Computational Baby Learning

Xiaodan Liang[†] [⋆] Si Liu[†] Yunchao Wei[†] [*] Luoqi Liu[†] Liang Lin[⋆] Shuicheng Yan [†]
[†] National University of Singapore [⋆] Sun Yat-sen University [*] Beijing Jiaotong university
{xdliang328,fifthzombiesi,wychao1987,llq667}@gmail.com
linliang@ieee.org eleyans@nus.edu.sg

Abstract

A baby inherently possesses the capability of recognizing a new visual concept (e.g., chair, dog) by learning from only very few positive instances taught by parent(s) or others, and this recognition capability can then be gradually further improved by exploring and/or interacting with the real instances in the physical world. In this work, we aim to build a computational model to interpret and mimic this baby learning process, based on prior knowledge modelling, exemplar learning, and learning with video contexts. The prior knowledge of a baby, inherited through genes or accumulated from life experience, is modelled with a pre-trained Convolutional Neural Network (CNN), and the convolution layers form the knowledge base of the baby brain. When very few instances of a new concept are taught, an initial concept detector is built by exemplar learning over the deep features from the pre-trained CNN. Furthermore, when the baby explores the physical world, once a positive instance is detected/identified with high score, the baby shall further observe/track the variable instance possibly from different view-angles and/or different distances, and thus more instances are accumulated. We mimic this process by the massive online unlabeled videos and well-designed tracking solution. Then the concept detector can be fine-tuned based on these new instances. This process can be repeated again and again till the baby has a very mature concept detector in the brain. Extensive experiments on Pascal VOC-07/10/12 object detection datasets [8] well demonstrate the effectiveness of the proposed computational baby learning framework. It can beat the state-of-the-art full-training based performances by learning from only two positive instances for each object category, along with 20,000 videos which mimic the baby exploration of the physical world.

1. Introduction

From the birth of a human being, he/she never stops learning in all his/her life. For a human baby, learning to recognize a new concept (e.g. chair or dog) seems to be a very natural process. First, the baby brain has the basic cognition about real instances in the physical world (e.g. what an object is), which is endowed by the inherited genes and accumulated life experience. Second, after the parent(s) or others teach the baby a few positive instances about a new concept, the initial recognition capability about the concept can be built. Third, during continuously exploring and/or interacting with the diverse instances and scenes in real life, the baby can associate the initial simple instances with other variants by using various information linkages. Finally, based on the accumulated knowledge about the concept, the baby can gradually improve the recognition capability and recognize diverse instances he/she never saw.

Recent successes in computer vision [23] [14] [27] [4], however, largely rely on the big number of labeled instances of visual concepts, which may require considerable human efforts. The construction of an appearance-based object detector is costly and difficult because the number of training examples must be large enough to capture different variations in the object appearance. Some researchers have made efforts on improving the initial models by using very few labeled data, along with the detection/search results from web images [2] [6] [3] or weakly annotated videos [21] [1]. However, a baby does not learn from scratch and he/she often utilizes prior knowledge in the brain for learning. The prior knowledge modeling (like the baby brain) has been largely unexplored before. Also, the fact that a baby learns by exploring the physical world has not been modeled for designing solutions for visual recognition. In this paper, we make the first attempt and build a computational model to interpret and mimic this baby learning process. As illustrated in Figure 1, we propose a robust learning framework which can effectively model the prior knowledge, build the initial model by exemplar learning with very few positive instances for a new concept, and gradually improve the capability by exploring more diverse instances in real-world videos.

First, we model the prior knowledge (for a baby, it comes from genes and life experience) with a pre-trained Convolutional Neural network (CNN). The learned convolution layers can mimic the layered brain neurons in the baby brain

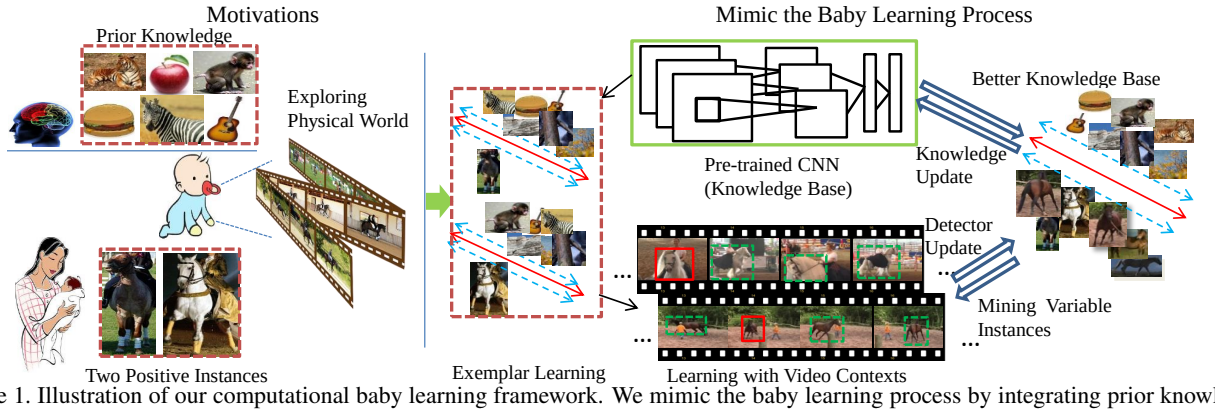

Figure 1. Illustration of our computational baby learning framework. We mimic the baby learning process by integrating prior knowledge modelling, exemplar learning and learning with video contexts. The prior knowledge of a baby from genes and life experience is modelled with a pre-trained CNN. When very few positive instances of a new concept (e.g., horse) are taught, an initial concept detector can be built by exemplar learning based on the learned knowledge base from CNN. Once a positive instance in a frame is detected with the highest score (red solid box), more variable instances (green dashed box) can be tracked from video contexts. The concept detector can be gradually improved with these new instances and we repeat this process again and again. In addition, the pre-trained CNN will be gradually fine-tuned if enough instances are collected, which leads to more informative features for training detectors in turn.

to provide knowledge base for recognition. Specifically, the genes are simulated with a general pre-trained CNN and the life experience can be regarded as the previously learned visual concepts in the baby brain. Accumulating life experience of the baby can be posed as adapting the general pre-trained CNN (i.e. genes) to the domain-specific network using the instances of these learned visual concepts.

