Traffic Risk Assessment: A Two-Stream Approach Using Dynamic-Attention

Corcoran Gary (Patrick), Clark James
Department of Electrical and Computer Engineering
McGill University
Montreal, Quebec
gary.corcoran@mail.mcgill.ca, clark@cim.mcgill.ca

***Abstract*—The problem being addressed in this research is performing traffic risk assessment on visual scenes captured via outward-facing dashcam videos. To perform risk assessment, a two-stream dynamic-attention recurrent convolutional neural architecture is used to provide a categorical risk level for each frame in a given input video sequence. The two-stream approach consists of a spatial stream, which analyses individual video frames and computes high-level appearance features and a temporal stream, which analyses optical flow between adjacent frames and computes high-level motion features. Both spatial and temporal streams are then fed into their respective recurrent neural networks (RNNs) that explicitly models the sequence of features in time. A dynamic-attention mechanism which allows the network to learn to focus on relevant objects in the visual scene is added. These objects are detected by a state-of-the-art object detector and correspond to vehicles, pedestrians, traffic signs, *etc*. The dynamic-attention mechanism not only improves classification performance, but it also provides a method to visualise what the network "sees" when predicting a risk level. This mechanism allows the network to implicitly learn to focus on hazardous objects in the visual scene. Additionally, this research introduces an offline and online model that differ slightly in their implementations. The offline model analyses the complete video sequence and scores a classification accuracy of 84.89%. The online model deals with an infinite stream of data and produces results in near real-time (7 frames-per-second); however, it suffers from a slight decrease in classification accuracy (79.90%).**

***Keywords*-Traffic risk assessment, dynamic-attention, recurrent neural network, two-stream, spatial stream, temporal stream, optical flow**

I. Introduction

Driving any vehicle can be a difficult task even for the most experienced drivers. This difficulty is shown by the high number of collisions in Canada alone. In 2015, there were $118,404$ collisions that were either fatal or involved a personal injury [2]. In about 84% of these accidents the cause could be traced back to driver error [7]. Although the majority of vehicles are currently equipped with passive safety systems, *i.e.* systems to help reduce the outcome of an accident such as seat belts, airbags, *etc.*, there are still a high number of serious incidents. Newer intelligent car models are becoming equipped with active safety systems that utilize an understanding of the vehicle's state to avoid and minimize the effects of a crash. Some of these systems include collision warning, adaptive cruise control, automatic braking, *etc*. Research into these active safety systems have expanded into applications that work with or for the driver. This new generation of advanced driver-assistance systems go beyond automated control systems by attempting to work in combination with the driver. These advanced safety systems include predicting driver intent [22], warning drivers of lane departures [15], *etc*. Additionally, one of the most recent trends in the automotive industry is the emergence of autonomous vehicles. In North America, especially in Canada, development and production of autonomous vehicles are growing at a rapid pace. These vehicles are integrated with active safety systems to enhance vehicle safety and reduce road accidents.

Active safety systems have many benefits; however, they are often difficult to implement as they require knowledge about the driver, the vehicle, and the environment. To address this problem, various research papers have attempted to gather information by utilizing multiple cameras, sensors, GPS locations, vehicle trajectories, and the list continues [10, 28, 18, 21]. Although many of these systems can provide a broad understanding of the driver and their surroundings, they require a difficult installation and calibration process. To this end, the proposed research aims to gather knowledge related to driver safety from a single outward-facing dashcam. To accomplish this, dashcam videos are processed by a two-stream dyanmic-attention recurrent convolutional architecture to produce a label corresponding to the perceived risk level in each visual scene. The risk levels are divided into four categories:

1) Low risk: visual scenes which do not include any hazards, resulting in a little-to-no probability of an incident (ideal driving situation).
2) Moderate risk: visual scenes which include hazards with a low-to-medium probability to cause an incident (normal driving situations).
3) High risk: visual scenes which include hazards with a high probability to cause an incident (unsafe driving situations).
4) Critical risk: visual scenes which include hazards with an almost certain probability to cause an incident (impending disaster).

