Postprint of Dynamic Hand-held Object Recognition Based on SSU-SGD

Authors: Zhao Wencang, Chen Congcong, Zheng Honglei

Date: 2018-12-13T00:00:00+00:00

Abstract

Continuous targets contain richer information. To better capture visual information from dynamic hand-held objects, targeting dynamic hand-held objects in various backgrounds, three benchmarks suitable for dynamic hand-held object recognition are proposed based on the step-size self-learning updated SGD algorithm (abbreviated as SSU-SGD). By self-learning different step sizes, consolidation training is conducted respectively on the basis of known classes, unknown classes, and known objects for subsequent dynamic hand-held object recognition. Programming experiments and simulations are performed on naive strategies and cumulative strategies under the three different benchmarks using AlexNet and VGG networks. Experimental verification demonstrates that this method can effectively improve running speed and training accuracy, and significantly enhances the real-time performance of the dynamic hand-held object recognition process, enabling further practical applications.

Full Text

Preamble

Vol. 37 No. 2
Application Research of Computers

Dynamic Handheld Object Recognition Based on SSU-SGD

Zhao Wencang, Chen Congcong, Zheng Honglei
(Institute of Automation & Electronic Engineering, Qingdao University of Science & Technology, Qingdao, Shandong 266061, China)

Abstract: Continuous objects contain richer information. To better capture visual information from dynamic handheld objects across different backgrounds, this paper proposes three benchmarks for dynamic handheld object recognition based on the Step-size Self-learning Update SGD (SSU-SGD) algorithm. By

self-learning different step sizes, the model performs consolidation training on known classes, unknown classes, and known objects respectively for subsequent dynamic handheld object recognition. Programming experiments and simulations were conducted using AlexNet and VGG networks under both naive and cumulative strategies across the three benchmarks. Experimental verification demonstrates that this method can effectively improve running speed and training accuracy while enhancing the real-time performance of dynamic handheld object recognition, enabling further practical applications.

Keywords: continuous object; SSU-SGD algorithm; benchmark for dynamic handheld object recognition
Classification Number: TP391.41
DOI: 10.19734/j.issn.1001-3695.2018.05.0568

Introduction

Continuous objects are common in daily life, such as moving or rotating objects [1]. Continuous targets often reveal richer information, and analyzing consecutive frames can yield more meaningful visual information [2]. The human eye can not only perceive stationary objects but also recognize, locate, and track moving targets [3, 4]. From image recognition [5] to video detection technology [6], continuous object recognition has become a research hotspot in artificial intelligence.

Current single-frame image recognition technologies are relatively mature; however, recognizing high-dimensional data streams remains a challenging research problem [7]. Due to the massive sample libraries of high-dimensional data streams, recognition methods become more complex [8]. Deep learning methods have been successfully applied to image recognition [9], speech recognition [10], and natural language processing [11], and can similarly be applied to dynamic handheld object recognition.

This paper utilizes deep learning [12, 13] methods to recognize dynamic handheld objects [14], employing AlexNet and VGG networks for training. AlexNet [15] has achieved excellent results in image recognition but exhibits lower accuracy in continuous object recognition. VGG networks [16] are characterized by numerous convolutional layers and large computational requirements, resulting in slower training speeds. The proposed SSU-SGD algorithm enables parameters to self-learn different step sizes based on feature frequency, avoiding the tedious step-size tuning in traditional SGD [17, 18], improving iteration efficiency, and preventing iteration termination due to continuously decreasing step sizes. Based on this algorithm, three benchmarks suitable for dynamic handheld object recognition are proposed, which can effectively improve recognition accuracy and rapidly obtain optimal solutions for dynamic handheld object recognition problems, better applying deep learning methods to high-dimensional

data stream recognition. Experiments demonstrate that these SSU-SGD-based benchmarks can effectively improve training speed and classification accuracy.

1.1 SGD

Stochastic Gradient Descent (SGD) is a popular optimization algorithm in deep learning. Gradient descent methods can be categorized into three types: batch gradient descent requiring the entire training set, mini-batch gradient descent using partial samples, and stochastic gradient descent optimizing with a single random sample.

SGD minimizes the objective function by updating model parameters in the opposite direction of the gradient. For a training sample x_i and label y_i, the parameter update formula is as follows:

$$\lambda_{k+1} = \lambda_k - \eta \nabla J(\lambda_k; x_i, y_i)$$

where λ represents the weight coefficients, x_i and y_i denote the input and output features of the i-th group respectively. SGD adopts:

$$\lambda_{k+1} = \lambda_k - \eta \nabla h_\lambda(x_i)$$

Since SGD selects only one sample per calculation, it operates faster. However, gradient descent introduces significant fluctuations in the objective function value, resulting in high variance [19, 20]. Another critical issue is step-size selection; overly large or small step sizes can prevent proper convergence [21].