Second, when very few positive instances of a new concept are given, the initial concept detector is built by exemplar learning [17], which trains a separate linear classifier for every exemplar in the training set based on the deep features from intermediate layers of the pre-trained CNN. Other learned visual concepts are used as negative instances to enhance the discriminative capability.

Third, when the baby explores the physical world, once a positive instance is identified with high confidence, the baby shall further track more variable instances from different view-angles and/or distances. In our proposed framework, the object instances can be accumulated in a similar way as the baby. More specifically, the exploration of the physical world can be naturally simulated with the unlabeled video clips from the online video sharing websites (e.g., YouTube.com). The positive instance with highest confidence in each clip is selected as the seed instance, and then region-based video tracking is performed to accumulate the variable instances by constraining the appearance consistency and spatial correspondence. The concept detector can thus be progressively improved based on these newly tracked instances. After this process repeats again and again, a very mature concept detector can be obtained, as the growing baby brain.

Finally, when with enough instances for the new concept, the pre-trained CNN can also be further improved/fine-tuned, which can provide better knowledge base (i.e. deep features) for learning concept detectors. In this way, our proposed framework can effortlessly improve any new concept detector based on the prior knowledge, very few positive instances and large easily-obtained video data. The new concept detector is gradually improved in a never ending way as long as more videos are continuously explored.

Extensive experiments on three challenging object detection datasets including Pascal VOC 2007, VOC 2010 and VOC 2012 well demonstrate the superiority of the proposed computational baby learning framework over other state-of-the-arts [12] [20] [26] [11]. For all three datasets, we only need to learn one detector for each concept, while all previous works train different models for different datasets. Our framework beats other state-of-the-arts by learning from only two positive instances along with about 20,000 unlabeled videos.

The contributions of this paper can be summarized as the followings. 1) To our best knowledge, the proposed framework makes the first attempt to effectively mimic the baby learning process with the computational techniques, and integrate the prior knowledge modelling, exemplar learning and learning with video contexts. 2) Only two positive instances are required for learning a new concept detector and then the detector is refined with new variable instances from fully unlabeled real-world videos. There is no assumption that a video must contain a specific object, which makes our framework more scalable and robust for learning more concept detectors in an online way. 3) The knowledge of learned concepts will be effectively retained in our model (like baby memory) and conveniently utilized to learn new concepts.

2. Related Work

Recently, Convolutional Neural Networks (CNNs) have been shown to perform well in a variety of vision tasks

with millions of annotated training images and thousands of categories, including classification [23], detection [22] [7], segmentation [9] [13] and fine-grained recognition [30]. Notably, Krizhevsky et al. [14] and Christian et al. [23] achieved great progress in the image classification task with large and deep supervised CNN training. Girshick et al. [12] proposed to fine-tune the pre-trained Krizhevsky's network with the PASCAL VOC dataset and achieved the state-of-the-art object detection performance. However, the large performance increase achieved by these methods is only possible due to massive efforts on manually annotating millions of images which can provide good coverage of the space of possible appearance variations.

To minimize human efforts, some attempts have been devoted to learning reliable and robust models with very few labeled data. The methods can be summarized into two categories: learning from unlabeled web images (image-based) or video data (video-based). For the first category, existing image-based approaches [21] [2] [6] [3] iteratively used image search and detection results to cover more variations. Also text-based [6] and semantic relationships [2] [3] were further used to provide more constraints on selecting instances. One problem with these approaches is that the data variations (e.g., different viewpoints or background clutters) cannot be effectively expanded when only with image-based visual similarities. Some other works proposed to transfer the annotated image-level labels [10] or ground-truth bounding boxes [25] from labeled images to unlabeled images for semantically related classes. However, still a lot of labeled images are required to build the adequate subspaces for knowledge transferring. For the second category, video-based approaches [19][29][1] utilized intrinsic motion cues and appearance correlations within temporal adjacent frames to augment the model training. All existing methods trained their models on the weakly-labeled data where each video is assumed to contain objects of a target class. Static objects (e.g., chair or TV monitor) cannot be localized because they heavily rely on motion cues. Our framework differs from the above ones: 1) all the components (i.e. prior knowledge modelling, exemplar learning and learning with video contexts) benefit from each other and can be gradually improved as the process iterates; 2) our framework utilizes fully unlabeled real-world videos, and no additional weakly labelling is required; 3) any concept detector can be boosted from only two positive instances, including moving or static objects.

3. Computational Baby Learning Framework

3.1. Framework Overview

Figure 1 shows our proposed computational baby learning framework. Our method mimics the baby learning process based on prior knowledge modelling, exemplar learning and learning with video contexts. More specifically, the prior knowledge is modeled with a pre-trained CNN and the convolution layers constitute the knowledge base of the baby brain. Given very few positive instances for each new concept, an initial concept detector can be learned with exemplar learning over the deep features from the pre-trained CNN. More variable instances (e.g., those with different views or occlusions) can be obtained by exploring from real-world videos. After that, the detector can be fine-tuned based on these new instances. This process is repeated again and again to obtain a mature detector. The pre-trained CNN can also be gradually fine-tuned to generate more informative features along with more positive instances, which will improve the concept detector.

