



HAL
open science

Discriminative part model for visual recognition

Ronan Sicre, Frédéric Jurie

► **To cite this version:**

Ronan Sicre, Frédéric Jurie. Discriminative part model for visual recognition. *Computer Vision and Image Understanding*, 2015, <http://www.sciencedirect.com/science/article/pii/S1077314215001642#>. 10.1016/j.cviu.2015.08.002 . hal-01132389v2

HAL Id: hal-01132389

<https://inria.hal.science/hal-01132389v2>

Submitted on 25 Aug 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Discriminative part model for visual recognition

Ronan Sicre and Frédéric Jurie

GREYC, CNRS UMR 6072, University of Caen Basse-Normandie, ENSICAEN, France
Email: {ronan.sicre, frederic.jurie}@unicaen.fr

Abstract

The recent literature on visual recognition and image classification has been mainly focused on Deep Convolutional Neural Networks (Deep CNN) and their variants, which has resulted in a significant progression of the performance of these algorithms. Building on these recent advances, this paper proposes to explicitly add translation and scale invariance to Deep CNN-based local representations, by introducing a new algorithm for image recognition which is modeling image categories as a collection of automatically discovered distinctive parts. These parts are matched across images while learning their visual model and are finally pooled to provide images signatures. The appearance model of the parts is learnt from the training images to allow the distinction between the categories to be recognized. A key ingredient of the approach is a *softassign*-like matching algorithm that simultaneously learns the model of each part and automatically assigns image regions to the model's parts. Once the model of the category is trained, it can be used to classify new images by finding image's regions similar to the learned parts and encoding them in a single compact signature. The experimental validation shows that the performance of the proposed approach is better than those of the latest Deep Convolutional Neural Networks approaches, hence providing state-of-the art results on several publicly available datasets.

Keywords:

Computer Vision, Image Classification, Visual Recognition, Part-based

9 **1. Introduction**

10 The arrival of effective approaches based on Deep Convolutional Neural
11 Networks (Deep CNN), such as the remarkable work of Krizhevsky *et al.*
12 [1] has been perceived as a new trend in image classification, relegating the
13 not so distant approaches such as the bag-of-words [2, 3, 4] or the even more
14 recent Fisher vectors [5] to what some consider now to be a legacy of previous
15 time.

16 Since then, the literature on *image classification* – the task consists in
17 predicting whether an image contains an object or, more generally, a visual
18 concept based on the content of the image – has benefited from a revival of
19 interest because of the new perspective Deep CNN provides (*e.g.* [6, 7, 8, 7, 9],
20 to cite only a few recent of them).

21 However, even if Deep CNN obtain very good performance, most of the
22 recent approaches do not explicitly model objects or scenes as deformable
23 configurations which can potentially result in a lack of robustness to appear-
24 ance/viewpoint changes. One can see this as a limitation, since scenes (and
25 therefore images) can be seen as spatial arrangements of objects or parts,
26 and a decomposition into distinctive parts can results in more expressive and
27 discriminative models [10, 11, 12].

28 One motivation of this paper is hence to bring together the advantages of
29 Deep CNN and part-based model. The results achieved by Oquab[13] *et al.*
30 constitute one interesting step toward that end. They indeed shown that it
31 is possible to transfer image representations learned with CNNs trained on
32 large datasets to different tasks, even in presence of limited training data.
33 Their method uses ImageNet pre-trained layers of CNN to compute mid-level
34 image signature and can be utilized as an efficient feature encoding system.
35 We use this framework as an alternative to Bag-of-words (BOW) or Fisher
36 vector to encode image regions.

37 Another key issue raised by the representation of images in the context

38 of image classification, is how to efficiently use geometric information and,
39 as aforementioned, how to decompose images into stable and distinctive re-
40 gions. While the early works were building on pure bag-of-words *e.g.* [2],
41 which consists of pooling the visual features without using their spatial co-
42 ordinates in any way, it has been shown later (*e.g.* by [4]) that performance
43 can be significantly improved by encoding separately a set of multiple (pos-
44 sibly overlapping) regions, which constitutes a first step toward the use of
45 geometry. Using fixed regions (usually image quad-trees) is obviously limited
46 as the corresponding implicit segmentations of the image is not adapted to
47 the image’s content. Several more recent works such as [3, 14, 15] have intro-
48 duced more flexibility by adapting the shape/position of the regions, but a
49 strong limitation of these works is that the layout of images is still supposed
50 to be fixed, for a given category.

51 The proposed work starts with the observation that images within a given
52 category can have very different layouts or spatial organization, even if they
53 can be interpreted globally as sharing the same meaning. In line with this
54 observation, several recent works have shown that categories can be efficiently
55 represented by a set of distinctive regions either called *parts* or *fragments*
56 [16, 10, 11, 12], see Figure 1. For example, if ‘car’ images can be recognized
57 because of the joint presence of ‘wheel’, ‘road’ or ‘window’-like parts, the
58 position of these regions can be any as long as they are in the image. This idea
59 of introducing some invariance (or alignment) with respect to the position
60 of the parts have been successfully utilized in the Deformable Part Model of
61 [17]. However, in the case of image classification the relative position of the
62 parts is much less constrained than in the case of object detection.

63 In reaction to these observations and concerns, another motivation of our
64 work is precisely to propose a new way to describe images by a set of parts
65 that are aligned across images by construction, without having to use strong
66 geometric constraints between them. This is achieved by proposing a new
67 model for categories, which is based on the fact that (i) a category is defined
68 by a set of K parts (ii) these parts are distinctive in the sense that they

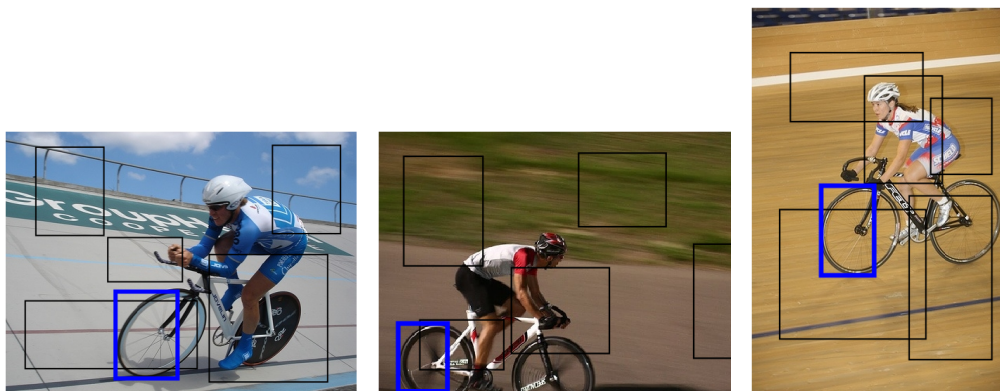


Figure 1: Our system aims at discovering distinctive parts (blue boxes) from a set of regions (black boxes) randomly extracted from images of a category.

