DOI: 10.14529/ctcr170302

METHODS AND PRINCIPLES OF USING A PRIORI KNOWLEDGE IN RECOGNITION TASKS

***V.A. Parasich**, pva16@yandex.ru,*
***A.V. Parasich**, parasich_av@yandex.ru,*
***I.V. Parasich**, parasichiv@mail.ru*

South Ural State University, Chelyabinsk, Russian Federation

The using of a priori knowledge is an important part of the development of pattern recognition systems. Often the proper use of a priori knowledge allows bring quality of recognition algorithm to the level of practical usage. The main advantage of using a priori knowledge is that the classification algorithms are prone to errors, whereas a priori statements are always true. In the article will be show how to improve the quality of recognition system using a priori knowledge. The evolution of approaches to the use of knowledge considered by the example of the task of object detection, the advantages and disadvantages of these approaches analyzed. The basic principles of using a priori knowledge in recognition algorithms formulated.

Keywords: object recognition, machine learning, object detection, convolution neural networks, Deformable Parts Models, Implicit Shape Model, knowledge representation.

One of the main problems with which the developers of pattern recognition systems are faced is the following: users need a system with a minimum risk of error, ideally tending to zero, otherwise, the system will be useless. However, computer vision and machine learning algorithms are usually unreliable and error prone. It is very difficult to get quality close to 100 % using classic techniques of machine learning and pattern recognition. An example of problems that are difficult to solve solely by means of machine learning but can be solved through the use of a priori knowledge is *double-counting phenomena* – the problem of entanglement symmetrical body parts (left and right arm or left and right leg) in the human pose estimation task. Symmetrical parts of the body have a very similar appearance, so they are difficult to distinguish from each other solely by the classifier.

The way to solving this problems is the use of a priori knowledge about recognizable objects. In most recognition tasks there are some constraints on possible configurations of recognizable objects that are always executed. In the human pose estimation task we know that person has only one head, two hands, two legs, arms grow from the shoulders, the length of the limbs does not change during the recognition time. In handwriting recognition task, we have the vocabulary of possible words and knowledge of the syntactic structure of a sentence.

The main advantage of using a priori knowledge – the algorithms of recognition are prone to errors, whereas a priori statements are always true (have 100 % reliability to which we aspire). In addition, the adjustment of machine learning algorithms for a particular task is a very nontrivial process with unobvious regularities that does not guarantee internal consistency of the result at the output of the system. At the same time, a priori statements are usually simple and understandable, algorithms based on them are easily configurable, and it is possible within certain limits to ensure the correctness of the result.

In fact, in any real recognition system, a priori knowledge about recognizable objects is used in some form. The using of such knowledge can significantly raise the quality of the recognition systems. Often the proper use of a priori knowledge allows bringing the quality of recognition algorithm to the level of practical application. Despite this correct use of a priori knowledge is an open question in the modern theory of pattern recognition.

For the object detection task, several basic approaches to the use of knowledge were developed: *Deformable Part Models, Mixture-of-Parts, Implicit Shape Model, Stacking*. Each of these approaches has its own advantages, disadvantages, scope of applicability and ways to improve. In most tasks of pattern recognition, there are similar methods of using a priori information can be applied.

1. Global object model (*Deformable Part Models*)

Let us divide the object of interest into its component parts (in the task of detecting a bicycle, such parts can be wheels, a rudder, a saddle and pedals). Separately we will train different detector for each of the components. In this case, the model of the object is determined by the set of permissible mutual positions of the component parts. Such model allows small changes in the mutual positions of parts in a certain range (deformation). Let us say that the components of the sought object were detected. An object is considered as detected if the recognized positions of its components form a correct configuration that satisfies all constrains of the model.

