

Visual Memorability for Robotic Interestingness via Unsupervised Online Learning

Chen Wang^[0000-0002-4630-0805],
Wenshan Wang^[0000-0002-4488-5619], Yuheng Qiu^[0000-0003-2966-8103],
Yafei Hu^[0000-0002-8142-1629], and Sebastian Scherer^[0000-0002-8373-4688]

Carnegie Mellon University, Pittsburgh, PA 15213, USA
chenwang@dr.com, {wenshanw,yuhengq,yafeih,basti}@andrew.cmu.edu
<https://github.com/wang-chen/interestingness>

Abstract. In this paper, we explore the problem of interesting scene prediction for mobile robots. This area is currently underexplored but is crucial for many practical applications such as autonomous exploration and decision making. Inspired by industrial demands, we first propose a novel translation-invariant visual memory for recalling and identifying interesting scenes, then design a three-stage architecture of long-term, short-term, and online learning. This enables our system to learn human-like experience, environmental knowledge, and online adaption, respectively. Our approach achieves much higher accuracy than the state-of-the-art algorithms on challenging robotic interestingness datasets.

Keywords: Unsupervised, Online, Memorability, Interestingness

1 Introduction

Interesting scene prediction is crucial for autonomous exploration [28], which is one of the most fundamental capabilities of mobile robots. It has a significant impact on decision making and robot cooperation. For example, the finding of a door shown in Fig. 1 (f) may affect the future planing, the hole on the wall in Fig. 1 (h) may attract more attentions. However, prior algorithms often have difficulty when they are deployed to unknown environments, as the robots not only have to find interesting scenes, but also have to lose the interests on repetitive scenes, *i.e.*, interesting scenes may become uninteresting during robot exploration after repeatedly observing similar scenes or following moving objects. For example in Fig. 6, we expect to have high interests on the truck when it appears but loss the interests when it exists for a long time. Nevertheless, the recent approaches of interestingness detection [17,21], as well as saliency detection [41], anomaly detection [42,27], novelty detection [2], and meaningfulness detection [18] algorithms cannot achieve this online updates scheme.

To this end, we propose to establish an **online learning** scheme to search for interesting scenes for robot exploration tasks. On the other hand, existing algorithms are heavily dependent on back-propagation algorithm [32] for learning, which is very computational expensive. To solve this problem, we introduce



Fig. 1: In this paper, we aim to predict robotic interesting scenes, which are crucial for decision making and autonomous cooperation. To enable the behavior of online losing interests on repetitive scenes for exploration of mobile robots, we propose to establish an online update scheme for interesting scene prediction. This figure shows several examples of both uninteresting and interesting scenes in SubT data [1] taken by autonomous robots. The height of green strip located at the right of each image indicates the interestingness level predicted by our unsupervised online learning algorithm when it sees the scene for the first time.

a novel *translation-invariant 4-D visual memory* to identify and recall visually interesting scenes. Human beings have a great capacity to direct visual attention and judge the interestingness of a scene [6]. For mobile robots, we find the following properties are necessary to establish a sense of visual interestingness.

Unsupervised: As shown in Fig. 1, the interesting scenes in robot operating environments are often unique and unknown, thus the labels are normally difficult to obtain, but prior research mainly focuses on supervised methods [3,6] and suffers in prior unseen environments. We hypothesize that a sense of interestingness can be established in an unsupervised manner.

Task-dependent: In many practical applications, we might only know uninteresting scenes before a mission is started. In the example of tunnel exploration task in Fig. 1, the deployment will be more efficient and easier if the robots can be taught what is uninteresting within several minutes. In this sense, we argue that the visual interestingness prediction system should be able to learn from negative samples quickly without accessing to data from unsupervised learning, thus an incremental learning method is necessary. Note that we expect the model is capable of learning from negative samples, but it is not necessary for all tasks.

To achieve the above properties, we propose a three-stage architecture:

Long-term learning: In this stage, we expect a model to be trained off-line on a large amount of data in an unsupervised manner as human beings acquire common knowledge from experience. We also expect the training time on single machine to be no more than the order of days.

Short-term learning: For task-dependent knowledge, the model then should be able to learn from hundreds of uninteresting images in minutes. This can be done before a mission started and beneficial to quick robot deployment.

Online learning: During mission execution the system should express the top interests in real-time and the detected interests should be lost online when similar scenes appear frequently, regardless if they exist in the uninteresting images or not. Another important aspect for online learning is no data leakage, *i.e.*, each frame is proceed without using information from its subsequent frames. This is in contrast to prior works [21,17] and datasets [8], where interesting frames are selected after an entire sequence is processed [15]. Since robots need to respond in real-time, we require that our algorithms are able to adapt quickly. To measure such capability of online response, we propose a new evaluation metric.

In summary, our contributions are:

- We introduce an extremely simplified three-stage architecture for robotic interesting scene prediction, which is crucial for practical applications. Concretely, we leverage long-term learning to acquire human-link experience, short-term learning for quick robot deployment and task-related knowledge, and online learning for environment adaption and real-time response.
- To accelerate the short-term and online learning, we propose a novel 4-D visual memory to replace back-propagation. Concretely, we introduce cross-correlation similarity for translational invariance, which is crucial for perceiving video stream, we also introduce tangent operator for safe writing, which is crucial for incremental learning from negative samples.
- To measure the online performance, we propose a strict evaluation metric, *i.e.*, the area under the curve of online precision (AUC-OP) to jointly consider precision, recall rate, and online performance.
- It is demonstrated that our approach achieves much higher overall performance than the state-of-the-art algorithms.