69 occur more frequently in the image of the category than in those from other
 70 categories (iii) the presence of regions visually similar to the model's parts
 71 is expected in the images of the category. These definitions are implemented
 72 into an objective function which is optimized during a learning stage. The
 73 objective function relies on a *match* function which automatically discovers
 74 and relates model's parts to image regions. Training can be achieved from
 75 a set of images describing the category to be recognized, without having to
 76 provide any extra annotations. In particular, bounding boxes revealing ob-
 77 jects locations are not necessary. During training, a part classifier is learned
 78 in conjunction with the alignment of parts to image regions. In a second
 79 time, these classifiers can be used to build a global visual descriptor of im-
 80 ages, which combines the signatures of the regions discovered in the image.
 81 More precisely, the paper proposes 3 representations: one is obtained by ag-
 82 gregating the Deep CNN signatures of the different image regions, another
 83 consists in aggregating the scores of individual part classifiers while the third
 84 encodes the distinctive regions of an image with a Fisher vector.

85 The proposed approach is experimentally validated on three classification
 86 datasets. First, Willow [18] aims at classifying 7 human actions in still im-
 87 ages, while the goal of Boats Datasets is to classify 5 different categories of

88 boats. Finally the MIT 67 dataset [19] contains images of 67 types of scenes
89 which are to be recognized. These experiments show that not only the pro-
90 posed method outperform Deep CNN but also that it offers state-of-the-art
91 results on the very competitive MIT 67 dataset.

92 The rest of the paper is organized as follows. Related work is presented in
93 Section 2, while Section 3 provides details on the proposed system that learns,
94 aligns, and encodes distinctive parts. Finally, the experimental validation is
95 given in Section 4, before concluding the paper.

96 **2. Related Work**

97 *Image classification.* has received a large attention from the computer vision
98 community, *e.g.* see the abundant literature related to the Pascal VOC [20]
99 and ImageNet [21] challenges. A large part of the modern approaches follow
100 the bag-of-word model [2], composed of a 4 step pipeline: 1) extraction of
101 local image features, 2) encoding of local image descriptors, 3) pooling of
102 encoded descriptors into a global image representation, 4) training and clas-
103 sification of pooled image descriptors for the purpose of object recognition.
104 Several studies evaluated the influence of the first step: the low level features
105 *e.g.* gradient, shape, color, and texture descriptors, such as [22], while other
106 proposed combining different levels (low - mid - high) of information [23].
107 Regarding the second step: image encoding; Fisher vectors [5] were consid-
108 ered as achieving state-of-the-art performance, in many cases. The third,
109 pooling, step is also shown to provide improvements, and spatial and feature
110 space pooling techniques have been widely investigated [24, 4]. Moreover,
111 [3, 14] have recently proposed two different strategies for embedding spa-
112 tial information into the bag-of-words framework. Finally, regarding the last
113 step of the pipeline, discriminative classifiers such as Support Vector Ma-
114 chines (SVM) are widely accepted as the reference in terms of classification
115 performance.

116 During the last months, the deep CNN approaches have been successfully
117 applied to large-scale image classification datasets, such as ImageNet [21] [1],

118 obtaining state-of-the-art results significantly above Fisher vectors or bag-
119 of-words approaches. These networks have a much deeper structure than
120 standard representations, including several convolutional layers followed by
121 fully connected layers, resulting in a very large number of parameters that
122 have to be learned from training data. By learning these networks parameters
123 on large image datasets, a structured representation can be extracted at an
124 intermediate to a high-level, depending on the extracted layers [25, 26]. Deep
125 CNN representation have been recently combined with VLAD descriptors [27]
126 or Fisher vectors [9]

127 *Mid-level features.* Several authors have shown the importance of adding
128 intermediate representations [28], also referred as the mid-level features, for
129 leveraging the performance. We observe three mains trends of mid-level
130 description in the recent literature: hand-crafted, learned, and unsupervised
131 features. *Hand-crafted mid-level features* aim at encapsulating information
132 on groups of pixels such as superpixels [29, 30], patches [31] or segments
133 [32]. These descriptors are computed similarly for any given image and do
134 not require any learning. On the other hand, a large variety of *learned*
135 *mid-level features* have been proposed. One of the original method was the
136 Deformable Part Model, proposed by [17]. We can also mention the semantic
137 attributes [33, 34] which have received a lot of interest. Within the learned
138 mid-level features techniques, we observe a large variety in terms of learning
139 data utilized. While some feature are based on extra training data such as
140 labeled fragments [35], sketch tokens [36] or pre-trained object detectors [37],
141 most methods use a standard split of training and testing data to learn the
142 distinctive features, as the *structural element patch model* [38] or the *blocks*
143 *that shout* [11]. Finally, regarding *unsupervised mid-level features*, the work
144 of [39] aims at detecting distinctive patches in an image dataset without any
145 label information.

146 *Learned mid-level features.* Our work aims at learning distinctive parts with-
147 out extra annotations. Therefore, closely related work includes the De-
148 formable Part Model (DPM) [17]. The DPM models categories by using

149 a mixture of parts and classify image regions as object vs non object regions.
150 Classifiers are applied to a representation in which the parts are aligned, by
151 shifting the parts with respect to the root filter. However, for image classifi-
152 cation, the variability of parts positions as well as the variation of appearance
153 within a category makes the problem different. Our work also bears simi-
154 larities with [16], which tries to discover the *fragments* that maximize the
155 mutual information between the category and the presence of the fragment
156 in the image. However, [16] suffers from that (i) contrarily to [17], part are
157 just image patches and not discriminative classifiers (ii) the decision is made
158 by verifying the presence of the fragments in the image, instead of training
159 a classifier taking fragment descriptors as input. Our approach takes the
160 advantages of both approaches without having their drawbacks.

161 More recently, [10] proposes a learning framework for the automatic dis-
162 covery of image's parts, assuming that partial correspondence between in-
163 stances of a category are available. These partial correspondences allow the
164 training of part detectors, used in a first time to extract candidates regions.
165 While we share the same motivations, our approach does not require any
166 supervision. In addition, it is worth mentioning [11] and [12] which both
167 propose algorithms for learning parts that are good representatives of a given
168 category. In the same way, [40] proposed a part localization model leveraging
169 Deep CNN features computed on bottom-up region proposals, by learning
170 part appearance models and enforcing geometric constraints between parts.
171 Our work follows the same objectives, without the localization constraints
172 imposed by [12, 40] and the large computation requirement and unoptimized
173 encoding of [11]. This work finally shows the importance of mid-level infor-
174 mation and justifies its use to improve recognition capabilities.

175 The automatic discovery of distinctive parts is a very active area explain-
176 ing that some closely related papers have been published since the submis-
177 sion of this article. The very recent work of Parizi *et al.* [41] shares the
178 same objectives than ours and proposes to learn a part-based model by si-
179 multaneously learning an image classifiers and a set of shared parts. Three

180 different recent papers have explored how Deformable Part Models and Deep
181 CNN can be combined: [42] shows that a DPM can be formulated as a
182 CNN, thus providing a synthesis of the two ideas, [43] also integrates the non
183 maximal suppression phase in the CNN architecture, while [44] proposes a
184 deformation-constrained pooling layer designed to learn shared visual pat-
185 terns and their deformation properties for multiple object classes.

186 This article is an extension of [45] providing a richer description of the
187 related works, more details and improvement of the method as well as a much
188 more experimental validation.

189 **3. Proposed method**

190 From a general point of view, the overall approach consists in three steps:
191 (i) a learning step during which some category’s distinctive parts are discov-
192 ered, (ii) a representation step in which a global signature of the image is
193 computed, on the basis of parts presence in the image, (iii) a classification
194 step relying on a linear SVM classifier. The originality of the work is in
195 the discovery of category’s distinctive parts and their use in the encoding of
196 images (two first steps), which are the subject of this section, and not in the
197 classification step which is the most classic.