The gain from using this approach arises because the detectors of the component parts are easier to train to a high quality of recognition, since the appearance variability of the constituent part of the object is less than the appearance variability of the entire object. Therefore, a much smaller amount of training data and a simpler detector can be used. Let's say an object consists of *5* component parts, each of which has two variants of appearance and can be combined with the following part in *2* ways. In such a simple example, the task of learning to recognize an entire object can be considered as $2^8 = 256$ times more complex than learning to recognize a particular part. In real tasks, the difference will be even more significant.

The advantage of the method is a certain guarantee of the correctness of the result. The object with wrong configuration of parts (for example, a bicycle without wheels) will never be issued as a successful recognition.

In the literature, this class of models is called *Deformable Part Models*. An example of the application of this approach can be found in [1], where it is called *Star model*. In this work, models for recognition of the component parts of the objects are trained with using of HOG-descriptors as features.

However, this approach has a number of fundamental drawbacks:

Recognizing objects from different view angles. If the view angle is changed, the appearance of the object of interest and mutual arrangement of its parts can changes fundamentally. For example, the image of a bicycle or a car on side view can be very different from the image of the same bicycle or car in front view. This does not allow the application of a single model for object recognition from different viewpoints. Because of this, we have to build a separate model for each variant of the viewpoint (an example of several models for a bicycle shown in Fig. 1). But even with this improvement, at once there are problems with intermediate viewpoints arises.

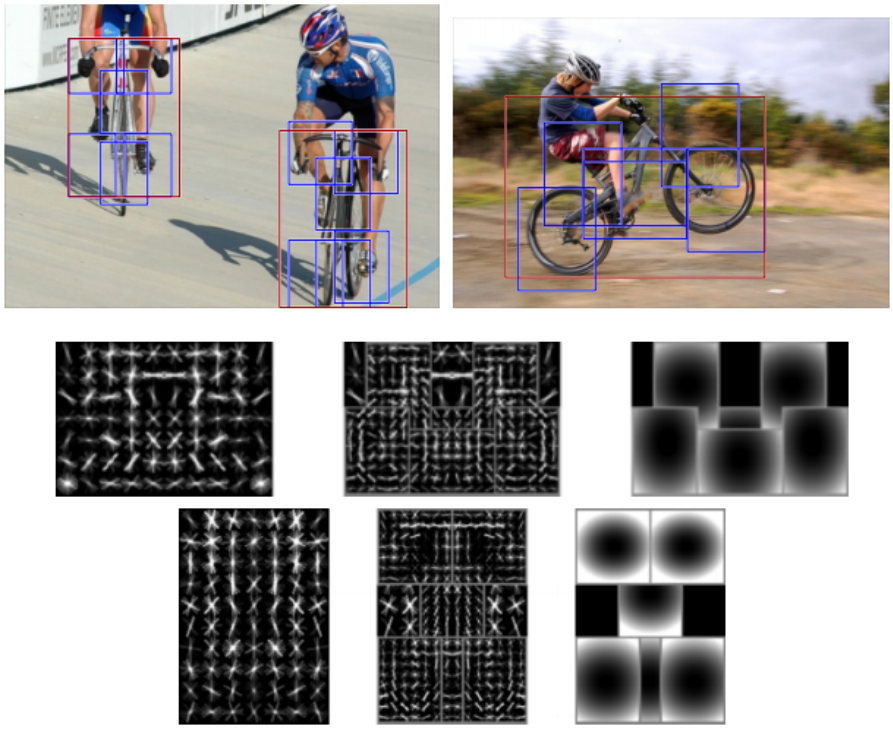

Fig. 1. Example of a model for recognizing a bicycle from different view angles [1]

The need for manual markup of training data. In addition to the previous drawback, in order to train the detectors of the elementary parts of the object it is necessary to manually select its component parts in the image, since usually a detailed description of the images is not available. In order to achieve good recognition quality, it is often necessary to train an algorithm on a large number of images (up to several million in the case of the use of convolution neural networks), so this drawback can become critical.