2 Related Work

A learning system that encodes the three-stage architecture for interesting scene prediction has not been achieved, thus the formulation as well as performance evaluation will be quite different from prior approaches. Some works on interestingness prediction have different objectives [3], *e.g.*, Shen *et al.* aimed to predict human interestingness on social media [34]. In this section we will mainly review the related techniques, as some methods used in saliency, anomaly, and novelty detection are also useful for our work.

The definition of interestingness is subjective, thus the annotation has to be averaged over different participants. To mimic the human judgment, prior works have paid great attentions to investigate the relationship between human visual interestingness and image features [3]. They are typically inspired by psychological cues and heavily leverage human annotation for training, which results in a large family of supervised learning methods. For instance, Dhar *et al.* designed three hand-crafted rules, including attributes of composition, content, and sky-illumination to approximate both aesthetics and interestingness of images [10].

Jiang *et al.* extended image interestingness to video and evaluated hand-crafted visual features for predicting interestingness on the YouTube and Flickr datasets [21]. Fu *et al.* formulated interestingness as a problem of unified learning to rank, which is able to jointly identify human annotation outliers [11,12].

Deep neural networks played more and more significant roles in recent works on interestingness prediction. For example, Gygli *et al.* introduced VGG features [35] and leveraged a support vector regression model to predict the interestingness of animated GIFs [17]. Chaabouni *et al.* constructed a customized CNN model to identify salient and non-salient windows for video interestingness prediction [5]. Inspired by a human annotation procedure of pairwise comparison, Wang *et al.* combined two deep ranking networks [39] to obtain better performance, and this method ranked first in the 2017 interestingness prediction competition [9]. Shen *et al.* combined both CNN and LSTM [19] for feature learning to predict video interestingness [34] for media contents.

However, the aforementioned methods are highly dependent on human annotation for training, which is labor expensive and not suitable for interestingness search [6]. Some efforts for unsupervised learning of interestingness have been made in [20], where interesting events of videos are detected using the density ratio estimation algorithm with the HOG feature [7]. However, in practice the approach cannot adapt well to changing distributions.

In the long-term stage, we introduce an autoencoder [23] for unsupervised learning, which has been widely used for feature extraction in many applications. For example, Hasan *et al.* showed that an autoencoder is able to learn regular dynamics and identify irregularity in long-duration videos [18]. Zhang *et al.* introduced dropout into the autoencoder for pixel-wise saliency detection in images [41]. Zhao *et al.* proposed a spatio-temporal autoencoder to extract both spatial and temporal features for anomaly detection [43].

In order to learn online, we introduce a novel visual memory module into the convolutional neural networks. Visual memory has been widely investigated in neuroscience [30]. While in computer vision, memory aided neural networks received limited attentions and used for several different tasks. For example, Graves *et al.* proposed a differentiable neural Turing machines (NTM) [16], which coupled external memory with recurrent neural networks (RNN). Santoro *et al.* extended NTM and designed a module to efficiently access the memory [33]. Gong *et al.* introduced memory module into an auto-encoder to remember normal events for anomaly detection [13]. Kim *et al.* introduced the memory network into GANs to remember previously generated samples to alleviate the forgetting problem [22]. However, the memories in the above works are defined as flattened vectors, thus the spatial structural information cannot be retained. In this paper, we propose a translation-invariant memory module and introduce online learning to solve the problem of robotic interestingness prediction.

3 Visual Memory

To retain the structural information of visual inputs, the visual memory \mathbf{M} is defined as a 4-D tensor, *i.e.*, $\mathbf{M} \in \mathbb{R}^{n \times c \times h \times w}$, where n is the number

of memory cubes and c , h , and w are the channel, height, and width of each cube, respectively. Intuitively, memory writing is to encode visual inputs into the memory, while reading is to recall one's memory regarding the visual inputs.

3.1 Memory Writing

We desire that the visual memory is able to balance new visual inputs and old knowledge. To this end, we denote visual inputs at time t as $\mathbf{x}(t) \in \mathbb{R}^{c \times h \times w}$ and define the writing protocol for the i_{th} memory cube \mathbf{M}_i at time t as

$$\mathbf{M}_i(t) = (1 - \mathbf{w}_i) \cdot \mathbf{M}_i(t-1) + \mathbf{w}_i \cdot \mathbf{x}(t), \quad (1)$$

where \mathbf{w}_i is the i_{th} element of a weight vector $\mathbf{w} \in \mathbb{R}^n$,

$$\mathbf{w} = \sigma(\gamma_w \cdot \tan(\frac{\pi}{2} \cdot D(\mathbf{x}(t), \mathbf{M}(t-1)))), \quad (2)$$

where $\sigma(\cdot)$ is the softmax function and $D(\mathbf{x}, \mathbf{M})$ is a cosine similarity vector, in which the i_{th} element $D_i(\mathbf{x}, \mathbf{M})$ is

$$D_i(\mathbf{x}, \mathbf{M}) = \frac{\sum(\mathbf{x} \odot \mathbf{M}_i)}{\|\mathbf{x}\|_{\mathbf{F}} \cdot \|\mathbf{M}_i\|_{\mathbf{F}}}, \quad (3)$$

where \odot , \sum , and $\|\cdot\|_{\mathbf{F}}$ are element-wise product, elements summation, and Frobenius norm, respectively. The writing protocol in (1) is a moving average, whose learning speed can be controlled via the writing rate γ_w ($\gamma_w > 0$), so that the training samples can be learned with an expected speed.