1.2 Implementation of the SSU-SGD Algorithm

To avoid the cumbersome step-size adjustment in SGD and improve iteration efficiency, this paper proposes the SSU-SGD algorithm. This self-learning step-size approach prevents training termination caused by continuously decreasing step sizes and addresses the problem of skipping optimal solutions due to improper step-size settings.

The core idea is to enable parameters to self-learn different step sizes based on feature frequency: parameters corresponding to frequently occurring features learn smaller step sizes, while parameters for infrequent features learn larger step sizes. Two step-size self-learning rules are defined:

Rule 1: When iteration update frequency is high, self-learn a smaller step size:
Rule 2: When iteration update frequency is low, self-learn a larger step size:

where η represents the update magnitude and ω denotes the iteration update frequency. When ω increases, η becomes smaller. The constant ϵ prevents the step size from gradually decreasing to zero during iterations, ensuring training does not terminate. When ω decreases, η gradually increases.

The SSU-SGD algorithm uses exponential decay averaging for rapid convergence upon finding convex structures. To prevent overfitting, the excess error after T iterations is analyzed, with magnitude $O(1/T)$. The SSU-SGD update formulas are given by:

$$\lambda_{t+1} = \lambda_t - \frac{\eta}{\sqrt{E[\Delta\lambda_t^2] + \epsilon}} \nabla J(\lambda_t)$$

where the root mean square errors are calculated as:

$$E[\Delta\lambda_t^2] = \gamma E[\Delta\lambda_{t-1}^2] + (1-\gamma)\nabla J(\lambda_t)^2$$

The algorithm is typically initialized with $\eta = 0.01$, allowing step sizes to adapt during learning. This avoids the denominator accumulation problem in standard SGD, ensuring network update capability does not weaken in later training stages. When applied to non-convex neural network training, the learning trajectory reaches a locally convex bowl region after passing through different structures, enabling rapid convergence as if initialized by SGD in that bowl.

The step-size update process is illustrated in the flowchart. The algorithm takes current iteration number n as input, initializes step size η_{in} and a minimal constant ϵ, then iteratively applies the self-learning rules until convergence.

2.1 Benchmarks for Dynamic Handheld Object Recognition

Applying deep learning to dynamic handheld object recognition [22] requires considering not only object shape, size, position, and lighting but also motion direction and trajectory [23-25]. The enormous training sample volume and complex, variable environments result in poor overall system performance and low recognition rates during each iteration. The SSU-SGD algorithm addresses this by enabling parameters to self-learn different step sizes based on feature frequency, rapidly obtaining optimal solutions for dynamic handheld object recognition.

Based on the SSU-SGD model, this paper proposes three benchmarks for dynamic handheld object recognition:

Benchmark 1: Self-learns different step sizes and iteration counts during training divided into 8 batches. The first batch trains all objects in a specific scene, and the classification results adjust subsequent dynamic handheld object recognition training. Since classification results are obtained in the first batch,

subsequent batches perform improvement and consolidation training on known classes.

Benchmark 2: Self-learns different step sizes and iteration counts during training divided into 8 batches. Each batch completes classification training across 8 scenes. The first batch' s results cannot be used for subsequent training, so each batch trains on unknown classes, with continuous batches performing consolidation training.

Benchmark 3: Self-learns different step sizes and iteration counts, with each batch performing classification training. Each training batch not only obtains classification results but also meets recognition requirements for each object across different scenes. Continuous batches perform improvement and consolidation training on both unknown classes and objects.

2.2 CORe50 Dataset

This paper employs the CORe50 dataset, which contains 50 objects divided into 10 categories for experiments. Considering object position and lighting, 300 RGB-D frames were collected across 11 different scenes (8 indoor and 3 outdoor), totaling 50×11×300 frames for training. Randomly selected 50×3×300 frames from 3 scenes (including indoor and outdoor) were used for testing, while the remaining 50×8×300 frames from 8 scenes were used for training. As shown in [Figure 1: see original paper], the dataset includes 10 categories of handheld items: plug adapters, mobile phones, scissors, light bulbs, cans, glasses, balls, markers, cups, and remote controls.

The experimental platform uses Linux Ubuntu 16.04 with an NVIDIA GTX 1080Ti GPU operating at 11 GHz with 11 GB available memory.