Unrobustness to the errors of one of the detectors. If one of the parts of the object is mistakenly did not recognized, this can lead to a false negative result of the detection of the entire object. Moreover, a small error in the localization of one of the parts can lead to the non-satisfiement of local constraints and as a consequence to the refusal to detect the entire object.

Intraclass variety of objects. Not always objects correspond exactly to the model constraints. Sometimes the position of one of the parts of the object relative to other parts may change unpredictably. This will be critical for the methods of this group, but it is uncritical for methods without strict constraints. Some classes of objects (for example, chairs) have a very large variability of forms, which cannot be described with the help of such models. For such cases, models that are more flexible are required.

The need for manual preparation of models. For each object of interest, it is necessary to build its model (or several models) manually. It is quite acceptable if you want to detect single object (as in the pedestrian search task) or a small number of objects (recognition of digits). If you need to find hundreds of different kinds of objects on images (as in some competitive tasks of classifying images), the application of this approach is difficult.

Instability to overlapping. Some of the parts of the object of interest may not be visible on the image because of overlaps. The object can be recognized by visible parts, using the classical methods of machine learning. Moreover, the quality of recognition using the model may turn out worse than the quality of recognition without using models. To eliminate this drawback, the object model must be supplemented with explicit overlap modeling. In the end, it is more advantageous to explicitly model overlaps than to add images with overlapping in the training set. In the case of explicit modeling, the system will focus only on continuous overlapping of adjacent parts of the object, and for the algorithm without a priori knowledge, both continuous overlaps and disconnected overlaps of individual parts of the object are equally possible, whereas disconnected overlaps are not possible in reality.

2. Modeling of local constraints (Mixture-of-Parts)

Instead of specifying the complete structure of an object, you can use constraints on pair-wise disposition of its neighboring parts. Just as in the algorithms of the class *Deformable Part Models* [1], in the methods of the class *Mixture-of-Parts*, the object of interest is divided into component parts. Knowledge of the object's parts relative location is used in the form of explicit constraints or in the form of binary potentials in algorithms of energy optimization, dynamic programming or message passing. In case of the human pose estimation task, this algorithm uses several classes to represent one part of the human body. Each separate class corresponds to a certain configuration of its «own» part of the body and its neighboring parts (for example, the palm class for the up-raised arm or elbow class of a straight horizontal hand). This allows you to check the consistency of the recognition of nearby pixels of a body part and to carry out optimization of posture based on this information, as well as to recognize the invisible parts of the body.

As a result, the problem of *Deformable Part Models* with recognition of the object from different view angles is solved – there is no need to create separate models for the frontal view and the side view.

We will outline the main shortcomings and limitations of algorithms based on *Deformable Part Models* and *Mixture-of-Parts* approaches.

Computational complexity in the recognition phase. It is required to run the detector of each object part for each pixel of the image, but the overwhelming number of detector's runs will give a negative result. Therefore, most calculations are useless.

A collection of hard negative learning examples is required. In any task of learning the objects detector, the *hard negative samples mining* [2] problem arise – the need for special algorithms to search for or create learning examples that are similar in appearance to the object of interest, but which are not the object of interest. Under natural conditions, such examples encountered much less often than examples that are certainly not an object of interest, which are easy to classify.

Different orientations of parts of the object. In the case of object detection (for example, a person's posture), the orientations of object's parts can arbitrarily vary, because of which it will be necessary to put into the detector all possible orientations of the part. This will worsen the overall quality of the classification and increase the number of training sample required. Therefore, in the task of recognizing the person's posture, not the detectors of the entire body part (the entire bone of the arm or leg) uses, but the joints detectors (for hands, elbows, knees).

To correct these shortcomings, the algorithm *Implicit Shape Model* was developed [3].