It is worth noting that, to promote the sparsity of memory writing, we introduce a tangent operator in (2) to map the range of cosine similarity $[-1, 1]$ in (3) to $[-\inf, \inf]$, thus memory writing can be focused on fewer but more relevant cubes via the softmax function. This leads to easier incremental learning and efficient space usage, which will be further explained in Sec. 4.2 and Sec. 6.1.

3.2 Memory Reading

Recall that convolutional features (visual inputs) are invariant to small input translations due to the concatenation of pooling layers to convolutional layers [14]. To obtain invariance to large translations, we need other techniques such as data augmentation, which is very computationally heavy. To solve this problem, we introduce translation in memory reading, leveraging that the structural information of visual inputs are retained in memory writing. Denote 2-D circular translation along the width and height directions with (x, y) elements of the i_{th} memory cube at time t as $\mathbf{M}_i^{(x,y)}(t)$, memory reading $\mathbf{f}(t) \in \mathbb{R}^{c \times h \times w}$ is

$$\mathbf{f}(t) = \sum_{i=1}^n \mathbf{r}_i \cdot \mathbf{M}_i^{(x,y)}(t), \quad (4)$$

where \mathbf{r}_i is the i_{th} element of reading weight vector $\mathbf{r} \in \mathbb{R}^n$,

$$\mathbf{r} = \sigma(\gamma_r \cdot \tan(\frac{\pi}{2} \cdot S(\mathbf{x}(t), \mathbf{M}(t)))), \quad (5)$$

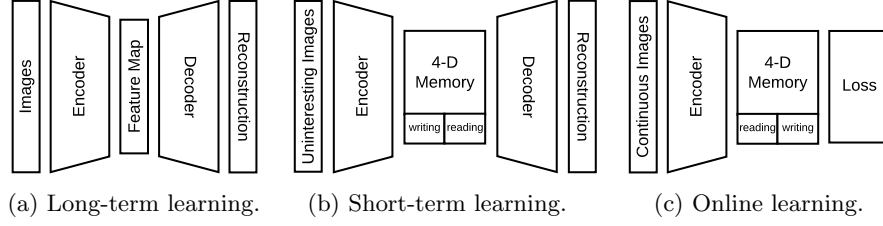


Fig. 2: The proposed three learning stages. (a) In long-term learning, the parameters in both encoder and decoder are trainable. (b) In short-term learning, the parameters in the encoder and decoder are frozen; the memory writing is performed before reading. (c) In online learning, the parameters in the encoder are frozen; the memory reading is performed before writing.

where $\gamma_r > 0$ is the reading rate. The i_{th} element of $S(\mathbf{x}, \mathbf{M})$ is the maximum cosine similarity of \mathbf{x} with $\mathbf{M}_i^{(a,b)}$, where $a = 0 : h - 1$ and $b = 0 : w - 1$ imply all translations. Intuitively, to find the maximum cosine similarity, we need to repeatedly compute (3) for translated memory cube $h \times w$ times, resulting in a high computational complexity. To solve this problem, we leverage the fast Fourier transform (FFT) to compute the cross-correlation [36]. Recall that 2-D cross-correlation is the inner-products between the first signal and circular translations of the second signal [38], we can compute $S_i(\mathbf{x}, \mathbf{M}_i)$ as

$$S_i(\mathbf{x}, \mathbf{M}) = \frac{\max \mathcal{F}^{-1}(\sum^c \hat{\mathbf{x}}^* \odot \hat{\mathbf{M}}_i)}{\|\mathbf{x}\|_{\mathbf{F}} \cdot \|\mathbf{M}_i\|_{\mathbf{F}}}, \quad (6)$$

where $\hat{\cdot}$ is the 2-D FFT, \cdot^* is the complex conjugate, and \sum^c is element-wise summation along channel dimension. The translation (x, y) in (4) for the i_{th} memory cube is corresponding to the location of the maximum response, *i.e.*,

$$(x, y) = \arg \max_{(a,b)} \left(\sum^c \hat{\mathbf{x}}^* \odot \hat{\mathbf{M}}_i \right) [a, b]. \quad (7)$$

In this way, the computational complexity for each memory cube can be reduced from $\mathcal{O}(ch^2w^2)$ to $\mathcal{O}(chw \log hw)$. Another advantage of translation-invariance in memory reading is that memory usage becomes more efficient, since scene translation is common in video stream for many robotic applications, *e.g.*, robot exploration and object search, which will be further explained in Sec. 6.3.

4 Learning

4.1 Long-term Learning

Inspired by the fact that human has a massive memory storage capacity [4], we use an autoencoder in Fig. 2a for long-term learning for the following reasons. **Unsupervised Knowledge:** A reconstruction model can be trained in an unsupervised way, hence we can collect massive number of images from the internet

or in real-time during execution to train the model without much efforts. This agrees with our objective of long-term learning that is to remember as many scenes as possible. In this stage, we still leverage back-propagation for the training, so that the large amount of knowledge will be ‘stored’ in the trainable parameters, which will be frozen afterwards. In this sense, the learned knowledge can be treated as unforgettable human-like experience.

Detailed and Semantic: To precisely reconstruct images by smaller feature maps, the output of the bottleneck layer has to contain both detailed and semantic information. This is crucial for visual interestingness, since both texture and object-level information may attract one’s interests. Feature maps are invariant to small translations due to CNN, we leverage the invariance of visual memory to large translations in short-term and online learning. We construct the encoder following VGG [35] and concatenate 5 deconvolutional blocks [26] for decoder.

4.2 Short-term Learning

As aforementioned, we normally only know the uninteresting scenes before a robotic mission is started. For known interesting objects, we prefer to use supervised object detectors. Therefore, we expect that our unsupervised model can be trained *incrementally* with negative labeled samples within *several minutes*. This is beneficial for learning environmental knowledge and quick robot deployment.