3.2. Prior Knowledge Modelling

We model the prior knowledge of a baby, inherited from genes and accumulated from life experience, with the pre-trained CNN. Two steps are performed to model the inherited genes and life experience. First, to simulate the genes of the baby, we pre-train a general CNN on the Imagenet [5] with image-level annotations. Since different babies usually inherit different genes and thus have different learning capabilities, we explore two CNN architectures for pre-training: the 7-layer architecture by Krizhevsky et al. [14] and the Network in Network (NIN) proposed by Lin et al. [15]. We use the same parameter settings for these two network architectures as in [14] and [15]. Second, we model the life experience of a baby with domain-specific fine-tuning on the previous pre-trained CNN. The accumulated life experience can be treated as the learned concepts in the baby brain. Because we validate our framework on the PASCAL VOC challenge, we thus use the 179 object classes on the ILSVRC2013 detection dataset as the learned concepts, which exclude the corresponding 21 classes of the VOC 20 classes. During fine-tuning, we only replace the 1000-way classification layer of the pre-trained CNN with a randomly initialized (N+1)-way classification layer (where N is the number of learned concepts, plus one for background). In our setting, N = 179. We use the validation set (20,121 images) in the ILSVRC2013 detection dataset and only the images that contain any object of the 179 classes are used. All region proposals with ≥ 0.5 intersection-over-union (IoU) overlap with a ground-truth box are regarded as positives for 179 learned concepts and the rest as negatives. We choose the selective search [24] to enable a controlled comparison with the previous detection work [12], though our framework can use any category-independent region proposal method. The CNN fine-tuning starts SGD with a learning rate of 0.001 for both two networks. For the 7-layer architecture [14], we uniformly sample 32 positive windows (over all classes) and 96 background windows to construct a mini-batch of size 128. The fine-tuning is run for 70k SGD iterations and takes 9 hours on a single NVIDIA GeForce

Figure 2. Some exemplar negative samples. Top row shows the collected general background images and bottom row shows the exemplar instances of previously learned concepts.

GTX TITAN GPU. For NIN [15], a mini-batch of size 80, consisting of 20 positive windows and 60 background windows, is used. The fine-tuning is run for 150k SGD iterations and takes 16 hours on a single NVIDIA GeForce GTX TITAN GPU. Alternatively, our framework can also boost any concept detector without any previously learned concepts just like the newborn baby, which can be simply implemented by eliminating the fine-tuning step with learned concepts.

3.3. Exemplar Learning

When only a few positive instances are taught by parent(s) or others, a baby possesses the initial recognition capability for the new concept. In our framework, the convolution layers in the above pre-trained CNN form the knowledge base of a baby, which interprets how a baby understands the objects layer by layer, and an initial concept detector can be learned based on the pre-trained CNN and a few positive instances of a new concept.

Feature extraction. For all positive and negative instances, we enlarge the tight bounding boxes to contain 16 pixels of image contexts and then wrap it into a fixed 227 $\times$ 227 size as used in [12]. Deep features are computed as the outputs from the penultimate fully-connected layer (4096-dimension) by forward propagating a mean-subtracted 227 $\times$ 227 image through the pre-trained CNN.

Selection of two positive instances. Good selection of the two positive instances often leads to the good generality of the initial detector. In real life, the parent(s) or others often teach the baby a few common instances. In this paper, we select two common positive instances for each concept from the PASCAL VOC 2007 training set. Specifically, we cluster all positive instances into 10 clusters by k-means. For the top-2 largest clusters, two samples nearest to the two centroids are selected as two seeds for each concept.

Negative set collection. To fairly justify our method, we do not use any annotations of PASCAL VOC Challenge to obtain the negative instances. The negative set used in our framework contains a batch of general background images (i.e. no specific object is included) and learned concept instances. As illustrated in the top row of Figure 2, we collect $4,000$ general scene images from Flickr and use the categories in the SUN scene dataset [28] as the search keywords. All region proposals in these background images are used as negative samples. For the learned concepts, the region proposals with ≥ 0.5 IoU overlap with the bounding box of 179 object class instances in the ILSVRC 2013 detection validation set are also treated as negative samples, as shown in the bottom row of Figure 2. Our initial experiment indicates that only using general background images, versus our negative set, results in about 4% drop in mAP.

Exemplar SVM training. Inspired by [17], we train a separate linear SVM classifier for each positive instance, and each SVM classifier is highly tuned according to the exemplar's appearance. The exemplar's decision boundary is thus decided, in large part, by the negative samples. For each test image, we thus independently run each exemplar detector and use the non-maximum suppression to create a final set of detections.

3.4. Learning with Video Contexts

The baby can gradually improve recognition capability by observing and/or interacting with the real instances in the physical world. We mimic this process by iteratively improving the concept detectors by mining more variable instances from the real-world videos. About 20,000 videos for each new concept are crawled from the online video sharing website (i.e., YouTube.com) using the keywords from the VOC dataset collection. No manual annotation for these videos is performed. In each iteration, we select one seed instance with highest score for each clip, and then region-based tracking is performed to accumulate the variable instances. Finally, detector and knowledge updating are performed to update the concept detectors and deep features, respectively. The four steps are described in detail in the following.

Seed instance selection. For each video clip, there is much redundant information with few appearance differences in temporal adjacent frames. To guarantee appearance variance of tracked instances and limit computational complexity, we only analyze key frames of each video. We select the image with ℓ_2 norm of the global GIST [18] feature difference larger than 0.01 as a key frame, compared with its temporal adjacent frames. For all key frames, we perform object detection with the initial detector. We only select the video containing the region with detection score larger than 1, and the region with highest score in this video is selected as the seed positive instance.