3. Implicit Shape Model

With the help of clustering, a dictionary of the most frequently found fragments of the image of the object of interest is constructed. For each fragment from the dictionary, the average offset (or several most common offsets) from the center of the fragment to the center of the object of interest is calculated and stored in the dictionary together with this fragment's descriptor. At the recognition stage, each of the fragments of the image compared with fragments from the dictionary by comparing their descriptors (often only those fragments of the image on which special points found). If match is found, the offsets stored together with this fragment are accumulated (they «vote» for the specific position of the object in the image). The set of offsets obtained in this way is processed with a help of any of the algorithms for *Non-Maximum Suppression* (usually using *MeanShift* [4]). If the sum of the vote's weights in the accumulator (or value of some function of the set of votes in the accumulator) is greater than a certain threshold, the object considered as detected. Despite the fact that the model does don't have any explicit knowledge about the structure of recognized object, the availability of a coordinated voting of fragments for some location of this object in the image allows to expect the presence of the object on this image.

The huge advantage of this approach is the absence of the need for manual model design, manual setting of constraints and manual marking of training data (you only need to know the center of the object in each image). It is not required to explicitly break the object of interest into component parts, for us this will be done by the clustering algorithm (although no one forbids doing this manually). Another very important advantage is the more efficient use of computational resources in comparison with the previously considered methods, because of every image pixels would be classified once.

The problems of voting and vote's accumulation and Non-Maximum Suppression. Each of the codewords votes for the position of the object has several parameters: offset to the center of the object, the probability to find the center of the object using the specified offset, variance of the position of the object's center relative to offset**.** There is no single answer to the question of how to accumulate the offsets of different fragments. For one fragment, there may be several offset classes to the center of the object (as in the case of the wheel of the car in Fig. 2). At the learning stage, the question arises which of these offsets should be retained for the recognition phase, and which ones should be discarded. You may need a fine adjustment of the learning and vote accumulating parameters (kernel size in the case of using *MeanShift* [4]), that outweigh the benefits of not having to manually construct the object model in some cases.

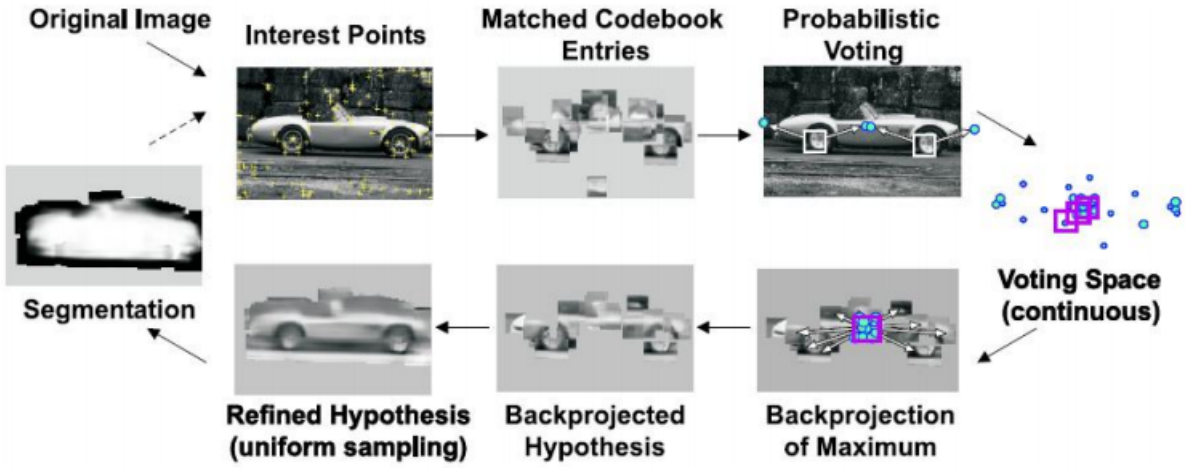

Fig. 2. Example of Implicit Shape Model for car detection [1]

The global correctness of the result is not guaranteed. The absence of any explicit constraints in the models of this class generates the absence of any guarantees of the result correctness at the output. If the values of the weights are not successful, the car without the wheels, and the car with the wheels on top can be recognized as correct car. This is not always acceptable from the point of view of the user of the system.