To this end, we propose the short-term learning architecture in Fig. 2b. The memory module is inserted into the trained reconstruction model, in which all parameters are frozen. For each sample, the output of the encoder is first written into the memory, then memory reading is taken as inputs of the decoder. Intuitively, the images cannot be reconstructed well initially, as feature maps are not fully learned by the memory, and memory reading will be different from the encoding outputs. In this sense, we can inspect the reconstruction error to know whether the memory has learned to encode the training samples or not.

The memory leaning is much faster than back-propagation and has several advantages. Recall that the gradient descent algorithms cannot be directly applied to neural networks for incremental learning, since all trainable parameters are changed during training, leading the model to be biased towards the augmented data (new negative labeled data), and forgetting the previously learned knowledge. Although we can train the model on the entire data, which takes the learned parameters from long-term learning as an initialization, it is too computationally expensive and cannot meet the requirements for short-term learning. Nevertheless, memory learning is able to solve this problem inherently. One of the reasons is that the tangent operator in (2) promotes writing sparsity, thus less memory cubes are affected, resulting in safer and faster incremental learning.

4.3 Online Learning

Online learning is one of the most important capabilities for a real-time visual interestingness prediction system, as human feelings always keep changing according to one’s environments and experiences. Moreover, people tend to lose interests when repeatedly observing the same objects or exploring the same scenes, which is very common in a video stream from a mobile robot. Therefore, we aim to establish such an online learning capability for real-time robotic sys-

tems, instead of selecting interesting frames after processing an entire sequence [6]. We design a few control variables that can be simply adjusted for different applications, *e.g.*, a hyper-parameter to control the rate of losing interests will be useful for objects search. To this end, we propose an architecture for online learning in Fig. 2c, in which only the frozen encoder and memory are involved.

In this stage, memory reading is performed before writing and the inputs are continuous image sequences (a video stream), which is different from short-term learning. If unobserved scenes or objects appear suddenly, memory reading confidence will be lower than before, which can be treated as a new interest. Moreover, since the new scenes or objects are then written into the memory, their reading confidence level will become higher in the following images. Therefore, the model will learn to lose interests on repetitive scenes once the scene is remembered by the memory. In this sense, a visual interestingness is negative correlated with the memory reading confidence. In experiments, we adopt averaged cosine similarity over feature channel to approximate the reading confidence.

During online learning, a large translation often happens during robot exploration, hence an invariance to large translations introduced in (4) is able to further reduce memory consumption and improve the system robustness.

5 Experiments

Evaluation Metric Prior research typically only focused on the precision or recall rate and is not able to capture the online response of interestingness. Therefore, we propose a new metric, *i.e.*, area under curve of online precision (AUC-OP) to evaluate one frame without using the information from its subsequent frames (no data leakage). This metric is stricter and jointly consider online response, precision, and recall rate. Intuitively, if K frames of a sequence are labeled as interesting in the ground truth, an algorithm is perfect if the set of its top K interesting frames are the same with the ground truth.

Consider a sequence $I_{[1:N]}$, we take an interestingness prediction $p(I_t)$ as a true positive (interesting) if and only if $p(I_t)$ ranks in the top $K_{t,n}$ among a subsequence $p(I_{t-n+1}), p(I_{t-n+2}) \dots, p(I_t)$, where $K_{t,n}$ is the number of interesting frames in the ground truth. Note that the subsequence $I_{[t-n+1:t]}$ only contains frames before I_t , as data leakage is not allowed in the online performance. Therefore, we may calculate an online precision score for length n subsequences as $s(n) = \sum \text{TP} / (\sum \text{TP} + \sum \text{FP})$, where TP and FP denote the number of true positives and false positives, respectively. Since all true positives rank in the top $K_{t,n}$, this means that no false negative is allowed. Recall that a recall rate can be calculated as $r = \sum \text{TP} / (\sum \text{TP} + \sum \text{FN})$, which means that the proposed online precision score $s(n)$ requires a 100% recall rate. For a better comparison, we often accept true positive predictions as ranking in the top $\delta \cdot K_{t,n}$, where $\delta \geq 1$. Therefore, the overall performance of that jointly considers online performance, precision, and recall rate is the AUC of online precision $s(\frac{n}{N}, \delta)$ where $\frac{n}{N} \in (0, 1]$, which considers all subsequence length as $n = [1 : N]$. In practice, we often allow some false negatives and $\delta = 2$ is recommended for most of exploration task.

Table 1: The SubT dataset. “Normal” and “Difficult” means that the percentage of frames labeled as interesting by at least 1 subjects or 2 subjects, respectively.

Video	I	II	III	IV	V	VI	VII	Overall
Length (min)	53.1	55.7	79.4	80.0	59.0	57.5	83.0	467.7
Normal (%)	11.11	15.07	9.37	17.51	24.52	22.77	11.04	15.14
Difficult (%)	2.76	4.49	3.02	4.29	4.07	3.30	3.21	3.58

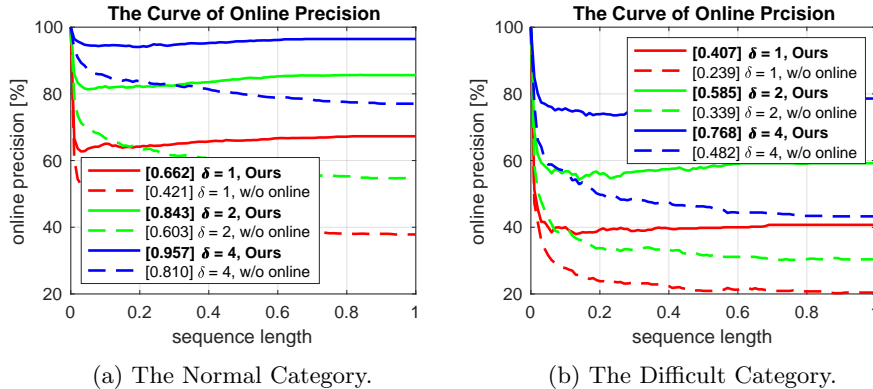


Fig. 3: The performance on SubT with and without (w/o) online learning.

