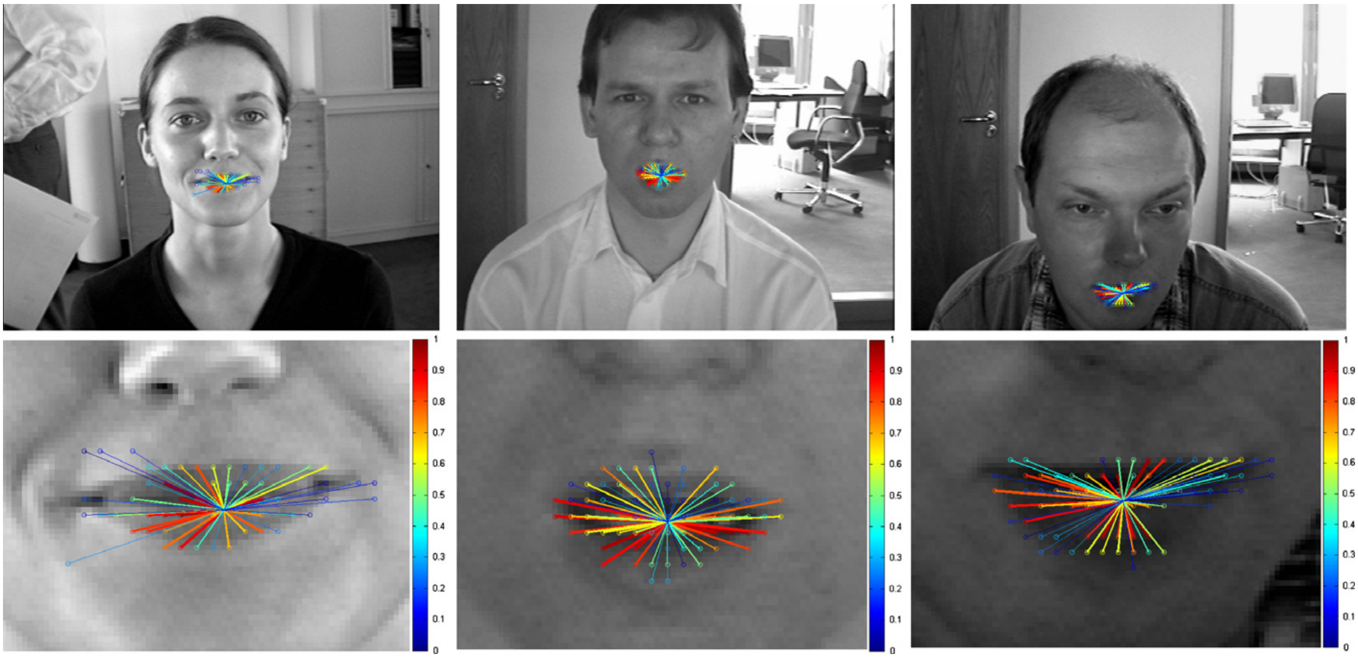

Fig. 8. Selected mouth detections on BioID Face Database. Color-coded voting elements are associated to their individual importance for the extracted coherent sets. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

package provides readily trained trees for mouth detection, presumably those used in [32]. However, we used these trees for evaluating on some selected images of the BioID Face Database [27]. Here, we always took the first set that was found by our enumeration heuristic. The obtained solutions (i.e. the support forming the coherent vote sets) live in the standard simplex and therefore each voting element is associated with a probability, describing its individual importance for the set. In Fig. 8 we illustrate the importance of the individual votes on some sample images. It can be seen that elements truly belonging to the mouth regions are associated with higher probabilities.

5. Conclusions

In this work we showed a novel method to identify multiple, possibly overlapping instances of an object category in a Hough-voting based detection framework. Particularly, we identified individual objects by their sets of mutually compatible voting elements, opposed to analyzing the accumulative Hough Space. We proposed solutions for two challenging problems. First, we introduced the *Hough environment*, where we mapped geometrical compatibilities of votes in terms of mutual object center agreements and orientational similarities. In the second part, we introduced a novel, game-theoretic algorithm for finding multiple objects by assembling pairs of mutually compatible voting elements in the Hough environment. To this end, we designed a *Hough Game* where we modelled the competition of contributing voting elements as Darwinian selection process, driven by geometrical compatibilities with respect to the object category of interest. In contrast to existing methods, our novel detection algorithm can identify several coherent sets with overlapping elements as well as leave non-compatible elements completely unassigned which is particularly helpful in strongly occluded or noisy scenarios. As shown in several experiments, our detection algorithm can successfully cope with severe amounts of noise and outliers.

Acknowledgement

Peter Kontschieder acknowledges financial support of the Austrian Science Fund (FWF) from project 'Fibermorph' with number P22261-N22.