Let us say we trained a car detector, and the input of the system received an image from the car repair shop, in which closely in several rows there are wheels, and no car are in this image. The sum of votes from the wheels will be enough to find the car on the image, despite the fact that there are no other parts of the car on this image. The problem can be solved by introducing additional tags into the votes. Each of the parts of the object will have its own tag; while the object will be considered as detected only if all the tags are among the votes in some neighborhood.

Modeling of multi-level hierarchies. This method assumes a one-level hierarchy of the object of interest. The efficiency of the approach for objects with a multi-level hierarchy and high variability in the interconnection of the component parts seems to be highly questionable.

Problems of forming a dictionary. It is difficult to construct a qualitative metric for comparing fragments of images (patches). The appearance of the same part of the object can vary very much in different images. Besides, the procedure for finding the most similar fragment in a large patch dictionary is computationally very laborious, because of what it is necessary to reduce the volume of the dictionary; as a result, the accuracy may decrease. Therefore, instead of clustering patches, the training of a tree is often used. A leaf of a tree considered as a separate «cluster». When learning decision trees, there is an automatic selection of the most informative features, that eliminates the need to manually select the patch comparison metric, and improves the quality of matching similar parts of the image. The use of decision trees also allows you to significantly increase the performance both in comparison with classic *Deformable Part Models* [1] and in comparison with the *Implicit Shape Model* based on the dictionary.

Dependence of the votes weights on balancing of the collection. To determine the weights of the votes at the learning stage, you can use completely different algorithms. In addition, a balance must be made between the correctness of the detection (minimization of false positive detections rate) and the accuracy of the localization of the object. The selection of a good formula for calculating weights is also difficult as choosing a good metric of the similarity of patches. In most cases, the main criterion in choosing a weight is the ratio of the number of correct detections to the number of false positives within a given patch. However, such a ratio is often more dependent on the balancing of samples amount of different types in the training set (the ratio of the number of examples of different types) than on the real properties of the patch. So often votes weights have to be adjusted by balancing the training sample or manually.

The advantages of Mixture-of-Parts in terms of testing the consistency of votes are not used. The method does not use any information about the consistency of votes from neighboring pixels; obviously wrong votes are not rejected, since the algorithm scheme does not contain the ideas of *Mixture-of-Parts*. Somehimes it is possible to determine wittingly incorrect votes with using additional information, containing in offsets. The combination of the advantages of the methods Mixture-of-Parts and Implicit Shape Model is of great scientific and practical interest.

4. The usage of Deformable Parts Model in fine-grained image categorization tasks

Deformable Parts Model are actively used in *fine-grained image categorization* tasks – assignment of an object to one of a large number of visually similar classes of objects, when some classes differ only in small details (for example, the classification of birds or cars). Recently, convolution neural networks have been used as classifiers of separate parts of the object instead of *HOG-SVM* classifiers. In tasks of this type, the use of *Deformable Parts Model* makes it possible to significantly improve the quality of classification. In particular, convolution neural networks are very sensitive to superfluous details in the image. The use of *Deformable Parts Model* instead of a single classification of the entire image with a help of single neural network makes it possible to narrow *receptive field* of the neural network, thereby excluding the influence of extraneous image details on the result and increasing the influence of small details important for the recognition of *fine-grained* classes (for example, two models of the car can visually differ only in appearance of the radiator, and two breeds of birds can differ only in the color of the top of the head). With *Deformable Parts Model* we can firstly localize specific part of an object, and then classify this part independently.