198 The model we propose for representing image categories consists in a
199 collection of K distinctive parts defined by their visual appearance, without
200 any geometric relationships between them. It is expected that positive images
201 (with respect to a given category) contain regions visually similar to these
202 parts (considered as instances of the parts) while there are fewer of them in
203 negative images. The distance from an image to the class is then defined as
204 a function of the set of distances between image regions and parts (*e.g.* using
205 max pooling).

206 As aforementioned, the main contribution of this paper lies in the method
207 allowing to automatically discover distinctive parts in the images of a given
208 category and thus to learn the model of this category. These parts are further
209 aligned with images regions, which are utilized to produce images signatures.

210 Signatures are subsequently used in a standard classification framework.

211 This section first presents our part-based model and its associated cost
212 function, which is to be optimized during learning. In a second time, we
213 explain how the parameters of the model can be learned using an iterative
214 framework inspired from the *softassign* algorithm. Then, more details are
215 given on the algorithm initialization step. Finally, we explain how images
216 signatures can be computed using the learned model.

217 3.1. Part model and objective function

218 First, let us introduce some notations. We assume having a set of images
219 belonging to the category to be modeled, considered as positive training
220 images and denoted as \mathcal{I}^+ . $|\mathcal{I}^+|$ represents the number of positive images. In
221 the same way, \mathcal{I}^- is the set of (negative) images belonging to other categories.
222 The whole training set is denoted as $\mathcal{I} = \mathcal{I}^+ \cup \mathcal{I}^-$ and contains $|\mathcal{I}|$ images.
223 From each image $I \in \mathcal{I}$, we extract a dense random set of image regions
224 denoted as \mathcal{R}_I . Each region r is represented by its signatures x_r , which is,
225 in practice, the bag-of-word or CNN, representation of the region. More
226 details on the description choices are discussed in section 3.4. The model
227 of the category includes a set of parts denoted as \mathcal{P} . The number of parts,
228 $K = |\mathcal{P}|$, is fixed. In the following, $p \in \mathcal{P}$ denotes one of these parts.

229 As explained before, our model relies on three assumptions: first, the
230 model is supposed to be composed of a set of K different parts. Second, it
231 is expected that each part of the model is present in each positive image.
232 Third, parts should be representatives of the category, which means that
233 they should occur more frequently in positive images than in negative ones.

234 We implement the second constraint by introducing the match function
235 $m(r, p)$ associating model parts and image regions, and by imposing that
236 $\forall \mathcal{I} \in \mathcal{I}^+ r \in \mathcal{I}$ and $\forall p \in \mathcal{P}$, $\sum_{r \in \mathcal{I}} m(r, p) = 1$. The match function is defined
237 as:

$$m(r, p) = \begin{cases} 1 & \text{if region } r \text{ is assigned to part model } p \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

238 In practice, the match function can be seen as a binary matrix with one
 239 row per part and one column per image region. We add the first constraint
 240 ensuring that an image region can be assigned to at most one part, which is
 241 written as: $\forall I \in \mathcal{I}^+, \forall r \in I, \sum_{p \in \mathcal{P}} m(r, p) \leq 1$.

242 Regarding the third assumption, which states that regions should be dis-
 243 criminative, one way to achieve this would be to measure to which extent
 244 each part can be matched with regions from the negative set, and promote
 245 those occurring more on positive images. However, such process would be
 246 very costly. Therefore, as suggested by [11], we use the LDA technique of
 247 [46], which consists in learning once and for all a universal model of negative
 248 patches. We note that our method differs largely from the one proposed in
 249 [11]: Our description, initialization, and learning methods are totally differ-
 250 ent. However, we share the same goal of discovering parts representative of
 251 a class, as well as using the LDA technique and using a similar encoding
 252 of parts response. In practice, the parameter vector w of a part classifier,
 253 corresponding to the part p , is defined simply as:

$$w(p, m) = \Sigma^{-1} \left(\frac{\sum_{r \in I, \forall I \in \mathcal{I}^+} m(r, p) \times x_r}{\sum_{r \in I, \forall I \in \mathcal{I}^+} m(r, p)} - \frac{\sum_{r \in I, \forall I \in \mathcal{I}} x_r}{|r \in I, \forall I \in \mathcal{I}|} \right), \quad (2)$$

254 where Σ is the covariance matrix obtained by taking the whole set of re-
 255 gions from both positive and negative images. Consequently, the part models
 256 $w(p, m)$ are fully defined once the match function is defined. In addition, the
 257 similarity between a region r and a part p of the model can be computed as
 258 $w^T(p, m) \times x_r$.

259 The model is thus fully defined by giving the match function $m(r, p)$. Fol-
 260 lowing the afore mentioned constraints, we define the optimal match function,
 261 denoted as \hat{m} , as the one maximizing:

$$\left\{ \begin{array}{l} \hat{m} = \arg \max_m \sum_{p \in \mathcal{P}} \sum_{I \in \mathcal{I}^+} \sum_{r \in I} m(r, p) \times w^T(p, m) \times x_r \\ s.t. \forall I \in \mathcal{I}^+, \forall p \in \mathcal{P}, \sum_{r \in I} \hat{m}(r, p) = 1 \\ s.t. \forall I \in \mathcal{I}^+, \forall r \in I, \sum_{p \in \mathcal{P}} m(r, p) \leq 1. \end{array} \right. \quad (3)$$

262 Learning this model hence consists in the (combinatoric) optimization
 263 of Eq. (3). Finding the global optimum is not computationally feasible,
 264 nevertheless we propose to adapt the point matching algorithm of [47] to
 265 obtain an approximate solution, as explained in the following section. This
 266 algorithm was first introduced to solve simultaneously the correspondence
 267 problem as well as the pose estimation of 3D and 2D data. In [47], two sets
 268 of points X_j and Y_k are related by a geometric transformation. Both sets
 269 can contain outliers. The *match matrix* m_{jk} is defined as the correspondence
 270 matrix such that $m_{jk} = 1$ if point X_j corresponds to point Y_k and 0 otherwise.
 271 The problem is further presented as finding the *pose* (*i.e.* the geometric
 272 transformation) and the corresponding match matrix m_{jk} that best relates
 273 the two sets of points. These two problems are finally solved simultaneously
 274 using an iterative process aiming at minimizing an energy function.

275 3.2. Learning with softassign

276 Now, our main goal is to efficiently find a good (sub-optimal) solution
 277 of the objective function given by Eq. (3). If we ignore, for the moment,
 278 the inequality constraint (last constraint of Eq. 3), then the match matrix
 279 m can be seen as a permutation matrix. We use the *deterministic annealing*
 280 method of [48] to turn our combinatoric problem into a continuous one, mak-
 281 ing the optimization simpler and more efficient. The key idea is to minimize
 282 a sequence of objective functions controlled by a parameter β representing
 283 the inverse temperature of the system. By increasing the parameter, the
 284 objective functions leans towards the discrete function.