Dataset To test the online performance on robotic systems for visual interesting scene prediction, we choose two datasets recorded by fully autonomous robots, *i.e.*, the SubT dataset [1] for unmanned ground vehicles (UGV) and the Drone Filming dataset [40] for unmanned aerial vehicles (UAV).

The SubT dataset is based on the DARPA Subterranean Challenge (SubT) Tunnel Circuit. In this challenge, the competitors are expected to build robotic systems to autonomously search and explore the subterranean environments. The environments pose significant challenges, including a lack of lighting, lack of GPS and wireless communication, dripping water, thick smoke, and cluttered or irregularly shaped environments. Each of the tunnels has a cumulative linear distance of 4-8 km. The dataset listed in Table 1 contains seven long videos (1h) recorded by two fully autonomous UGV from Team Explorer¹. Each sequence is evaluated by at least 3 persons. It can be seen that the SubT dataset is very challenging, as human annotation varies a lot, *i.e.*, only 15% and 3.6% of the frames are labeled as interesting by at least 1 (normal category) and 2 subjects (difficult category), respectively. Some of the interesting scenes predicted by our algorithms are presented in Fig. 1, in which we can see that our method predicts many interesting scenes correctly.

¹ Team Explorer won the first place at the DARPA SubT Tunnel Circuit.

Table 2: The comparison with the state-of-the-art method on AUC-OP.

(a) The SubT Normal Category.				(b) The SubT Difficult Category.			
Methods	$\delta = 1$	$\delta = 2$	$\delta = 3$	Methods	$\delta = 1$	$\delta = 2$	$\delta = 4$
baseline [25]	0.622	0.798	0.904	baseline [25]	0.352	0.544	0.700
ours	0.662	0.842	0.923	ours	0.407	0.585	0.768

The Drone Filming dataset [40] is recorded by quadcopters during autonomous aerial filming. It also contains challenging environments, *e.g.*, intensive light changes, severe vibrations, and motion blur, *etc.* Different from other sources such as surveillance camera, robotic visual systems pose extra challenges due to fast background changes, limited computational resources, and unique and even dangerous operating environments in which human beings cannot get access to.

Implementation In all experiments in this section, a memory capacity of 1000 and mean square error (MSE) loss are adopted. The memory reading and writing rate are set as $\gamma_r = \gamma_w = 5$. Our algorithm is implemented using the PyTorch library [29] and conducted on a single Nvidia GPU of GeForce GTX 1080Ti.

Efficiency During long-term learning, we perform unsupervised training with the coco dataset [24]. It takes about 3 days running on single GPU. For short-term learning, our model takes about 10 minutes for learning 912 uninteresting images in the SubT dataset, which is feasible for deployment purpose of most practical applications. For online learning, it runs about 72.01ms per frame on single GPU, which is feasible for real-time² robotic interestingness prediction.

Performance Online learning is able to remove many repetitive scenes thus it is able to reduce the number of false positives dramatically. The curve of online precision of our model for the normal category and difficult category are presented in Fig. 3a and Fig. 3b, respectively, where the overall AUC-OP is shown in the associated square brackets. It can be seen that our model achieves an average of 20% higher overall performance than the model without online learning, which verifies the importance of the proposed online learning.

Comparison To the best of our knowledge, robotic visual interestingness prediction is currently underexplored, and existing methods in saliency or anomaly detection have poor performance in this scenario. In this section, we select the state-of-the-art method, frame prediction in [25] as the baseline, which has very good generalization ability. Basically, it introduces temporal constraint into the video prediction task to detect anomaly. The overall performance of the AUC-OP of our method is presented in Table 2a and Table 2b, respectively. It can be seen that our method achieve an average of 4.0%, 4.4%, 1.9% and 5.5%, 4.1%, 6.8% higher overall accuracy in the two categories for $\delta = 1, 2, 4$, respectively, which verifies its effectiveness. We next present analysis to show the effects of the proposed writing sparsity, translational invariance, and short-term learning.

² Real-time means processing images as fast as human brain, *i.e.*, 100ms/frame [31].

Table 3: The effects of the proposed modules on SubT (AUC-OP).

Methods	Normal Category			Difficult Category		
	$\delta = 1$	$\delta = 2$	$\delta = 4$	$\delta = 1$	$\delta = 2$	$\delta = 4$
w/o sparsity	0.437	0.633	0.846	0.260	0.373	0.523
w/o invariance	0.330	0.510	0.752	0.212	0.268	0.379
w/o short-term	0.508	0.711	0.913	0.329	0.450	0.621
ours	0.662	0.842	0.957	0.407	0.585	0.768

Effect of Writing Sparsity To show its effectiveness, we replace our proposed writing protocol with the one used in [16], which is denoted as ‘without (w/o) sparsity’ in the first row of Table 3. It can be seen that our model achieves about 20-30% higher overall accuracy, which verifies the effectiveness of our method.