Another similar application field of *Deformable Parts Model* is *fine-grained action recognition* task: the detection of an action on the image or video, using only a small area of the image (the fact of conversation on the mobile phone can be recognized by a small area of the image containing the hand with the phone brought to the ear). The first neural network searches for image regions that are likely to contain some action; the second neural network classifies the regions found by the first one. In this case, the variety of data on which the second neural network is trained and dealt with is substantially smaller than the full variety of all possible data, which allows neural network to concentrate its attention on important small details of the images. The approach described above successfully worked in the competition held on the *Kaggle* platform [5]. In particular, the winning solution used a similar method.

5. Knowledge incorporation into learning process

An interesting area is the modeling of a priori knowledge in the learning process of an object detector. In work [6] there is an example of using this approach in the pedestrian detection task (*Informed Haar*). Recognition of pedestrians is carried out using decision forest, and special kind of integral images used as features for forest training. The sum pixel's intensities within a certain rectangle are calculated. In this case, pixels divided into three types: the intensities of pixels of the first type taken with plus sign; the intensities of pixels of the second type – with minus sign; the intensities of pixels of the third type are ignored. For the generation of features, a human model is used (Background-torso-arms-legs-head). A certain rectangle selected on the model, the pixels corresponding to different parts of the model are assigned different types in the resulting feature. For example, a rectangle could be selected in the shoulder area, pixels falling into the head zone are ignored, pixels corresponding to the trunk pattern will be taken with a positive sign, and sum of their intensities will be compared with sum of pixel's intensities corresponding to the assumed background. Examples of these features shown on Fig. 3. Such features will well detect the upper part of the human body.

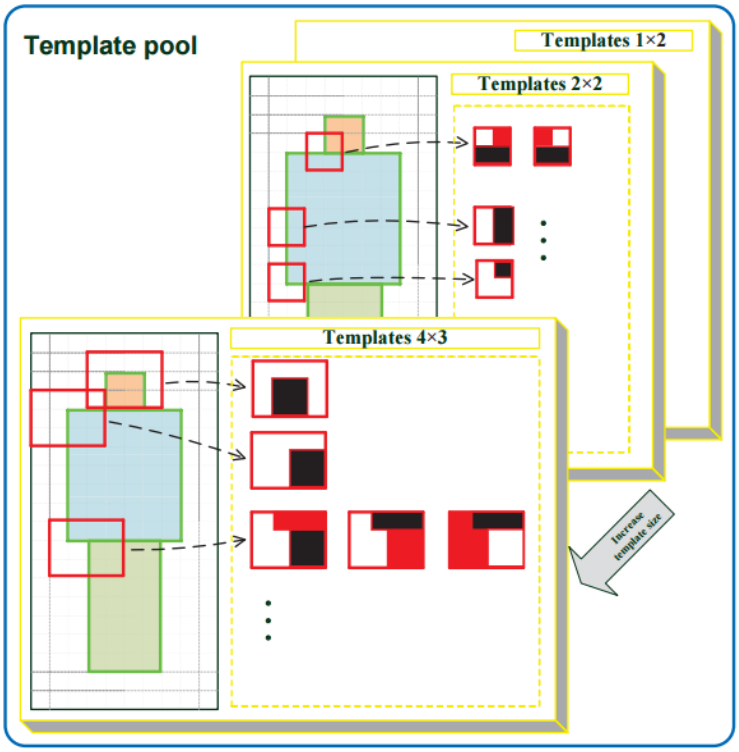

Fig. 3. Feature generation in *Informed Haar* method

6. Stacking

Stacking is the use of predictions from some classifier as features for the other classifiers. In the case of image pixels classification, it uses the results of classification of the pixel's neighborhood. Currently, various stacking variations are used in most human pose estimation works (for example [7]), showing

the best results on publicly available datasets. The principle of stacking is based on the following mechanism. Some parts of the human body (trunk, head, and neck) are much easier to recognize than others (arms and legs). The results of classification of difficult body parts after the first stage (as a result of using a conventional classifier) may be completely erroneous. However, with information about the location of other body parts, these errors can be corrected (for example, using information that the elbow is some distance from the shoulder as shown in Fig. 4). In this case, the system learns to use this information automatically, without using specialized *human-designed* procedures in the recognition algorithm code. Now the stacking of convolution neural networks is mainly used.