285 The constraints are then relaxed from a permutation matrix constraints
 286 to *doubly stochastic matrix* constraints, meaning that every rows and columns
 287 of the matrix should sum up to 1 (see [47] for more explanations). Therefore,
 288 the computation of the match function can be achieved iteratively using the
 289 *softmax* formulation:

$$\forall I \in \mathcal{I}^+, \forall r \in I, m(r, p) = \frac{\exp(\beta \times w^T(p, m^*) \times x_r)}{\sum_{r' \in I} \exp(\beta \times w^T(p, m^*) \times x_{r'})}. \quad (4)$$

290 Where $w(p, m^*)^T \times x_r$ is the score function relating the similarity between
 291 the part p and the region r of the image I , using the match function m^*
 292 computed at the previous iteration. Such a formulation does produce values
 293 in the interval $[0, 1]$, which is expected. Furthermore, when $\beta \rightarrow \infty$, there
 294 will be one region per image for which $m(r, p) = 1$, while for the other ones
 295 $m(r, p) = 0$, therefore satisfying the first constraint.

296 However, we utilized the following formulation, which improves numerical
 297 stability. $\forall I \in \mathcal{I}^+$ and $\forall r \in I$,

$$m^\dagger(r, p) = \exp(\beta((w^T(p, m^*) \times x_r) - \max_{\forall r' \in I} (w^T(p, m^*) \times x_{r'}))). \quad (5)$$

298 In addition, the match matrix m has also to satisfy the doubly stochastic
 299 constraints. This can be achieved by using Sinkhorn (see more details in
 300 [47]), by iteratively normalizing rows and columns, see Algorithm 1.

301 Up to this point, we ignored the inequality constraint stating that $\forall I \in \mathcal{I}^+$
 302 and $\forall r \in I$, $\sum_{p \in \mathcal{P}} m(r, p) \leq 1$. Gold *et al.* [47] turned the inequality con-
 303 straint into an equality constraint by adding a slack variable [49]. However,
 304 unlike Gold *et al.* [47], our problem is not symmetrical. In order to handle the
 305 inequality constraint, we add non-linearities to the process by setting to zero
 306 the very low values (*inferior to 10^{-7} in practice*), of m^\dagger right after its calcula-
 307 tion, see Equation 5. Then, the following process is the normalization, which
 308 distributes the weights, except for the null values that remain unchanged.
 309 Therefore, the normalized match-matrix satisfies the previous constraints:
 310 $\forall I \in \mathcal{I}^+, \forall p \in \mathcal{P}, \sum_{r \in I} m^\dagger(r, p) = 1$ and $\forall I \in \mathcal{I}^+, \forall r \in I, \sum_{p \in \mathcal{P}} m^\dagger(r, p) \leq 1$.
 311 For example, if a region obtains a very low score for all parts, a column of
 312 the match-matrix $m^\dagger(r, p)$ is set to 0. In other words, image regions not
 313 matching any parts, *with very low scores*, will not contribute to any parts
 314 and while the parameter β is increased the selection will be more strict and
 315 more regions will be discarded. This process further allows to speed up the
 316 normalizations in the algorithm.

317 *3.3. Providing initial correspondences between parts and image regions*

318 The learning process allowing to learn distinctive parts by iteratively
 319 refining the match function m , as presented in the previous section, is a
 320 process requiring to know m from the previous iteration, and hence raises
 321 the question of the initialization of m .

322 It seems reasonable to think that because the optimization process is not
 323 convex, the algorithm will perform better if the initial part to regions cor-
 324 respondences already involve discriminative regions. To select these initial
 325 discriminative regions, we first extract the signatures x_r of the regions sam-
 326 pled from positive training images. These signatures are then clustered, using
 327 K-means. Then, we use again the LDA acceleration of [46] to learn initial
 328 classifiers. For each cluster, the classifier w is defined as $w = \Sigma^{-1}(\bar{x} - \mu_0)$
 329 where \bar{x} is the average of the signatures within the cluster and μ_0 and Σ the
 330 overall mean and covariance matrix.

331 These classifiers are further applied on the regions of the training images.
 332 Maximum responses to the classifiers are then selected per image and aver-
 333 aged over positive and negative subsets, giving us the two scores s^+ and s^- ,
 334 for a given cluster j , defined as:

$$\begin{aligned} s_j^+ &= \frac{1}{|\mathcal{I}^+|} \sum_{r^* \in \mathcal{I}^+} w_j^T x_{r^*} \\ s_j^- &= \frac{1}{|\mathcal{I}^-|} \sum_{r^* \in \mathcal{I}^-} w_j^T x_{r^*}. \end{aligned} \quad (6)$$

335 Where $\forall I \in \mathcal{I}^+$, $r^* = \arg \max_{r \in I} (w_j^T x_r)$. Then, we denote as C_p the K
 336 clusters having the largest s_j^+ / s_j^- ratios, which are selected as initial discrim-
 337 inative regions. These initial regions are further used to compute the initial
 338 part classifier $w(p, m_0)$ as : $w(p) \leftarrow \Sigma^{-1}(C_p - \mu_0)$, used to compute the initial
 339 match matrix $m_0(r, p)$.

340 *3.4. Computing region and image signatures*

341 *3.4.1. Image region signatures*

342 First, we would like to comment on the patch signatures x_r , used in the
 343 learning process. We note that these descriptors must be compact, *i.e.* no

344 more than a few thousand dimensions, to allow the learning to be effective.
 345 In fact, we remind that each image is represented by a few thousand of these
 346 patches, or regions. Therefore, we first used the simple BOW description
 347 using $k = 1000$ clusters, as in [12]. Later, following [25], we extracted the
 348 seven-th layer of the CNN representation. This intermediate representation
 349 offers much higher results, as we can see in section 4, and allows a better
 350 comparison to the current best performing methods. Therefore, we build
 351 two systems, or pipelines, the standard one or BOW based on SIFT and the
 352 CNN-based pipeline.

353 3.4.2. Image signatures

354 Once the model is learned, images signatures can be computed using the
 355 distinctive parts of the model. Let us denote as I an image to be encoded. We
 356 first extract a set of random regions $r \in I$ and compute their corresponding
 357 descriptors x''_r . We can measure to which extent each region is similar
 358 to one of the model parts by using the scoring function defined previously
 359 by Eq. (2) as $w^T(p, m) \times x_r$, where m is the match function learned during
 360 training. Then, we pool the per part similarities to produce a global signature
 361 of the image.

362 We propose three different pooling/encoding strategies: the Bag-of-parts
 363 inspired from [11] and two novel approach so-called the *Fisher-on-parts* and
 364 the *CNN-on-parts*.

365 *Encoding images with Bag-of-parts.* To compute the bag-of-parts (BOP), the
 366 per parts scores are computed for each extracted region on an image. The
 367 signature of the image is then given by aggregating, for each part of the
 368 model, the average and the maximum of the region scores over the image.
 369 Namely, if p_j is one of the K parts of our model, the signature of the image
 370 I will be represented by the two following components:

$$\frac{\sum_{r \in I} w^T(p_j, m) \times x_r}{|r \in I|} \quad \text{and} \quad \max_{r \in I} w^T(p_j, m) \times x_r. \quad (7)$$

371 When the problem is a multi-class problem, we do the same for each class
372 and aggregate the results. Therefore, we obtain a $2 \times K \times C$ -dimensional
373 descriptor, where C is the number of classes.