Based on the SSU-SGD algorithm, this paper proposes three benchmarks for dynamic handheld object recognition. Each benchmark self-learns different step sizes based on feature frequency. Experiments were conducted using AlexNet and VGG networks in Caffe. The CORe50 dataset' s 50×8×300 frames were used for training and 50×3×300 frames for testing. In dynamic handheld object recognition strategies, the naive strategy displays only current batch results, while the cumulative strategy displays results from current and previous batches, theoretically proving more effective. Both AlexNet and VGG networks were trained using naive and cumulative strategies under Benchmark 2 (B2), with the training hierarchy illustrated in [Figure 2: see original paper].

3.2 Experimental Results and Analysis

Experiments were conducted over 2000 iterations, with prediction accuracy serving as the performance metric. To evaluate the proposed method' s performance

on dynamic RGB-D scenes, experiments were performed according to the three benchmarks, yielding results shown in [Figure 3: see original paper] through [Figure 5: see original paper].

Benchmark 1 achieved high accuracy with both AlexNet and VGG networks, with both recognition strategies performing well. In Benchmarks 2 and 3, naive strategies suffered catastrophic forgetting across different networks, while cumulative strategies performed well. Benchmark 3 most closely approximates real-world dynamic handheld object recognition scenarios. By increasing training batches to 78, test result variations become more visible. Comparing [Figure 3: see original paper] through [Figure 5: see original paper], VGG network results reach approximately 70% accuracy, outperforming AlexNet, particularly in Benchmarks 2 and 3 where naive strategies become completely unusable. These comparisons demonstrate that VGG network with cumulative strategy best satisfies dynamic handheld object recognition requirements.

3.3 Verification of Test Results

To validate the proposed method, the three SSU-SGD-based benchmarks were compared against traditional recognition benchmarks: Mid-CNN from scratch and Mid-CNN+SVM. Training and testing on the CORe50 dataset produced the verification results shown in .

TABLE:1 Recognition accuracy under different benchmarks

Benchmark	AlexNet Naive	AlexNet Cumulative	VGG Naive	VGG Cumulative
Mid-CNN from scratch	-	-	-	52.3%
Mid-CNN+SVM	-	-	-	58.7%
Benchmark1	65.2%	68.4%	69.8%	71.2%
Benchmark2	45.1%	66.7%	48.3%	70.5%
Benchmark3	42.8%	65.9%	46.7%	69.8%

The results demonstrate that the proposed SSU-SGD-based dynamic benchmarks achieve higher recognition accuracy for dynamic handheld object recognition compared to Mid-CNN from scratch and Mid-CNN+SVM baselines.

A comparison between SSU-SGD and traditional SGD algorithms is presented in .

TABLE:2 Experimental comparison of SSU-SGD and SGD algorithms

Network	Strategy	Benchmark1	Benchmark2	Benchmark3
AlexNet	Naive	65.2%	45.1%	42.8%
AlexNet	Cumulative	68.4%	66.7%	65.9%
VGG	Naive	69.8%	48.3%	46.7%
VGG	Cumulative	71.2%	70.5%	69.8%

The results show that SSU-SGD-based benchmarks achieve higher recognition accuracy than traditional SGD for dynamic handheld object recognition.

4 Conclusion

This paper improves dynamic handheld object recognition benchmarks based on SSU-SGD, simultaneously training with AlexNet and VGG networks to improve training speed and recognition accuracy. The SSU-SGD algorithm ensures consistent results with less data compared to using full datasets, significantly increasing training speed while maintaining minimal training oscillations. The three proposed benchmarks outperform traditional Mid-CNN from scratch and Mid-CNN+SVM baselines for dynamic handheld object recognition. Due to current database limitations, testing on dynamic object recognition in complex environments was not conducted. Future work will involve constructing complex scene dynamic sample libraries to test the proposed method, improving algorithm reliability, and applying optimized algorithms to video target recognition.

References

[1] Bilen H, Fernando B, Gavves E, et al. Dynamic image networks for action recognition [C]//Computer Vision and Pattern Recognition. IEEE, 2016: 3034-3042.
[2] Shi Wei. Design of web interface based on visual information communication [J]. Revista De La Facultad De Ingenieria, 2017, 32(11): 679-684.
[3] Zhang Ling, Jiang Yi, Li Y, et al. Adaptive maneuvering target tracking with 2-HFSWR multisensor surveillance system [J]. IEEE Aerospace & Electronic Systems Magazine, 2018, 32(12): 70-76.
[4] Du Lan, Liu Bin, Wang Yan, et al. SAR image target detection algorithm based on convolution neural network [J]. Journal of electronics and information, 2016, 38(12): 3018-3025.
[5] Schwarz M, Schulz H, Behnke S. RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features [C]//Proc of International Conference on Robotics and Automation. IEEE, 2015: 1329-1335.
[6] Luo Zelun, Peng Boya and Huang De-An, et al. Unsupervised learning of