Region-based video tracking. The region-based video tracking is performed on the selected videos and initialized with the seed positive instance as illustrated in Figure 3. In our framework, we treat the tracking task as a region-based cluster mining problem for both moving and static concepts. Specifically, we extract a batch of region proposals in all key frames using selective search [24] and represent each region r_i with both the deep feature $\mathbf{x_i}$ and the spatial coordinates $\mathbf{p_i} = (c_i^x, c_i^y, w_i, h_i)$ corresponding to the position, width

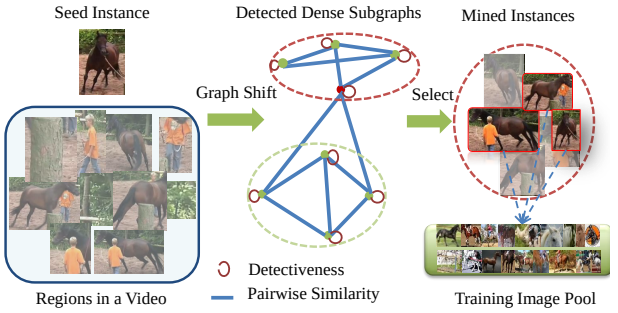

Figure 3. Region-based video tracking. Given the seed instance, we track other reliable instances from other regions. The affinity graph is built by combining both the pair-wise similarity and the detectiveness of each region. Then dense subgraphs are detected within the affinity graph by graph shift. The subgraph containing the seed instance (red point) is selected. Two instances with top-2 largest similarities with seed instances are placed into the training image pool for fine-tuning the detector in next iteration.

and height. Since we wish to select the instances from different frames, which may capture more diverse visual patterns, the similarity of two regions from the same frame is thus set as zero. The similarity $A_{i,j}$ for each pair (r_i, r_j) from different frames is thus defined by fusing the appearance similarity and the localization similarity,

$$A_{i,j} = \exp\{\frac{||\mathbf{x_i} - \mathbf{x_j}||^2}{\delta_1^2}\} + \alpha(\exp\{\frac{||\mathbf{p_i} - \mathbf{p_j}||^2}{\delta_2^2}\}), \quad (1)$$

where δ_1 and δ_2 are the empirical variances of $\mathbf{x}$ and $\mathbf{p}$, respectively, and we set $\alpha = 0.3$ because the appearance similarity is more important considering the camera moving and static objects. In addition, to make our framework robust to outliers (i.e. noisy regions), we also constrain the detection score to enlarge the possibility of the region to contain the concept. We thus define the detectiveness O_i of each region r_i by thresholding the detection score, where O_i of the region with detection score larger than -3 is set as 1 and otherwise set as 0. Note that we do not directly use the detector score because these variable instances may not be detected by the current detector but can bring more data diversity for further improving the detectors.

To seamlessly integrate the detectiveness of the region and the pairwise similarity, we use the graph shift algorithm [16] to obtain the variable instances. Formally, we define an individual graph $G = (R, A)$ for each video. $R = \{r_1, \ldots, r_n\}$ represents all the regions and A is a symmetric matrix with non-negative elements. The diagonal elements of A represent the detectiveness of the regions while the non-diagonal elements measure the pair-wise similarities between regions. To find these subgraphs, we can solve the quadratic optimization problem, i.e., $\max g(y) = y^T Ay, y \in \Delta^n$, where $\Delta^n = \{y \in R^n : y \geq 0 \text{ and } ||y||_1 = 1\}$, as in [16]. The strongly connected subgraphs correspond to large local maxima of $g(y)$. We can thus select the target subgraph that contains the seed instance. To prevent the rapid semantic drift, we only select two instances in this subgraph, which appear in different frames and have highest similarities with the seed instance.

Detector Updating. After observing enough real instances in real life, the baby can recognize unknown instances of one object he/she never saw. Similarly, with more mined instances from all the videos, a new large set of positive instances for the new concept is collected, which enables the concept detector to be further improved in the next iteration. The frames selected in the previous iterations will not be considered in later iterations, which makes the model equipped with different instances in every iteration. For updating the concept detector, the newly mined instances are added into the positive set. The regions from general background images and learned concepts are treated as negatives. Once the deep features are extracted for all positive and negative instances, we re-train one linear SVM for each new concept and the hard negative mining method is also used. After this process repeats again and again, we can achieve a very mature concept detector with a considerable number of mined instances. For fair comparison, we use the same detection strategies as the previous work [12] in testing.

Knowledge Updating. With more life experience and accumulated knowledge, the cognitive ability of a baby can be progressively improved and better prior knowledge base is obtained. In our framework, once enough instances of each new concept (about 10,000 instances) are obtained, the pre-trained CNN (i.e. prior knowledge of a baby) can be further improved to generate more informative features by fine-tuning it with these new positive instances. During fine-tuning, we replace the (N+1)-way output layer of the pre-trained CNN in Section 3.2 with a randomly initialized (M+1)-way classification layer (including M new concepts and one for background). We set $M = 20$ in our experiments. Since these mined instances may contain some noisy data (e.g. inaccurate bounding box of the object), we only use the original set of mined instances during fine-tuning and no additional data augmentation (e.g., ≥ 0.5 IoU overlap) is performed. The negatives for training concept detectors are used as background. The fine-tuning is run for 50K SGD iterations for the 7-lay architecture [14] and 100K SGD iterations for NIN [15], respectively.