374 *Encoding images with Fisher-on-parts.* Fisher-on-parts (FOP) aims at en-
375 coding together the maximum response of each parts in an image I_t . As
376 in BOP, scores are computed for each region. Then, instead of aggregating
377 average and maximum scores as for the BOP, the maximum scoring region
378 r^* for the part p is selected, as follows:

$$r^* = \arg \max_{r \in I} w^T(p, m) \times x_r. \quad (8)$$

379 Finally, a Fisher vector is computed on the area of the image covered by the
380 K selected regions r^* . Therefore, the final FOP descriptors is $2 \times G \times D \times C$ -
381 dimensional vector, where G is the number of Gaussian in the mixture model
382 of the Fisher vector, D is the dimensionality of SIFT descriptors and C the
383 number of categories.

384 *Encoding images with CNN-on-parts.* In this case, regions are encoded with
385 CNN features and scores are obtained for each region of an image, as for
386 the Bag-of-part signature. For each part or the model, the region giving
387 the highest score (see previous paragraphs) is selected and it's descriptor
388 kept. All the descriptors so selected are further concatenated resulting in a
389 $D^* \times K \times C$. Where $D^* = 4096$ is the dimension of the CNN descriptor.

390 4. Experiments

391 This section presents an experimental validation of the proposed ap-
392 proach. We start by describing the datasets used in our experiments; then we
393 introduce some baseline algorithms used for comparison purposes and give
394 the details of our implementation; finally the performance obtained with the
395 proposed approach are exposed and compared to baselines and state-of-the
396 art algorithms.

Initialization: $w(p) \leftarrow \Sigma^{-1}(C_p - \mu_0)$

while $\beta \leq \beta_f$ **do**

while m^\dagger not converged or # of iteration $\leq I_0$ **do**

update match matrix by softassign

Compute $m^\dagger(r, p)$, based on Eq. 5

while \hat{m}^\dagger not converged or # of iteration $\leq I_1$ **do**

$\forall I \in \mathcal{I}^+$

Update \hat{m} by normalizing rows

$\hat{m}_1^\dagger(r, p) \leftarrow \frac{\hat{m}_0^\dagger(r, p)}{\sum_{r \in I} \hat{m}_0^\dagger(r, p)}$

Update \hat{m} by normalizing columns

$\hat{m}_0^\dagger(r, p) \leftarrow \frac{\hat{m}_1^\dagger(r, p)}{\sum_{p \in \mathcal{P}} \hat{m}_1^\dagger(r, p)}$

end

update parts using LDA

Compute $w(p, m_0^\dagger)$, based on Eq. 2.

end

$\beta \leftarrow \beta_r \beta$

end

Algorithm 1: Algorithm for learning the mach function.

397 *4.1. Datasets*

398 Three classification datasets are utilized to experimentally validate the
 399 proposed approach: The Willow actions dataset [18], the Boats Dataset, and
 400 the MIT 67 scenes dataset [19].

401 *The Willow actions dataset [18]* is a dataset for action classification on
 402 unconstrained consumer images from the Internet. The dataset contains 911
 403 images split into 7 classes of common human actions, e.g. ‘running’, ‘riding
 404 cycle’, etc. There are at least 108 images per actions, with 70 images used as
 405 training and the rest as testing images. We note that the dataset also offers
 406 bounding boxes fitted on humans performing the actions. In our case, we
 407 perform the test *without* using these bounding boxes, as we want to detect
 408 the relevant parts of images automatically without any prior knowledge on
 409 the scenes.

410 *The MIT 67 scenes dataset [19]* is composed of 67 categories of indoor
411 scenes. These categories include stores (e.g. bakery, toy store), home (e.g.
412 kitchen, bedroom), public spaces (e.g. library, subway), leisure (e.g. restau-
413 rant, concert hall), and work (e.g. hospital, TV studio). Some scenes can be
414 best characterized by their global layout (corridor), or by the objects they
415 contain (bookshop). Each category has around 80 images for training and
416 20 for testing.

417 *The RECONSURVE Boats Classification Dataset*¹ is composed of 2,877
418 images divided in 5 categories of boats (e.g. boating, fishing, merchant ship,
419 tanker, passenger).

420 In the following, the performance on the three datasets is measured using
421 the mean Average Precision (mAP).

422 4.2. Comparisons to baseline approaches

423 Our approach is compared to different state-of-the-art approach of the
424 literature.

425 On one hand we report results obtained with Bag-of-words and Fisher
426 vectors computed on the dense root SIFT, on the whole image (see [50] for
427 details). In addition, we used Fisher vectors computed on spatial pyramids,
428 using the two first layers *i.e.* 1×1 and 2×2 segments. A SVM classifier is
429 then trained on the train set and applied to the test images, following the
430 standard procedures for such image classification tasks.

431 We introduce a second baseline inspired by the work of [13]. A CNN
432 is first trained with CAFFE on ImageNet (for experiments on the Willow
433 action dataset) or ImageNet+Places datasets [26] (for the MIT 67 Scenes),
434 and the penultimate layer of the network is used as an image descriptor.
435 The images of the target dataset (*i.e.* MIT 67 or Willow) are then encoded
436 using this CNN-based descriptor and processed with a standard linear SVM
437 classier framework.

¹can be downloaded from <https://jurie.users.greyc.fr>

438 *4.3. Details of our classification pipeline*

439 As explained in the previous section, the proposed algorithm relies on
440 two steps: a first step in where the parts are learned and a second one
441 in which a global signature of the image is computed, using the selected
442 parts. In the first stage, two different encodings of the regions are evaluated
443 (SIFT based bag-of-words and CNN features). In the second steps 3 different
444 encoding/pooling strategies are considered: (i) Fisher-on-parts consisting in
445 computing SIFT-based Fisher vectors on the selected regions, (ii) Bag-of-
446 parts in which the scores of each part classifier are aggregated to form the
447 image descriptor and finally (iii) CNN-on-parts in which the CNN descriptors
448 of each region are concatenated to form the image descriptor (see section 3.4
449 for a description of these image descriptors).

450 Once images descriptors are computed they are processed by a linear
451 SVM such as usually done for these classification tasks. We remind that
452 the originality of the work is in the encoding of the image and not in the
453 classification step which is standard.

454 In the following paragraphs, we give the details of the implementation
455 used in this experimental validation.

456 *Extraction of image regions.* For each image, a set of regions is generated by
457 randomly sampling 2,000 regions per image, over the entire image. We note
458 that for the CNN-based pipeline, only 1,000 regions are extracted to save
459 time. The scale and aspect ratio of these regions are randomly chosen, but
460 regions are constrained to have a size of at least 5% of the image size and
461 aspect ratio should belong to $[0.5; 2]$.

462 *Regions descriptors.* As said above, two types of regions descriptors con-
463 sidered. For the BOW-based region descriptors, dense SIFT features are
464 extracted within the regions to be encoded, using VLFEAT [51]. We use the
465 default 4 scales, and sample points every 3 pixels. The SIFT features are
466 further square-rooted to get rootSIFT features and the feature dimension is
467 reduced to 80 using PCA, as suggested by [50]. Then each region is charac-

Table 1: Results for the (SIFT) Bag-of-parts showing the influence of the initialization and the optimization processes, on the Willow dataset.

Method	BOP on salient regions	BOP (random init)	BOP (proposed approach)
mAP	0.467	0.460	0.510

468 terized using a 1,000-dimensional bag-of-word. These choices are standard
 469 for this type of problem [50].