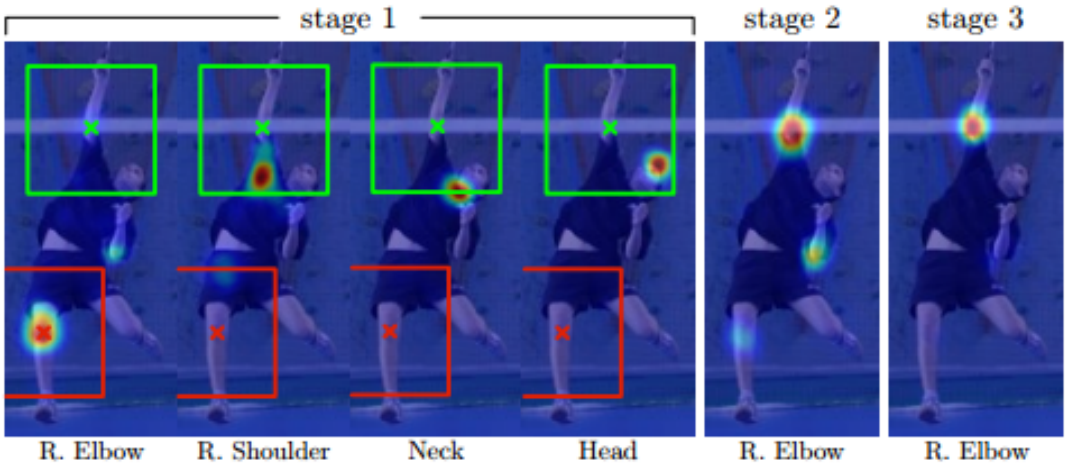

Fig. 4. Example of improving the right elbow recognition results during the three stages of the stacking

7. General principles of using knowledge in recognition tasks

Most methods of using knowledge in recognition tasks can be reduced to the use of several basic principles.

The decomposition of the object of interest into component parts is uses, which makes it possible to significantly reduce the variety of data for training and, at the same time, introduce additional constraints on the relative positioning of the parts.

A composite object can take a very limited number of configurations from the set of potentially possible. Knowing the position of some parts of the composite object, one can conclude where other parts are located, significantly narrowing the search space and cutting off deliberately incorrect variants. You can achieve a significant increase in the quality of the algorithm due to the use of various constraints arising from the structure of the object of interest. One of the ways is the transfer of knowledge obtained at a high level to a lower level, for example, the use of high-level information as features of the classifier.

The position and configuration of simple homogeneous objects with a small number of parameters can be determined more reliably or by incomplete analysis. One can use heuristic algorithms based on some invariants of the object of interest, train a good classifier or combine the answers of several algorithms for recognizing this object. One of the possible sources of information – multi-frame tracking: with very high probability, the object will either move slightly or stay in the same place in the next frame.

Conclusions

In this article, we show how the using of a priori knowledge can help to improve quality of recognition system. Using of a prior knowledge can give such advantages as guaranty of correctness of final recognition result, reduce of training complexity, making models more easy to adjust for concrete task, help to focus algorithm on parts of the object, that are most important for recognition. One of the biggest problems with the use of knowledge is the following: algorithmically generated high-level knowledge can be erroneous, and the more we rely on this information, the more likely it is that if our knowledge is erroneous, the whole algorithm will fail. As a practical recommendation, it can be advisable to embedding the use of knowledge in the lowest possible level of the algorithm (in the form of voices for the po-

sition of the object in *Implicit Shape Model* or as features of the classifier as example). In this case, even with erroneous knowledge, the algorithm has the opportunity to produce correct results based on correct low-level data. Another possible approach is to check the consistency of knowledge with the input data and to disqualify incorrect knowledge when there are discrepancies.