Finally, we also learn a bounding box regression model to fix many localization errors in the test stage as in [12]. From the mined positive instances, we select $2,000$ detected instances with highest detection scores in the later iterations as ground-truth boxes for training the regression model. We believe the concept detector in the later iterations will be very mature and these top detection boxes have high possibilities to locate the precise object locations. We only learn from a region proposal if it has an IoU overlap with ground-truth box greater than 0.8.

Table 1. **Detection average precision (%) on PASCAL VOC2007 test.** Rows 1-3 show the baselines. Rows 4-5 show R-CNN results fine-tuned with 179 extra classes in ILSVRC 2013 detection. Rows 6-10 show our results in different iterations, with/without fine-tuning and bounding box regression. "B_initial" and "B_I15" represent the results in the beginning with two positive instances and after the 15th iteration, respectively. After that, "B_FT" and "B_FT_I2" are the results after fine-tuning with the mined new instances and running 2 more iterations, respectively. Rows 11-14 show the results based on NIN [15].

VOC 2007 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
DPM HSC[20]	32.2	58.3	11.5	16.3	30.6	49.9	54.8	23.5	21.5	27.7	34.0	13.7	58.1	51.6	39.9	12.4	23.5	34.4	47.4	45.2	34.3
R-CNN[12]	64.2	69.7	50.0	41.9	32.0	62.6	71.0	60.7	32.7	58.5	46.5	56.1	60.6	66.8	54.2	31.5	52.8	48.9	57.9	64.7	54.2
R-CNN BB[12]	68.1	72.8	56.8	43.0	36.8	66.3	74.2	67.6	34.4	63.5	54.5	61.2	69.1	68.6	58.7	33.4	62.9	51.1	62.5	64.8	58.5
R-CNN 179	62.5	70.2	54.4	42.7	35.4	63.1	71.9	61.5	34.0	61.0	47.1	60.7	64.1	67.9	56.8	32.6	58.2	45.7	59.2	64.5	55.7
R-CNN 179 BB	69.8	73.2	60.2	43.8	38.7	66.2	75.2	65.3	36.1	66.8	56.1	65.0	70.7	70.8	60.6	33.7	64.2	49.1	64.2	65.2	59.7
B_Initial	26.3	11.9	3.2	12.9	9.3	16.0	2.5	6.4	0.9	14.3	4.1	9.7	13.2	21.6	13.7	6.0	15.9	3.5	11.1	31.4	11.7
B_I15	61.1	65.7	51.1	38.8	29.8	57.4	63.8	57.8	26.8	57.1	44.3	57.2	55.7	61.3	45.5	27.2	57.1	38.0	50.4	58.0	50.3
B_FT	65.2	71.6	53.8	39.5	32.2	64.1	70.4	63.0	33.9	60.9	50.2	58.5	64.8	65.9	54.0	27.4	60.6	45.8	59.3	60.7	55.1
B_FT_I2	68.9	70.5	55.6	42.7	37.0	64.1	71.1	66.1	34.5	63.1	51.8	60.9	63.0	67.1	52.8	31.6	62.1	45.8	57.6	64.2	56.5
B_FT_I2_BB	72.2	72.8	61.8	46.7	42.0	66.1	74.2	74.6	37.3	68.3	56.8	65.7	71.3	68.4	58.0	35.1	66.3	47.2	64.0	65.7	60.7
B_NIN_I15	69.0	69.4	52.3	42.4	36.3	65.6	68.9	67.5	33.2	70.7	50.3	68.1	68.8	68.9	40.1	26.3	67.3	57.1	61.5	67.6	57.6
B_NIN_FT	71.1	71.5	59.0	43.7	37.1	68.1	73.1	72.8	39.8	72.1	55.3	68.3	67.6	70.7	54.8	35.4	68.4	58.2	64.9	66.2	60.9
B_NIN_FT_I2	71.0	73.6	61.3	46.3	40.6	70.3	73.8	74.0	43.7	72.9	55.6	68.5	69.2	70.7	57.6	37.7	69.3	59.6	65.3	68.7	62.5
B_NIN_BB	**75.9**	**76.8**	**66.9**	**49.0**	**47.9**	**72.1**	**77.2**	**77.9**	**48.6**	**78.5**	**65.0**	**73.9**	**77.3**	**73.6**	**62.7**	**40.4**	**73.5**	**64.4**	**69.2**	**70.6**	**67.1**

4. Experiments

4.1. Datasets and Settings

We evaluate our proposed computational baby learning framework on the PASCAL Visual Object Classes (VOC) datasets [8], which are widely used as the benchmark for object detection. PASCAL VOC 2007, VOC 2010 and VOC 2012 are used for our experiments. For each object class, we train the object detector by using the selected two positive instances and about $20,000$ raw videos. Note that our method is independent of any specific test set and only one object detector is used for testing the three datasets. For VOC 2010 and VOC 2012, we evaluate test results on the online evaluation server. We compare our method with the state-of-the-art baselines, including DPM HSC [20], Regionlets [26], SegDPM [11] and R-CNN [12]. They used all data in the VOC train and val set for training detectors. We mainly compare our framework with R-CNN due to its highest performance. In addition, to learn prior knowledge, we use 179 extra classes in the ILSVRC 2013 detection task to fine-tune the pre-trained CNN by 1000 classes in the ILSVRC 2012 classification. We implement two versions of R-CNN (i.e. "R-CNN 179" and "R-CNN 179 BB" with bounding-box regression), which first fine-tune the classification CNN with 179 extra classes and then fine-tune the CNN with VOC 20 classes following the settings in [12].