470 Regarding the CNN descriptors, we use the 7-th layer of the CNN pro-
 471 posed by [52], resulting in a 4,096-dimensional vector. For the experiments
 472 on the Willow action dataset, we use the standard CAFFE CNN architecture
 473 [52] trained on ImageNet. For those on the MIT 67 Scenes dataset, we use
 474 the hybrid architecture [26] trained on ImageNet and on the Places dataset.
 475 Note that the same description method is used to compute region descriptors
 476 within the CNN-on-parts descriptor (such as defined section 3.4).

477 *Parameters of the learning algorithm.* Regarding the learning algorithm, we
 478 empirically set the parameters as suggested by [47]: $\beta = 0.41$, $\beta_r = 1.245$,
 479 $\beta_f = 1.2$, $I_0 = 4$, $I_1 = 30$ (see Algorithm 1 for the definition of these
 480 parameters). The algorithm iterates over the estimation of m until the sum
 481 over m of the absolute difference between two iterations is smaller than $\epsilon =$
 482 0.005.

483 4.4. Results

484 In this section, we first comment on the quantitative results then show
 485 some qualitative results, i.e. visualization of learned parts, in Figures 4 and
 486 5. As said above, the performance is measured using the mean Average
 487 Precision (mAP).

488 *Initial matchings between model parts and image regions.* First, we evalu-
 489 ate the impact of the initialization step in the part-learning process, on the
 490 Willow dataset. Results are given Table 1. The objective is to measure the

Table 2: Performance of the CNN-based Bag-of-parts descriptors. Left-hand side: using salient regions, Right-hand side: using the proposed learning approach.

Method	BOP on salient regions	BOP (proposed approach)
Willow	0.656	0.766
MIT 67	0.555	0.788

491 contribution of the initial set of correspondences between parts and regions,
 492 such as described in section 3.3, and to compare it with a simple random ini-
 493 tialization of the parts/correspondences. If we randomly initialize the match
 494 function we observe a mAP of 46.0% (with the SIFT based Bag-of-parts
 495 encoding). Adding the proposed initialization (based on salient regions) im-
 496 proves the mAP by 5% (51.0%).

497 In addition, to prove the usefulness of the proposed model versus a simple
 498 selection of discriminative regions, we evaluated the performance obtained
 499 by initializing the match function with salient regions (using the method
 500 proposed in section 3.3), without performing any subsequent optimization,
 501 *i.e.* without learning m but keeping the correspondences between parts and
 502 salient regions as they initially are. If we just use the salient regions, the
 503 performance drops to 46.7%; we did the same observation for the CNN-based
 504 pipeline, see Table 2 and Figure 3.

505 The experiments demonstrate that the proposed algorithm improves sig-
 506 nificantly over a simple selection of discriminative parts, and that a good
 507 initialization of our algorithm is better than a random initialization of the
 508 part to region correspondences.

509 *Convergence.* Theses experiments aim at understanding how the match ma-
 510 trix converges towards a sparse matrix hard-assigning regions to parts. Figure
 511 2 represents this convergence process, by showing the ratio $\frac{\sum m_{i,j}^2}{K \times |\mathcal{I}^+|}$, where K
 512 is the number of parts and $|\mathcal{I}^+|$ the number of positive images. The ratio
 513 should be of 1 in case of hard assignments. We note that there is a consistent

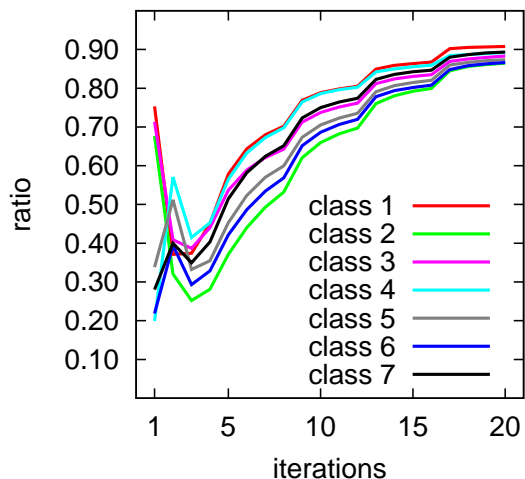


Figure 2: Convergence of the match matrix m for each class of the Willow actions dataset.

514 drop in the first few iterations, as the initial parts are not (yet) constraint
 515 to be generative, *i.e.* a parts should be observed in every positive image of
 516 the specific class. Finally, we observe a small step each time the temperature
 517 parameter is updated.

518 Overall, the convergence is behaving as expected.

519 *Bag-of-word based representation.* The SIFT-Bag-of-parts and Fisher-on-parts
 520 pipelines are then evaluated on the three datasets, see Table 3 and Table 5.
 521 For Willow actions, the performance of the two baseline algorithms (Bag-of-
 522 words and Fisher vectors) are respectively of 50.0% and 58.1%. One can note
 523 that the (SIFT) Bag-of-parts slightly outperforms the standard Bag-of-word.
 524 More interestingly, the proposed Fisher-on-parts representation outperforms
 525 Fisher vectors by more than 3%. Please note that the proposed approach
 526 does not use any extra annotations, contrarily to most of the proposed ap-
 527 proaches (*e.g.* [12] which uses the bounding boxes). This explains why we do
 528 not provide any comparisons with these methods, as they would be mean-
 529 ingless.

530 The Boats dataset also shows improvements on both the (SIFT) Bag-of-

Table 3: Results on Willow and Boats dataset. See text for details.

Method	Willow (mAP)	Boats (mAP)
Bag-of-words [50]	0.500	0.673
Fisher vectors [50]	0.581	0.827
Bag-of-parts	0.510	0.741
Fisher-on-parts	0.614	0.852

531 parts and the Fisher-on-parts. Specifically, we observe more than 6% and
 532 2% mAP increase over BOW and Fisher vectors respectively.

533 Concerning MIT 67, we first observe that our (SIFT) Bag-of-parts en-
 534 coding offers better performances than the Bag-of-word model as well as
 535 the Bag-of-parts proposed in [11]. We also notice that our Fisher-on-parts
 536 improves on the two previous methods. However, we do not obtain better
 537 performance than the Fisher vectors extracted on the full image. We believe
 538 that this result is due to the fact that the MIT 67 requires a lot of con-
 539 text information to recognize scenes, while our Fisher-on-parts encoding acts
 540 as a pooling system that encapsulates most information on the foreground.
 541 Combining Fisher-on-parts with Fisher vectors on the whole image (with
 542 SPM) gives a mAP of 60.0%, which is significantly better than any other
 543 approaches.

Table 4: Results on Willow dataset, using the CNN-based pipeline.

Method	Willow (mAP)
CNN on full image	0.763
Bag-of-parts (CNN)	0.766
CNN-on-parts (CNN)	0.816
Bag-of-parts & CNN-on-parts (CNN)	0.819

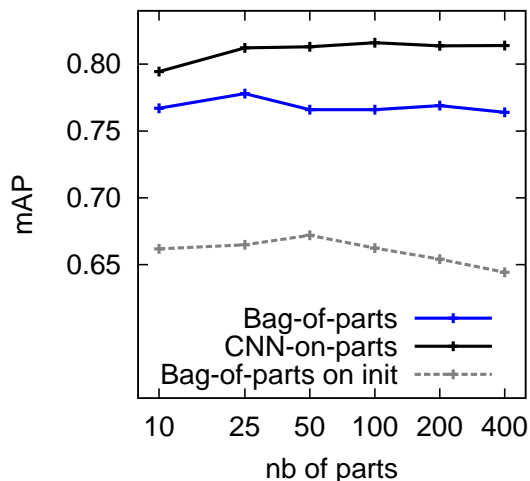


Figure 3: Scores for Willow actions as a function of the number of parts.