References

1. Felzenszwalb P.F., Girshick R.B., McAllester D., Ramanan D. Object Detection with Discriminatively Trained Part Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010, vol. 32, no. 9, pp. 1627–1645. DOI: 10.1109/TPAMI.2009.167

2. Canavet O., Fleuret F. Efficient Sample Mining for Object Detection. *Proceedings of the Asian Conference on Machine Learning (ACML)*, 2014, pp. 48–63.

3. Leibe B., Leonardis A., Schiele B. An Implicit Shape Model for Combined Object Categorization and Segmentation. *Springer Berlin Heidelberg*, 2006, pp. 508–524. DOI: 10.1007/11957959_26

4. Comaniciu D., Meer P. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, vol. 24. no. 5, pp. 603–619. DOI: 10.1109/34.1000236

5. *State Farm Distracted Driver Detection*. Available at: https://www.kaggle.com/c/state-farm-distracted-driver-detection (accessed March 2017).

6. Zhang S., Bauckhage C., Cremers A.B. Informed Haar-Like Features Improve Pedestrian Detection. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 947–954. DOI: 10.1109/cvpr.2014.126

7. Wei S.E., Ramakrishna V., Kanade T., Sheikh Y. Convolutional Pose Machines. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4724–4732. DOI: 10.1109/cvpr.2016.511

Received 25 March 2017

УДК 004.855.5 **DOI: 10.14529/ctcr170302**

МЕТОДЫ И ПРИНЦИПЫ ИСПОЛЬЗОВАНИЯ АПРИОРНЫХ ЗНАНИЙ В ЗАДАЧАХ РАСПОЗНАВАНИЯ

В.А. Парасич, А.В. Парасич, И.В. Парасич

Южно-Уральский государственный университет, г. Челябинск

Использование априорных знаний является важной частью разработки систем распознавания образов. Зачастую именно правильное использование априорных знаний позволяет довести качество алгоритма распознавания до уровня практической применимости. Главное преимущество использования априорных знаний состоит в том, что алгоритмы классификации неизбежно подвержены ошибкам, в то время как априорные утверждения всегда верны. В статье продемонстрированы пути улучшения качества системы распознавания с помощью использования априорных знаний. Рассматривается процесс эволюции подходов к использованию знаний в системах технического зрения на примере задачи поиска объекта на изображении, проводится анализ преимуществ и недостатков данных методов. Сформулированы базовые принципы, на которых основано большинство способов использования знаний в алгоритмах распознавания.

Ключевые слова: распознавание образов, машинное обучение, сверточные нейронные сети, Deformable Parts Models, Implicit Shape Model, представление знаний.

Парасич Виктор Александрович, канд. техн. наук, доцент кафедры электронных вычислительных машин, Южно-Уральский государственный университет, г. Челябинск; pva16@yandex.ru.

Парасич Андрей Викторович, аспирант кафедры электронных вычислительных машин, Южно-Уральский государственный университет, г. Челябинск; parasich_av@yandex.ru.

Парасич Ирина Васильевна, канд. техн. наук, доцент кафедры математического и компьютерного моделирования, Южно-Уральский государственный университет, г. Челябинск; parasichiv@ mail.ru.

Поступила в редакцию 25 марта 2017 г.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Parasich, V.A. Methods and Principles of Using a Priori Knowledge in Recognition Tasks / V.A. Parasich, A.V. Parasich, I.V. Parasich // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». – 2017. – Т. 17, № 3. – С. 15–23. DOI: 10.14529/ctcr170302

FOR CITATION

Parasich V.A., Parasich A.V., Parasich I.V. Methods and Principles of Using a Priori Knowledge in Recognition Tasks. *Bulletin of the South Ural State University. Ser. Computer Technologies, Automatic Control, Radio Electronics*, 2017, vol. 17, no. 3, pp. 15–23. DOI: 10.14529/ctcr170302