4.2. Comparison with the State-of-the-arts

Table 1 shows the complete results on the PASCAL VOC 2007 test set, and Table 2 and 3 show the results on VOC 2010 and 2012, respectively. It can be seen that the "R-CNN 179 BB" only performs slightly better than "R-CNN BB" (e.g., smaller than 0.6% increase on VOC2010 and VOC2012). The main reason is that the samples from 179 extra classes provide limited additional information when enough instances of 20 classes are already used in [12]. However, when only two instances of a new concept are observed, our method can benefit from these instances for domain-specific fine-tuning. All the variants of our framework strongly outperform the methods [20] [26] [11] based on hand-crafted features learning and deformable part models. Based on the 7-layer architecture [14], our method ("B_FT_I2_BB") achieves 60.7% in mAP, which is significantly superior to 58.5% of "R-CNN" [12] and 34.3% of "DPM HSC" [12]. Compared to R-CNN, our method also increases the performance by 2.8% and 2.7% on VOC 2010 and VOC 2012, respectively. When fine-tuning the CNN based on the Network in Network (NIN) [15], our method ("B_NIN_BB") can achieve 67.1% on VOC2007, 63.8% on VOC 2010, and 63.2% on VOC2012, which outperforms the "R-CNN BB" with a large margin of more than 8% on all three test sets. The bounding box regression can further fix a large number of mislocalized detections, boosting mAP by more than 3.5%. Note that our method only uses two positive instances and trains one single detector for all three datasets, while the baselines use different large training sets and carefully tune the model parameters for different test sets. This superiority well verifies the effectiveness and generality of our framework that automatically learns a significantly better detector than the fully supervised methods, and our mined new instances can successfully cover the large variance in the large manually labeled datasets (e.g., PASCAL VOC).

4.3. Computational Baby Learning Results

Figure 5 shows our performances in different iterations as well as before/after fine-tuning when more variable instances are collected. We show the results based on the 7-

Table 2. **Detection average precision (%) on PASCAL VOC2010 test.**

VOC 2010 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
Regionlets[26]	65.0	48.9	25.9	24.6	24.5	56.1	54.5	51.2	17.0	28.9	30.2	35.8	40.2	55.7	43.5	14.3	43.9	32.6	54.0	45.9	39.7
SegDPM[11]	61.4	53.4	25.6	25.2	35.5	51.7	50.6	50.8	19.3	33.8	26.8	40.4	48.3	54.4	47.1	14.8	38.7	35.0	52.8	43.1	40.4
R-CNN BB[12]	71.8	65.8	53.0	36.8	35.9	59.7	60.0	69.9	27.9	50.6	41.4	70.0	62.0	69.0	58.1	29.5	59.4	39.3	61.2	52.4	53.7
R-CNN 179	69.3	63.9	50.3	35.4	31.3	56.4	57.6	69.1	26.1	48.5	37.9	67.7	60.7	67.8	55.6	28.0	56.2	38.2	54.4	49.4	51.2
R-CNN 179 BB	73.1	66.2	55.2	38.7	36.5	59.2	60.2	71.8	27.1	51.6	42.0	69.5	63.9	71	59.2	29.7	60.0	38.5	61.8	51.2	54.3
B_FT_I2	70.5	66.3	54.3	40.0	34.3	58.4	58.7	67.6	26.8	54.1	38.2	67.7	63.6	70.2	55.3	26.6	60.1	38.7	50.6	53.8	52.8
B_FT_I2_BB	75.7	68.0	59.2	42.6	40.0	62.4	62.0	72.3	29.5	58.2	40.8	72.0	66.3	72.8	59.9	30.9	62.6	39.9	59.0	55.7	56.5
B_NIN_FT_I2	72.5	70.7	56.2	41.4	38.96	62.5	62.8	76.6	37.8	67.4	44.8	76.4	71.5	76.8	59.8	30.7	68.2	51.8	60.5	60.0	59.4
B_NIN_BB	**77.7**	**73.8**	**62.3**	**48.7**	**45.4**	**67.3**	**67.0**	**80.3**	**41.3**	**70.8**	**49.7**	**79.5**	**74.7**	**78.6**	**64.5**	**36.0**	**69.9**	**55.65**	**70.4**	**61.7**	**63.8**

Table 3. **Detection average precision (%) on PASCAL VOC2012 test.**

VOC 2012 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
R-CNN BB[12]	71.8	65.8	52.0	34.1	32.6	59.6	60.0	69.8	27.6	52.0	41.7	69.6	61.3	68.3	57.8	29.6	57.8	40.9	59.3	54.1	53.3
R-CNN 179	69.4	63.2	49.7	31.9	27.6	56.9	56.7	68.2	26.6	49.3	37.6	67.0	58.4	65.4	54.7	26.9	56.8	36.9	51.4	51.1	50.3
R-CNN 179 BB	72.7	66.0	55.0	34.0	32.0	59.5	60.5	70.9	29.2	51.5	40.2	70.0	62.3	68.4	58.9	30.8	58.2	37.7	58.3	53.9	53.5
B_FT_I2	70.7	66.0	53.2	37.1	31.9	59.0	59.0	67.6	26.6	54.6	37.8	67.1	63.3	69.1	54.8	26.4	58.8	39.5	49.6	54.7	52.3
B_FT_I2_BB	75.8	68.2	58.2	39.6	37.0	63.2	62.2	72.3	29.3	59.0	40.8	71.4	66.2	71.3	59.4	30.9	61.2	41.1	57.3	56.5	56.0
B_NIN_FT_I2	73.3	71.0	55.3	39.2	36.9	63.3	62.8	76.5	37.1	67.5	44.3	75.8	71.6	76.1	59.5	30.1	66.8	52.5	58.8	60.7	58.9
B_NIN_BB	**78.0**	**74.2**	**61.3**	**45.7**	**42.7**	**68.2**	**66.8**	**80.2**	**40.61**	**70.0**	**49.8**	**79.0**	**74.5**	**77.9**	**64.0**	**35.3**	**67.9**	**55.7**	**68.7**	**62.6**	**63.2**