Effect of Translational Invariance Without the large translational invariance, the performance will drop a lot, as translational movement is very common in robotic applications. As shown in the second row of Table 3, our model achieves about 20-30% higher accuracy than the one w/o translational invariance.

Effect of Short-term Learning Short-term learning plays an important role for quick robot deployment. The performance can be largely improved if some uninteresting scenes are known before a mission. It can be seen in the fourth row of Table 3 that our model achieves about 10-20% higher accuracy than the one without short-term learning (w/o short-term).

6 Ablation Study

In this section, we further test the proposed algorithm and aim to provide intuitive explanations for the influences of the proposed writing protocol, memory capacity, translational invariance, and capability of losing interests. Following the ablation principle, all configurations are the same unless otherwise stated.

6.1 Writing Protocol

It has been pointed out that the memory learning is highly dependent on the writing vector in (1), in which a tangent operator is introduced for writing sparsity. This section explores this effect and compare with the writing vector in (8) used in [16]. Note that memory defined in [16] is vectors, thus it is not invariant to large translation. Following the ablation principle, we use the same 4-D memory structure and the reading protocol proposed in this paper.

$$\mathbf{w} = \text{softmax}(\gamma \cdot D(\mathbf{x}, \mathbf{M})), \quad (8)$$

where γ is a parameter. To show the writing performance, we write two random 3-D tensors into the memory, *i.e.*, \mathbf{f}_1 and \mathbf{f}_2 , and compare their reading accuracy in terms of cosine similarity in (9).

$$S^c(\mathbf{r}, \mathbf{f}) = \frac{\sum(\mathbf{r} \odot \mathbf{f})}{\|\mathbf{r}\|_{\mathbf{F}} \cdot \|\mathbf{f}\|_{\mathbf{F}}}, \quad (9)$$

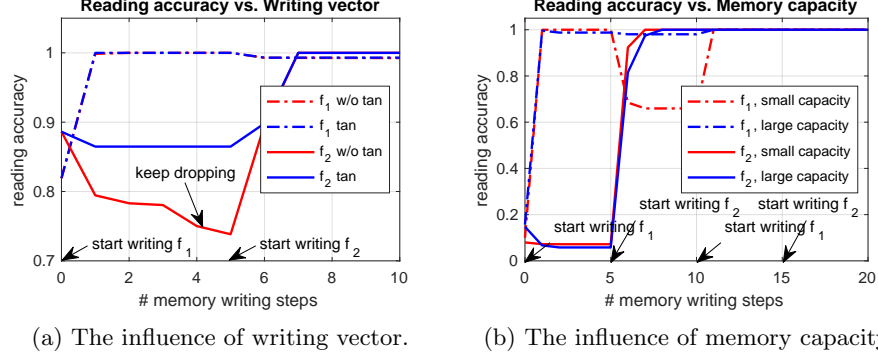


Fig. 4: The memory recall accuracy. (a) Writing vector with a tangent operator enables sparsity, thus less memory cubes are affected in learning. (b) Larger memory capacity leads to more computation but easy incremental learning.

where \mathbf{r} and \mathbf{f} are the memory reading and writing tensors, respectively. In experiments, we set $\gamma_w = \gamma = 5$ and write both \mathbf{f}_1 and \mathbf{f}_2 5 times continuously and show their reading accuracy in terms of number of writing in Fig. 4a. It can be seen that both memories are able to remember the random tensors after repeatedly writing. However, when writing a vector without a tangent operator, the accuracy of \mathbf{f}_2 keeps dropping even when \mathbf{f}_1 is learned, *i.e.*, $S^c(\mathbf{r}_1, \mathbf{f}_1) \approx 1$. This is because all memory cubes are affected due to the non-sparse writing vector in (8). This will be a severe issue when a robot keeps learning the same thing (observing the same scene), since the learned knowledge may be forgotten due to the non-sparse writing. Nevertheless, our proposed writing vector with the tangent operator is able to map the weight of \mathbf{f}_1 to infinite when \mathbf{f}_1 is learned, resulting in safer writing as only a few memory cubes are affected. This verifies the effectiveness of the proposed writing vector. We notice that sparsity is also mentioned in [42, 27, 13], while it is designed for different objectives using different strategies. For instance, [13] introduced a simple threshold and an entropy loss to promote sparsity for reducing reconstruction accuracy to detect anomaly.

6.2 Memory Capacity

This section explores the effects of memory capacity, *i.e.*, the number of memory cubes c , which is an important hyper-parameter for incremental learning. To this end, we write two same random 3-D tensors \mathbf{f}_1 and \mathbf{f}_2 five times sequentially into two different memories in terms of the memory capacity c . Their reading accuracy for the performance comparison is shown in Fig. 4b.

As can be seen, both memories are able to learn random samples, while the accuracy of \mathbf{f}_1 drops a lot for smaller capacity when start to write \mathbf{f}_2 , although it is remembered later when \mathbf{f}_1 is written again. We observe similar phenomenon when the number of samples is around the same or larger than the memory capacity. This means that a memory that has a small capacity quickly forgets old knowledge when learning new knowledge. We can also leverage this property



Fig. 5: Memory reading with translational invariance (WTI) recall translated scenes better than without translation-invariance (WOTI).

for model design, since uninteresting objects can become interesting in some cases. This means that for larger capacity, reading accuracy is less affected by new knowledge, resulting in safer and easier incremental learning.