544 *CNN-based pipeline.* In these experiments, the regions are described by CNN
 545 features. First, we evaluate Figure 3 the impact of the number of parts used
 546 to describe images, on Willow. It is interesting to note that for (CNN)
 547 Bag-of-parts and CNN-on-parts, performances are stable for any number of
 548 parts between 25 to 400 parts. Furthermore, utilizing only 10 parts offers
 549 reasonable performance. However, if we compute the Bag-of-parts on the
 550 initialization parts (salient regions), *i.e.* without learning m , we note that
 551 having more than 100 parts slightly reduces the performances.

552 We also observed a consistent improvement of the (CNN) bag-of-parts
 553 and CNN-on-parts over the CNN on the full image, as shown Table 4 and
 554 Table 5. For the Willow action dataset the CNN-on-part offers the largest
 555 improvement, while the (CNN) Bag-of-part is the best performing method
 556 on the MIT 67 scenes dataset. This result supports our observation with
 557 the Fisher-on-parts that a foreground pooling effect is very advantageous on
 558 Willow actions, while contextual information is better for MIT scenes.

559 Interestingly, we also observed that combining the two proposed encoding
 560 methods – by doing a simple concatenation of the (CNN) Bag-of-part and
 561 CNN-on-part representations – makes the performance even better, produc-

Table 5: Results on MIT 67 scenes dataset. See text for details.

Method	mAP
bag-of-words [50]	0.345
Fisher vectors [50]	0.550
bag-of-parts of [11]	0.373
our bag-of-parts	0.401
Fisher-on-parts	0.549
Fisher-on-parts based combination	0.600
CNN on full image [26]	0.726
Bag-of-parts (CNN)	0.788
CNN-on-parts (CNN)	0.778
Bag-of-parts & CNN-on-parts (CNN)	0.801

562 ing performance higher than any reported method to our knowledge (80% of
 563 mAP on MIT67).

564 These experiments show that our descriptors, based on distinctive parts
 565 learning, are capable of incorporating mid-level information and produce
 566 richer representations.

567 5. Conclusions

568 In this paper, we propose a new algorithm to recognize images by model-
 569 ing categories as set of distinctive parts that are discovered automatically and
 570 aligned across images, while learning their visual model. The parts that are
 571 discovered are free of any appearance constraint and allow the distinction
 572 between the categories to be recognized. We show how to use the softas-
 573 sign matching algorithm, to simultaneously learn the part models and assign
 574 image regions to model’s parts, starting from an initial set of randomly ex-
 575 tracted image regions. Based on the part model, signatures are computed
 576 to describe images. Finally, the proposed algorithm is validated on three

577 different datasets on which state-of-the-art performances are obtained.

578 **Acknowledgment**

579 This work is partly funded by the RECONSURVE project.

580 **References**

- 581 [1] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with
582 deep convolutional neural networks, in: Advances in neural information
583 processing systems, 2012, pp. 1097–1105.
- 584 [2] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray, Visual cate-
585 gorization with bags of keypoints, in: Intl. Workshop on Stat. Learning
586 in Comp. Vision, 2004.
- 587 [3] J. Krapac, J. Verbeek, F. Jurie, Modeling spatial layout with Fisher
588 vectors for image categorization, in: Proceedings of the International
589 Conference on Computer Vision, 2011.
- 590 [4] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyra-
591 mid matching for recognizing natural scene categories, in: Proceedings
592 of the IEEE Conference on Computer Vision and Pattern Recognition,
593 Vol. 2, 2006, pp. 2169–2178.
- 594 [5] F. Perronnin, J. Sanchez, T. Mensink, Improving the fisher kernel for
595 large-scale image classification, in: European Conference on Computer
596 Vision, 2010.
- 597 [6] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman, Return of the devil
598 in the details: Delving deep into convolutional nets, in: Proceedings of
599 the British Machine Vision Conference, 2014.
- 600 [7] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional
601 networks, in: Proceedings of the European Conference on Computer
602 Vision, Springer, 2014, pp. 818–833.

- 603 [8] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convo-
604 lutional networks for visual recognition, in: Proceedings of the European
605 Conference on Computer Vision, Springer, 2014, pp. 346–361.
- 606 [9] L. Liu, C. Shen, L. Wang, A. van den Hengel, C. Wang, Encoding
607 high dimensional local features by sparse coding based fisher vectors, in:
608 Advances in Neural Information Processing Systems, 2014, pp. 1143–
609 1151.
- 610 [10] S. Maji, G. Shakhnarovich, Part discovery from partial correspondence,
611 in: Proceedings of the IEEE Conference on Computer Vision and Pat-
612 tern Recognition, 2013, pp. 931–938.
- 613 [11] M. Juneja, A. Vedaldi, C. V. Jawahar, A. Zisserman, Blocks that shout:
614 Distinctive parts for scene classification, in: Proceedings of the IEEE
615 Conference on Computer Vision and Pattern Recognition, 2013.
- 616 [12] G. Sharma, F. Jurie, C. Schmid, et al., Expanded parts model for human
617 attribute and action recognition in still images, in: Proceedings of the
618 IEEE Conference on Computer Vision and Pattern Recognition, 2013.
- 619 [13] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring
620 mid-level image representations using convolutional neural networks, in:
621 Proceedings of the IEEE Conference on Computer Vision and Pattern
622 Recognition, 2014.
- 623 [14] J. Krapac, J. Verbeek, F. Jurie, Learning tree-structured descriptor
624 quantizers for image categorization, in: Proceedings of the British Ma-
625 chine Vision Conference, BMVA Press, 2011, pp. 47–1.
- 626 [15] H. E. Tasli, R. Sircé, T. Gevers, A. A. Alatan, Geometry-constrained
627 spatial pyramid adaptation for image classification, in: International
628 Conference on Image Processing, 2014.

- 629 [16] M. Vidal-Naquet, S. Ullman, Object recognition with informative fea-
630 tures and linear classification, in: Proceedings of the International Con-
631 ference on Computer Vision, 2003, pp. 281–288.
- 632 [17] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, D. Ramanan, Ob-
633 ject detection with discriminatively trained part-based models, *IEEE*
634 *Transactions on Pattern Analysis and Machine Intelligence* 32 (9) (2010)
635 1627–1645.
- 636 [18] V. Delaitre, I. Laptev, J. Sivic, Recognizing human actions in still im-
637 ages: a study of bag-of-features and part-based representations., in: Pro-
638 ceedings of the British Machine Vision Conference, Vol. 2, 2010, p. 7.
- 639 [19] A. Quattoni, A. Torralba., Recognizing indoor scenes, in: Proceedings
640 of the IEEE Conference on Computer Vision and Pattern Recognition,
641 2009.
- 642 [20] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman,
643 The PASCAL Visual Object Classes Challenge 2012 Results.
- 644 [21] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet:
645 A large-scale hierarchical image database, in: *Computer Vision and*
646 *Pattern Recognition, IEEE*, 2009, pp. 248–255.
- 647 [22] N. Pinto, Y. Barhomi, D. Cox, J. DiCarlo, Comparing state-of-the-art
648 visual features on invariant object recognition tasks, in: *Applications of*
649 *Computer Vision (WACV)*, 2011 IEEE Workshop on, 2011.
- 650 [23] S. Zheng, Z. Tu, A. Yuille, Detecting object boundaries using low-, mid-,
651 and high-level information, in: *Computer Vision and Pattern Recogni-*
652 *tion*, 2007.
- 653 [24] Y. Boureau, N. Le Roux, F. Bach, J. Ponce, Y. LeCun, Ask the locals:
654 multi-way local pooling for image recognition, in: *Proceedings of the*
655 *International Conference on Computer Vision*, 2011.