layer architecture on VOC 2007 and the corresponding AP for each class is presented in Table 1. In the beginning, we only obtain 11.7% in mAP with only two positive instances based on the 7-layer architecture. After the first round of learning with video contexts, we can substantially improve the mAP to 36.1%, which is even higher than mAP of DPM HSC [20]. That is, most of the easy test samples can be detected by our updated model after the second round. After 15 iterations are performed, we can achieve 50.3% in mAP ("B_I15") and collect about 10,000 instances for each class. We then further fine-tune the prior knowledge CNN with these mined instances and 4.8% improvement is achieved ("B_FT"). Using NIN as the CNN architecture, we achieve 57.6% after 15 iterations ("B_NIN_I15") and obtain 60.9% after fine-tuning ("B_NIN_FT"). It proves that more informative features (better knowledge base) can be learned by further fine-tuning the CNN with more instances of the concept. And the superiority of "B_NIN_FT" over "B_FT" well demonstrates that more informative CNNs can lead to better initialization of our framework and learning capabilities during the repetition of the process.

We then further improve the detectors by running 2 more iterations based the fine-tuned CNN and better detection performances can be achieved, shown by "B_FT_I2", "B_NIN_FT_I2" and their versions with bounding box regression. For both static (e.g., chair) or moving objects (e.g., bicycle), our model can be gradually improved, which speaks well for the effectiveness of our region-based video tracking. Note that our performance can be continuously improved by learning more iterations and fine-tuning better CNNs with more instances. Due to the limited time, our experiments only report the current results at two iterations after fine-tuning CNN. Moreover, with better CNN architectures (e.g., googleLeNet [23]), it is predictable that our detectors can be further improved by taking advantage of better knowledge base.

4.4. Initialization with More Training Data

We also validate whether we can further improve the state-of-the-arts (e.g., R-CNN) trained with all training sets by using our baby learning framework. The experiments are shown on "B_VOC_I2" and "B_VOC_I2_BB" in Table 4, in which all the data in VOC 2007 are used instead of two positive instances. The detectors are trained over the deep features from the 7-layer prior knowledge CNN, described in Section 3.2 and two more iterations are performed to mine more variable instances from videos. We obtain 62.0% mAP on VOC 2007, 3.5% higher than the original "R-CNN BB" (58.5% in mAP) and 2.3% higher than "R-CNN 179 BB". Similar improvements are also achieved when employing the trained detectors on VOC 2010 (57.1% vs 53.7% of "R-CNN BB") and VOC 2012 (56.6% vs 53.3% of "R-CNN BB"). Based on the observations from this experiment, it can be observed that our framework can achieve better performance with more accurately labeled instances as the initialization.

4.5. Visualizations

We show the two selected positive instances and some mined variable instances for four classes in Figure 4. We randomly select some mined instance pairs from all iterations. It can be observed that our method successfully tracks other variable instances with different view-angles (e.g., results of bird and chair), occlusions (e.g., results of bicycle), appearance variance (e.g., results for TV monitor) and background clutters (e.g., results of bird). Qualitative detection

Table 4. **Detection average precision (%) on PASCAL VOC2007 test by the version initialized with more training data.**

R-CNN[12]	64.2	69.7	50.0	41.9	32.0	62.6	71.0	60.7	32.7	58.5	46.5	56.1	60.6	66.8	54.2	31.5	52.8	48.9	57.9	64.7	54.2
R-CNN BB[12]	68.1	72.8	56.8	43.0	36.8	66.3	74.2	67.6	34.4	63.5	54.5	61.2	69.1	68.6	58.7	33.4	62.9	51.1	62.5	64.8	58.5
B_VOC_I2	69.2	70.3	58.5	42.7	38.3	64.6	71.7	67.3	37.2	64.7	52.1	62.6	64.6	69.1	54.1	33.3	62.7	46.6	58.8	66.84	57.8
B_VOC_I2_BB	**72.3**	**72.7**	**64.0**	**46.7**	**43.5**	**67.5**	**74.8**	**74.5**	**39.4**	**70.1**	**57.7**	**68.3**	**71.5**	**70.2**	**58.5**	**36.9**	**68.1**	**49.1**	**65.5**	**67.9**	**62.0**

Figure 4. Visualization of our tracking results for bird, chair, bicycle and TV monitor. We randomly select the tracking results among all mined instances in all iterations. For each class, the left column shows the initialized two positive instances. The top row shows the detected seed instances in each video and two tracked instances are presented in the corresponding column within the dashed box.

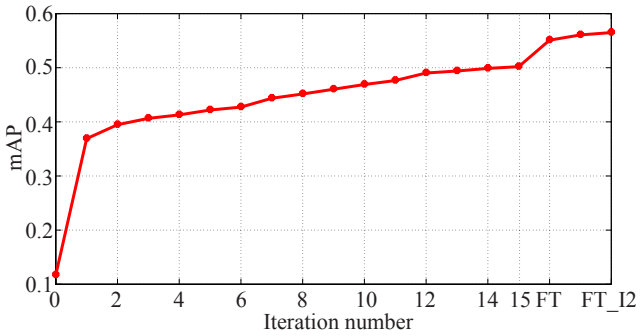

Figure 5. Performances in different iterations of our framework on VOC 2007. We run 15 iterations for mining more variable instances in videos to improve detectors. Then we fine-tune the CNN with mined new instances to generate more informative features ("FT") and 2 iterations are further performed to improve the detectors ("FT_I2").

results on the VOC 2007 test set are presented in Figure 6, which are obtained from our best model "B_NIN_BB". For each image, all detections from the detectors with a precision greater than 0.5 are shown, along with the detection scores. It can be observed that our model can effectively handle the heavy occlusions, different background clutters and viewpoint variance.