6.3 Translational Invariance

Although CNN features are invariant to small translations [37], they still fail to recall memory when large translations occur. To solve this problem, we introduce translational invariance by cross-correlation in (6). We next test it on the Drone Filming dataset [40] in Fig. 5. In this sequence an ambulance appears suddenly in the 1st frame and disappears in the 5th frame. We construct two memory modules to learn this video based on the online learning strategy presented in Sec. 4.3. The first module adopts the cross-correlation similarity presented in Sec. 3.2 for memory reading (denote as WTI), while another one adopts the cosine similarity (WOTI). It can be seen that both modules cannot recall the memory for the 1st frame, since the ambulance is not seen before. However, the module WTI is able to recall the memory precisely in the subsequent frames, while the module WOTI quickly fails, although its reading is still meaningful, *e.g.*, the 2nd and 4th frames have correct patterns for sky, trees, and ground. It can be seen that the recalled memory for the 3rd frame from module WTI is roughly a translated replica of the 2nd frame of the video (this also occurs at the 4th and 5th frame), which means that the module WTI correctly takes the 2nd frame as the most similar scene to the 3rd frame. This phenomenon verifies the translational invariance of our proposed reading protocol.

Note that there is a small translation for the 2nd frame from WTI. This is because the invariance to small translations of CNN features, *i.e.*, the features look the same for visual memory, although they appear with a small translational difference. Therefore, our proposed cross-correlation similarity together with the CNN features contribute complete invariance of translation to memory recall.

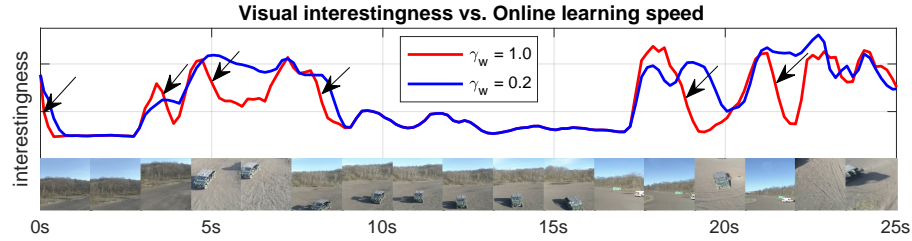


Fig. 6: The visual interestingness with different writing rates for drone video footage [40]. As indicated by the arrows, a larger writing rate results in a faster loss of interest for new objects during online learning.

6.4 Losing interest

To test the capability of losing interest of the algorithm, we perform a qualitative test on the Drone Filming dataset [40]. The objects tracked in the videos, *e.g.*, cars or bikes, are relatively stable, while the background keeps changing due to the movement of the objects. This makes it suitable for testing the capability of online learning. One of the video clips is shown in Fig. 6, where two different online learning speeds are adopted, *i.e.*, $\gamma_w = 1.0$ and $\gamma_w = 0.2$. It can be seen that the interestingness level of both settings become high when new objects or scenes appear, *i.e.*, both settings are able to detect novel objects. However, the interestingness level with a larger writing rate always drops faster, meaning it is quicker to lose interest of the similar scenes. This verifies our objective that a simple hyper-parameter can be adjusted for different missions.

7 Conclusion

In this paper, we developed an unsupervised online learning algorithm for visual robotic interestingness prediction. We first proposed a novel translation-invariant 4-D visual memory, which can be trained without back-propagation. To better fit for practical applications, we designed a three-stage learning architecture, *i.e.*, long-term, short-term, and online learning. Concretely, the long-term learning stage is responsible for human-like life-time knowledge accumulation and trained on unlabeled data via back-propagation. The short-term learning is responsible for learning environmental knowledge and trained via visual memory for quick robot deployment. The online learning is responsible for environment adaption and leverage the visual memory to identify the interesting scenes. The experiments show that, implemented on a single machine, our approach is able to learn online and find interesting scenes efficiently in real-world robotic tasks.

Acknowledgements

This work was sponsored by ONR grant #N0014-19-1-2266. The human subject survey was approved under #2019_00000522.

References

1. <http://theairlab.org/dataset/interestingness>
2. Abati, D., Porrello, A., Calderara, S., Cucchiara, R.: Latent space autoregression for novelty detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 481–490 (2019)
3. Amengual, X., Bosch, A., De La Rosa, J.L.: Review of methods to predict social image interestingness and memorability. In: International Conference on Computer Analysis of Images and Patterns. pp. 64–76. Springer (2015)
4. Brady, T.F., Konkle, T., Alvarez, G.A., Oliva, A.: Visual long-term memory has a massive storage capacity for object details. Proceedings of the National Academy of Sciences **105**(38), 14325–14329 (2008)
5. Chaabouni, S., Benois-Pineau, J., Zemmari, A., Amar, C.B.: Deep saliency: prediction of interestingness in video with cnn. In: Visual Content Indexing and Retrieval with Psycho-Visual Models, pp. 43–74. Springer (2017)
6. Constantin, M.G., Redi, M., Zen, G., Ionescu, B.: Computational understanding of visual interestingness beyond semantics: literature survey and analysis of covariates. ACM Computing Surveys (CSUR) **52**(2), 25 (2019)
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05). vol. 1, pp. 886–893. IEEE (2005)
8. Demarty, C.H., Sjöberg, M., Constantin, M.G., Duong, N.Q., Ionescu, B., Do, T.T., Wang, H.: Predicting interestingness of visual content. In: Visual Content Indexing and Retrieval with Psycho-Visual Models, pp. 233–265. Springer (2017)
9. Demarty, C.H., Sjöberg, M., Ionescu, B., Do, T.T., Gygli, M., Duong, N.: Mediaeval 2017 predicting media interestingness task (2017)
10. Dhar, S., Ordonez, V., Berg, T.L.: High level describable attributes for predicting aesthetics and interestingness. In: CVPR 2011. pp. 1657–1664. IEEE (2011)
11. Fu, Y., Hospedales, T.M., Xiang, T., Gong, S., Yao, Y.: Interestingness prediction by robust learning to rank. In: European conference on computer vision. pp. 488–503. Springer (2014)
12. Fu, Y., Hospedales, T.M., Xiang, T., Xiong, J., Gong, S., Wang, Y., Yao, Y.: Robust subjective visual property prediction from crowdsourced pairwise labels. IEEE transactions on pattern analysis and machine intelligence **38**(3), 563–577 (2015)
13. Gong, D., Liu, L., Le, V., Saha, B., Mansour, M.R., Venkatesh, S., Hengel, A.v.d.: Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1705–1714 (2019)
14. Goodfellow, I., Bengio, Y., Courville, A.: Deep learning. MIT press (2016)
15. Grabner, H., Nater, F., Druey, M., Van Gool, L.: Visual interestingness in image sequences. In: Proceedings of the 21st ACM international conference on Multimedia. pp. 1017–1026. ACM (2013)
16. Graves, A., Wayne, G., Danihelka, I.: Neural Turing machines. arXiv preprint arXiv:1410.5401 (2014)
17. Gygli, M., Soleymani, M.: Analyzing and predicting gif interestingness. In: Proceedings of the 24th ACM international conference on Multimedia. pp. 122–126. ACM (2016)
18. Hasan, M., Choi, J., Neumann, J., Roy-Chowdhury, A.K., Davis, L.S.: Learning temporal regularity in video sequences. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 733–742 (2016)

19. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural computation* **9**(8), 1735–1780 (1997)
20. Ito, Y., Kitani, K.M., Bagnell, J.A., Hebert, M.: Detecting interesting events using unsupervised density ratio estimation. In: *Proceedings of 3rd IEEE International Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Streams at ECCV2012*. pp. 151–161. Springer (2012)
21. Jiang, Y.G., Wang, Y., Feng, R., Xue, X., Zheng, Y., Yang, H.: Understanding and predicting interestingness of videos. In: *Twenty-Seventh AAAI Conference on Artificial Intelligence* (2013)
22. Kim, Y., Kim, M., Kim, G.: Memorization precedes generation: Learning unsupervised gans with memory networks. In: *the International Conference on Learning Representations (ICLR)* (2018)
23. Kramer, M.A.: Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal* **37**(2), 233–243 (1991)
24. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: *European conference on computer vision*. pp. 740–755. Springer (2014)
25. Liu, W., Luo, W., Lian, D., Gao, S.: Future frame prediction for anomaly detection—a new baseline. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 6536–6545 (2018)
26. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 3431–3440 (2015)
27. Luo, W., Liu, W., Gao, S.: A revisit of sparse coding based anomaly detection in stacked rnn framework. In: *Proceedings of the IEEE International Conference on Computer Vision*. pp. 341–349 (2017)
28. Oßwald, S., Bennewitz, M., Burgard, W., Stachniss, C.: Speeding-up robot exploration by exploiting background information. *IEEE Robotics and Automation Letters* **1**(2), 716–723 (2016)
29. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
30. Phillips, W.: On the distinction between sensory storage and short-term visual memory. *Perception & Psychophysics* **16**(2), 283–290 (1974)
31. Potter, M.C., Levy, E.I.: Recognition memory for a rapid sequence of pictures. *Journal of experimental psychology* **81**(1), 10 (1969)
32. Rumelhart, D.E., Hinton, G.E., Williams, R.J., et al.: Learning representations by back-propagating errors. *Cognitive modeling* **5**(3), 1 (1988)
33. Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., Lillicrap, T.: Meta-learning with memory-augmented neural networks. In: *International conference on machine learning*. pp. 1842–1850 (2016)
34. Shen, Y., Demarty, C.H., Duong, N.Q.: Deep learning for multimodal-based video interestingness prediction. In: *2017 IEEE International Conference on Multimedia and Expo (ICME)*. pp. 1003–1008. IEEE (2017)
35. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. In: *Intelnational Conference on Learning Research* (2015)
36. Wang, C.: Kernel learning for visual perception. Ph.D. thesis, Nanyang Technological University (2019)
37. Wang, C., Yang, J., Xie, L., Yuan, J.: Kervolutional neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 31–40 (2019)

38. Wang, C., Zhang, L., Xie, L., Yuan, J.: Kernel cross-correlator. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
39. Wang, S., Chen, S., Zhao, J., Jin, Q.: Video interestingness prediction based on ranking model. In: Proceedings of the Joint Workshop of the 4th Workshop on Affective Social Multimedia Computing and first Multi-Modal Affective Computing of Large-Scale Multimedia Data. pp. 55–61. ACM (2018)
40. Wang, W., Ahuja, A., Zhang, Y., Bonatti, R., Scherer, S.: Improved generalization of heading direction estimation for aerial filming using semi-supervised regression. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 5901–5907. IEEE (2019)
41. Zhang, P., Wang, D., Lu, H., Wang, H., Yin, B.: Learning uncertain convolutional features for accurate saliency detection. In: Proceedings of the IEEE International Conference on computer vision. pp. 212–221 (2017)
42. Zhao, B., Fei-Fei, L., Xing, E.P.: Online detection of unusual events in videos via dynamic sparse coding. In: CVPR 2011. pp. 3313–3320. IEEE (2011)
43. Zhao, Y., Deng, B., Shen, C., Liu, Y., Lu, H., Hua, X.S.: Spatio-temporal autoencoder for video anomaly detection. In: Proceedings of the 25th ACM international conference on Multimedia. pp. 1933–1941 (2017)