long-term motion dynamics for videos [J]. Computer Vision and Pattern Recognition, 2017: 7101-7110.
[7] Borji A, Izadi S, Itti L. iLab-20M: A large-scale controlled object dataset to investigate deep learning [C]//Computer Vision and Pattern Recognition. IEEE, 2016: 2221-2230.
[8] Liu Zhenan, Yan Tingrong, Zhang Rui. Dynamic image recognition in static background [J]. Computer technology and development, 2001, 11(1): 52-53.
[9] Horiguchi S, Amano S, Ogawa M, et al. Personalized classifier for food image recognition [J]. IEEE Trans on Multimedia, 2018, P(99): 1-1.
[10] Parthasarathy S, Rose R C. System and method for mobile automatic speech recognition [J]. Journal of the Acoustical Society of America, 2018, 124(6): 3373-3373.
[11] 褚晓敏, 朱巧明, 周国栋. 自然语言处理中的篇章主次关系研究 [J]. 计算机学报, 2017, 40(4): 842-860. (Zhu Xiaomin, Zhu Qiaoming and Zhou Guodong. Research on the relationship between major and subordinate in Natural Language Processing [J]. Journal of computer science, 2017, 40(4): 842-860.)
[12] Baldi P, Sadowski P, Whiteson D. Searching for exotic particles in high-energy physics with deep learning [J]. Nature Communications, 2014, 5(5): 4308.
[13] Zhou Jian, Troyanskaya Olga G. Predicting effects of noncoding variants with deep learning–based sequence model [J]. Nature Methods, 2015, 12(10): 931-934.
[14] Liu Weiwei. Video face detection based on deep learning [J]. Wireless Personal Communications, 2018(12): 1-16.
[15] Yuan Zhengwu, Zhang Jun. Feature extraction and image retrieval based on AlexNet [C]//Proc of the 8th International Conference on Digital Image Processing. 2016: 100330E.
[16] Verschuere B, Sophia V G G, Waldorp L, et al. What features of psychopathy might be central? A network analysis of the Psychopathy Checklist-Revised (PCL-R) in three large samples [J]. Journal of Abnormal Psychology, 2018, 127(1).
[17] Sa C D, Feldman M and Re C, et al. Understanding and optimizing asynchronous low-precision stochastic gradient descent [C]//Proc of ACM/IEEE, International Symposium on Computer Architecture. 2017.
[18] 邓卫钊. 随机梯度下降和对偶坐标下降算法的研究与应用 [D]. 秦皇岛: 燕山大学, 2016. (Deng Weizhao. Research and application of stochastic gradient descent and dual coordinate descent algorithm [D]. Qinhuangdao: Yanshan University, 2016.)
[19] 金钊. 基于改进随机梯度下降算法的 SVM [D]. 保定: 河北大学, 2017. (Jin Zhao. SVM based on improved stochastic gradient descent algorithm [D]. Baoding: Hebei University, 2017.)
[20] Klein S, Pluim J P, Staring M, et al. Adaptive stochastic gradient descent optimisation for image registration [J]. International Journal of Computer Vision, 2009, 81(3): 227.
[21] Gao Yuan, Ma Jiayi, Alan L. Yuile, et al. Semi-supervised sparse representation based classification for face recognition with Insufficient Labeled samples [J]. IEEE Trans on Image Processing A Publication of the IEEE Signal Process-

ing Society, 2017, 26(5): 2545.
[22] Liu Daqi, Yue Shigang. Event-driven continuous STDP learning with deep structure for visual pattern recognition [J]. IEEE Trans on Cybernetics, 2018: 1-14.
[23] Sobhani B, Paolini E and Giorgetti A, et al. Target tracking for UWB multistatic radar sensor networks [J]. IEEE Journal of Selected Topics in Signal Processing, 2017, 8(1): 125-136.
[24] Wang Xiao Feng, Zhang Minglu, Liu Jun. Research on robot perceptual learning based on incremental two-way principal component analysis [J]. Journal of electronic and information, 2018, 40(3): 618-625.
[25] Imran S, Ko Y B. A continuous object Boundary detection and tracking scheme for Failure-Prone sensor networks [J]. Sensors, 2017, 17(2).

Note: Figure translations are in progress. See original paper for figures.