5. Conclusion and Future Work

In this paper, we presented a novel computational framework to interpret and mimic the baby learning process by prior knowledge modelling, exemplar learning and learning with video contexts. Based on the pre-trained CNN, we learned the initial concept detectors. More variable instances from real-world unlabeled videos were then mined to update the detectors and knowledge with the repetition of the process. Significant improvements over fully-training based methods were achieved by using our framework on PASCAL VOC 2007, VOC2010 and VOC 2012. In real life, when a baby cannot accurately identify a certain object, he/she shall ask parent(s) or others for help. In our future work, we will investigate how to adequately utilize the active learning to annotate uncertain samples in each iteration of the baby learning.

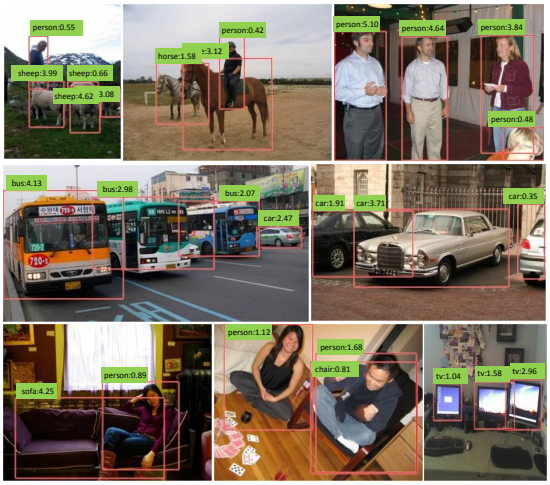

Figure 6. Some exemplar detection results. All detections with precision greater than 0.5 are shown. Each detection is labeled with the predicted class and the detection score from the detector. View digitally with recommended zoom.

References

[1] C.-Y. Chen and K. Grauman. Watching unlabeled video helps learn new human actions from very few labeled snapshots. In *Computer Vision and Pattern Recognition*, pages 572–579, 2013. 1, 3

[2] X. Chen, A. Shrivastava, and A. Gupta. Neil: Extracting visual knowledge from web data. In *International Conference on Computer Vision*, pages 1409–1416, 2013. 1, 3

[3] X. Chen, A. Shrivastava, and A. Gupta. Enriching visual knowledge bases via object discovery and segmentation. *Computer Vision and Pattern Recognition*, 2014. 1, 3

[4] J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, and H. Adam. Large-scale object classification using label relation graphs. In *European Conference on Computer Vision*, pages 48–64. 2014. 1

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, pages 248–255, 2009. 3

[6] S. K. Divvala, A. Farhadi, and C. Guestrin. Learning everything about anything: Webly-supervised visual concept learning. In *Computer Vision and Pattern Recognition*, 2014. 1, 3

[7] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. 2013. 3

[8] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 1, 6

[9] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1915–1929, 2013. 3

[10] R. Fergus, Y. Weiss, and A. Torralba. Semi-supervised learning in gigantic image collections. In *Advances in neural information processing systems*, pages 522–530, 2009. 3

[11] S. Fidler, R. Mottaghi, A. Yuille, and R. Urtasun. Bottom-up segmentation for top-down detection. In *Computer Vision and Pattern Recognition*, pages 3294–3301, 2013. 2, 6, 7

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv preprint arXiv:1311.2524*, 2013. 2, 3, 4, 5, 6, 7, 8

[13] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *European Conference on Computer Vision*, pages 297–312. 2014. 3

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1, 3, 5, 6

[15] M. Lin, Q. Cheng, and S. Yan. Network in network. *International Conference on Learning Representations*, 2014. 3, 4, 5, 6

[16] H. Liu and S. Yan. Robust graph mode seeking by graph shift. *International Conference on Machine Learning*, pages 671–678, 2010. 5

[17] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *International Conference on Computer Vision*, pages 89–96, 2011. 2, 4

[18] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001. 4

[19] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari. Learning object class detectors from weakly annotated video. In *Computer Vision and Pattern Recognition*, pages 3282–3289, 2012. 3

[20] X. Ren and D. Ramanan. Histograms of sparse codes for object detection. In *Computer Vision and Pattern Recognition*, pages 3246–3253, 2013. 2, 6, 7

[21] C. Rosenberg, M. Hebert, and H. Schneiderman. Semi-supervised self-training of object detection models. In *Computer Vision and Pattern Recognition*, pages 29–36, 2005. 1, 3

[22] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013. 3

[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014. 1, 3, 7

[24] K. E. Van de Sande, J. R. Uijlings, T. Gevers, and A. W. Smeulders. Segmentation as selective search for object recognition. In *International Conference on Computer Vision*, pages 1879–1886, 2011. 3, 4

[25] A. Vezhnevets and V. Ferrari. Associative embeddings for large-scale knowledge transfer with self-assessment. *arXiv preprint arXiv:1312.3240*, 2013. 3

[26] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In *International Conference on Computer Vision*, pages 17–24, 2013. 2, 6, 7

[27] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. Cnn: Single-label to multi-label. *arXiv preprint arXiv:1406.5726*, 2014. 1

[28] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer Vision and Pattern Recognition*, pages 3485–3492, 2010. 4

[29] Y. Yang, G. Shu, and M. Shah. Semi-supervised learning of feature hierarchies for object detection in a video. In *Computer Vision and Pattern Recognition*, pages 1650–1657, 2013. 3

[30] N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based r-cnns for fine-grained category detection. In *European Conference on Computer Vision*, pages 834–849. 2014. 3