References

[1] P. Hough, Machine analysis of bubble chamber pictures, in: Int. Conf. High Energy Accel. and Instrumentation 1959.
[2] D.H. Ballard, Generalizing the hough transform to detect arbitrary shapes, Pattern Recogn. 13 (2) (1981).
[3] B. Leibe, A. Leonardis, B. Schiele, Robust object detection with interleaved categorization and segmentation, Int. J. Comp. Vis. (IJCV) 77 (1–3) (2008) 259–289.
[4] J. Gall, V. Lempitsky, Class-specific Hough Forests for object detection, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), 2009, pp. 1022–1029.
[5] S. Maji, J. Malik, Object detection using a max-margin hough transform, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), 2009, pp. 1038–1045.
[6] A. Opelt, A. Pinz, A. Zisserman, A boundary-fragment-model for object detection, in: Proc. European Conference on Computer Vision (ECCV), 2006, pp. 575–588.
[7] P. Yarlagadda, A. Monroy, B. Ommer, Voting by grouping dependent parts, in: Proc. European Conference on Computer Vision (ECCV), 2010, pp. 197–210.
[8] O. Barinova, V. Lempitsky, P. Kohli, On detection of multiple object instances using hough transforms, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), 2010, pp. 2233–2240.
[9] J.W. Weibull, Evolutionary Game Theory, MIT Press, 1995.
[10] J. Gall, A. Yao, N. Razavi, L. van Gool, V. Lempitsky, Hough Forests for object detection, tracking, and action recognition, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 33 (11) (2011) 2188–2202.
[11] L. Breiman, Random forests, in: Machine Learning, 2001, pp. 5–32.
[12] Y. Amit, D. Geman, Shape quantization and recognition with randomized trees, Neural Comput. 9 (1997) 1545–1588.
[13] A. Criminisi, J. Shotton, E. Konukoglu, Decision Forests: A Unified Framework for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning, NOW Publishers, 2012.
[14] A. Bosch, A. Zisserman, X. Muñoz, Image classification using random forests and ferns, in: Proc. Intern. Conf. on Computer Vision (ICCV), 2007, pp. 1–8.
[15] A. Criminisi, J. Shotton, D. Robertson, E. Konukoglu, Regression forests for efficient anatomy detection and localization in ct studies, in: MICCAI workshop on Medical Computer Vision: Recognition Techniques and Applications in Medical Imaging, 2010, pp. 106–117.
[16] X. Ren, A.C. Berg, J. Malik, Recovering human body configurations using pairwise constraints between parts, in: Proc. Intern. Conf. on Computer Vision (ICCV), 2005, pp. 824–831.
[17] M. Leordeanu, M. Hebert, R. Sukthankar, Beyond local appearance: Category recognition from pairwise interactions of simple features, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), 2007, pp. 1–8.
[18] A. Torsello, S. Rota Bulò, M. Pelillo, Grouping with asymmetric affinities: a game-theoretic perspective, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), vol. 1, 2006, pp. 292–299.
[19] M. Pavan, M. Pelillo, Dominant sets and pairwise clustering, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 29 (1) (2007) 167–172.
[20] A. Albarelli, A. Torsello, S. Rota Bulò, M. Pelillo, Matching as a non-cooperative game, in: Proc. Intern. Conf. on Computer Vision (ICCV), 2009, pp. 1319–1326.
[21] X. Yang, L.J. Latecki, Affinity learning on a tensor product graph with applications to shape and image retrieval, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), 2011, pp. 2369–2376.
[22] H. Liu, S. Yan, Common visual pattern discovery via spatially coherent correspondences, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), 2010, pp. 1609–1616.
[23] S. Rota Bulò, I.M. Bomze, Infection and immunization: a new class of evolutionary game dynamics, Games Econ. Behaviour 71 (2011) 193–211.
[24] S. Rota Bulò, M. Pelillo, I.M. Bomze, Graph-based quadratic optimization: a fast evolutionary approach, Comput. Vision Image Understand. (2011) 984–995.
[25] S. Agarwal, A. Awan, D. Roth, Learning to detect objects in images via a sparse, part-based representation, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 26 (2004) 1475–1490.
[26] M. Wiltsche, M. Donoser, W. Bauer, H. Bischof, A new slice-based concept for 3d paper structure analysis applied to spatial coating layer formation, in: Proc. of the 13th Fundamental Paper Research Symposium, 2005.
[27] BioID Technology Research. <http://www.bioid.de/>.
[28] C. Lampert, M. Blaschko, T. Hofmann, Beyond sliding windows: object localization by efficient subwindow search, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), 2008, pp. 1–8.
[29] M. Andriluka, S. Roth, B. Schiele, People-tracking-by-detection and people-detection-by-tracking, in: Proc. Intern. Conf. on Computer Vision and Pattern Recogntion (CVPR), 2008.
[30] J. Shotton, A. Blake, R. Cipolla, Contour-based learning for object detection, in: Proc. Intern. Conf. on Computer Vision (ICCV), vol. 1 2005, pp. 503–510.
[31] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 22 (8) (2000) 888–905.
[32] G. Fanelli, J. Gall, L. van Gool, Hough transform-based mouth localization for audio-visual speech recognition, in: Proc. British Machine Vision Conference (BMVC), 2009.