- 656 [25] M. Oquab, L. Bottou, I. Laptev, J. Sivic, et al., Learning and trans-
657 ferring mid-level image representations using convolutional neural net-
658 works, *Computer Vision and Pattern Recognition*.
- 659 [26] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, A. Oliva, Learning Deep
660 Features for Scene Recognition using Places Database., in: *Advances in*
661 *Neural Information Processing Systems*, 2014.
- 662 [27] Y. Gong, L. Wang, R. Guo, S. Lazebnik, Multi-scale orderless pooling of
663 deep convolutional activation features, in: *Proceedings of the European*
664 *Conference on Computer Vision*, 2014.
- 665 [28] D. Parikh, Recognizing jumbled images: the role of local and global
666 information in image classification, in: *Proceedings of the International*
667 *Conference on Computer Vision*, IEEE, 2011, pp. 519–526.
- 668 [29] J. Tighe, S. Lazebnik, Superparsing: scalable nonparametric image pars-
669 ing with superpixels, in: *Proceedings of the European Conference on*
670 *Computer Vision*, Springer, 2010.
- 671 [30] R. Sire, T. E. Tasli, T. Gevers, Superpixel based angular differences as
672 a mid-level image descriptor, in: *International Conference on Pattern*
673 *Recognition*, 2013.
- 674 [31] B. Fernando, E. Fromont, T. Tuytelaars, Effective use of frequent item-
675 set mining for image classification, in: *Proceedings of the European*
676 *Conference on Computer Vision*, Springer, 2012, pp. 214–227.
- 677 [32] J. Carreira, R. Caseiro, J. Batista, C. Sminchisescu, Semantic segmen-
678 tation with second-order pooling, in: *Proceedings of the European Con-*
679 *ference on Computer Vision*, Springer, 2012, pp. 430–443.
- 680 [33] Z. Niu, G. Hua, X. Gao, Q. Tian, Context aware topic model for scene
681 recognition, in: *Proceedings of the IEEE Conference on Computer Vi-*
682 *sion and Pattern Recognition*, 2012, pp. 2743–2750.

- 683 [34] Y. Su, F. Jurie, Improving image classification using semantic attributes,
684 International journal of computer vision 100 (1) (2012) 59–77.
- 685 [35] Z. Liao, A. Farhadi, Y. Wang, I. Endres, D. Forsyth, Building a dic-
686 tionary of image fragments, in: Proceedings of the IEEE Conference on
687 Computer Vision and Pattern Recognition, 2012, pp. 3442–3449.
- 688 [36] J. J. Lim, C. L. Zitnick, P. Dollár, Sketch tokens: A learned mid-level
689 representation for contour and object detection, Proceedings of the IEEE
690 Conference on Computer Vision and Pattern Recognition, 2013.
- 691 [37] I. Endres, K. J. Shih, J. Jia, D. Hoiem, Learning collections of part
692 models for object recognition, Proceedings of the IEEE Conference on
693 Computer Vision and Pattern Recognition, 2013.
- 694 [38] J. Chua, I. Givoni, R. Adams, B. Frey, Learning structural element
695 patch models with hierarchical palettes, in: Proceedings of the IEEE
696 Conference on Computer Vision and Pattern Recognition, 2012, pp.
697 2416–2423.
- 698 [39] S. Singh, A. Gupta, A. A. Efros, Unsupervised discovery of mid-level
699 discriminative patches, in: Proceedings of the European Conference on
700 Computer Vision, Springer, 2012, pp. 73–86.
- 701 [40] N. Zhang, J. Donahue, R. Girshick, T. Darrell, Part-based r-cnns for
702 fine-grained category detection, in: Proceedings of the European Con-
703 ference on Computer Vision, 2014, pp. 834–849.
- 704 [41] S. N. Parizi, A. Vedaldi, A. Zisserman, P. Felzenszwalb, Automatic dis-
705 covery and optimization of parts for image classification, in: Interna-
706 tional Conference on Learning Representations, 2015.
- 707 [42] R. Girshick, F. Iandola, T. Darrell, J. Malik, Deformable part models are
708 convolutional neural networks, in: Proceedings of the IEEE Conference
709 on Computer Vision and Pattern Recognition, 2015.

- 710 [43] L. Wan, D. Eigen, R. Fergus, End-to-end integration of a convolutional
711 network, deformable parts model and non-maximum suppression, in:
712 Proceedings of the IEEE Conference on Computer Vision and Pattern
713 Recognition, 2015.
- 714 [44] W. Ouyang, X. Wang, X. Zeng, S. Qiu, P. Luo, Y. Tian, H. Li, S. Yang,
715 Z. Wang, C.-C. . C. Loy, Deepid-net: Deformable deep convolutional
716 neural networks for object detection, in: Proceedings of the IEEE Con-
717 ference on Computer Vision and Pattern Recognition, 2015, pp. 2403–
718 2412.
- 719 [45] R. Sircé, F. Jurie, Discovering and aligning discriminative mid-level fea-
720 tures for image classification, in: International Conference on Pattern
721 Recognition, IEEE, 2014, pp. 1975–1980.
- 722 [46] J. M. B. Hariharan, D. Ramanan, Discriminative decorrelation for clus-
723 tering and classification, in: Proceedings of the European Conference on
724 Computer Vision, 2012.
- 725 [47] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, E. Mjolsness, New algo-
726 rithms for 2d and 3d point matching:: pose estimation and correspon-
727 dence, *Pattern Recognition* 31 (8) (1998) 1019–1031.
- 728 [48] D. Geiger, F. Girosi, Parallel and deterministic algorithms from mrfs:
729 Surface reconstruction, *IEEE Transactions on Pattern Analysis and*
730 *Machine Intelligence* 13 (5).
- 731 [49] V. Chvatal, *Linear programming*. 1983, WH Freeman and Company,
732 New York.
- 733 [50] K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman, The devil is in
734 the details: an evaluation of recent feature encoding methods, in: Pro-
735 ceedings of the British Machine Vision Conference, 2011.
- 736 [51] A. Vevaldi, B. Fulkerson, Vlfeat an open and portable library of com-
737 puter vision algorithms, in: *ACM Multimedia*, 2010.

738 [52] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick,
739 S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast
740 feature embedding, in: ACM International Conference on Multimedia,
741 2014.



Figure 4: Visualization of parts locations for "riding cycle", "playing instruments", and "riding horse".



Figure 5: This figure shows the highest scoring regions for a set of parts learned for the *riding horse* action. Each row corresponds to a part.