Additionally, because of their novelty, self-driving vehi-

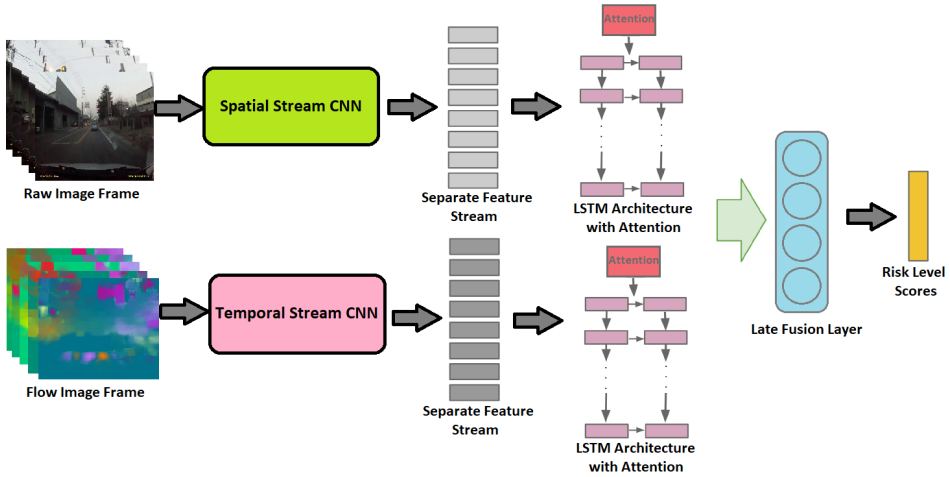

Figure 1: A depiction of the proposed two-stream dynamic-attention recurrent convolutional neural network. The raw image frame and the "flow image" are processed by separate CNN and LSTM architectures. The output of both LSTM architectures are combined via late fusion to provide a risk level.

cles and advanced driver-assistance system equipped vehicles have had little opportunity to learn from dangerous or at-risk traffic scenarios that provide drivers with a strong incentive to drive safely. The analysis of such incidents is an important step toward avoiding dangerous situations. The proposed method takes into consideration these at-risk scenarios, and any other dangerous traffic situations when producing a risk level. Using a dynamic-attention mechanism not only helps improve performance when categorizing at-risk scenarios, but it also provides a method to visualise what the network "sees" when calculating a risk level. This mechanism is shown to implicitly learn to identify hazardous objects in the visual scene. The complete model is shown in Figure 1.

The main contributions of this thesis are fourfold:

- Create a traffic risk assessment dataset that incorporates various driving situations and unsafe behaviours from a collection of outward-facing dashcam videos.
- Develop a two-stream dynamic-attention recurrent convolutional network that performs traffic risk assessment from a visual scene captured by an outward-facing dashcam.
- Demonstrate that using a dynamic-attention mechanism allows the network to implicitly learn to focus on hazardous objects in the visual scene when reasoning on the estimated risk level.
- Develop an offline model that demonstrates a high classification accuracy and an online model demonstrating the ability to run this algorithm on an infinite stream of data in real-time.

II. Related Works

Advanced driver-assistance systems (ADAS) are systems created in order to aid drivers in the driving process. When designed with a safe human-machine interface, the ideal system will increase car safety and more generally, road safety. Most road accidents occur due to human errors [7] and to mitigate these errors, advanced driver-assistance systems must automate, adapt, and enhance vehicle systems for safety and better driving. These features may include driver drowsiness detection, driver alertness, lane departure warning systems, alerting drivers to other road users or dangers, traffic warnings, *etc.* One popular field of research is modeling and monitoring drivers' attentiveness. Many of these works attempt to directly correlate driver attention to secondary measurements such as drowsiness, head movement and position, or alertness [10, 28, 18, 21]. [24] proposes a system using infrared bean sensors to measure eye closure, and in turn, attention levels. This system works by placing infrared bean sensors above the eye to detect eyelid positions. When the eyelids interrupt the bean, the system will measure the time that the bean was blocked and thus, providing eye closure measurements. [25] proposes a system using 3-D vision techniques to estimate and track the 3-D line of sight of a person. Their approach uses multiple cameras and multiple point light sources to estimate the line of sight without using user-dependent parameters. Several researchers have worked on head tracking [6] to mixed success. Similarly, [16] presents an approach that tracks the position of the head and estimates the respective head pose. It relies on 2-D template searching and a 3-D stereo matching. Other systems [27, 12] rely on measuring external car behaviors such as the vehicle's current distance to roadway lines. On the other end of the spectrum, [19] proposes a system to predict the driver's focus of attention. Their goal is to estimate what a person would pay attention to while driving. Their system uses a multi-branch deep neu-

ral architecture that integrates three sources of information: raw video, motion, and scene semantics. This architecture learns from a dataset consisting of driving scenes for which eye-tracking annotations are available.

Anticipating maneuvers can help prepare vehicles for unsafe road conditions and alert drivers if they intent on performing a dangerous maneuver. Maneuver anticipation complements existing ADAS by giving drivers more time to react and prepare for road situations, thereby providing an opportunity to prevent various accidents. Technologies such as lane keeping, blind spot check, *etc.*, have shown to be successful in alerting drivers when they commit a dangerous maneuver [8], however, there is still a need to detect these inappropriate maneuvers before they happen. Various methods attempt to anticipate driving maneuvers several seconds before they happen through sensory-rich environments [8, 17, 4]. These environments include information from multiple cameras, GPS locations, vehicle dynamics, *etc.* In particular, Jain *et al.* [8] demonstrate that a simple concatenation of multiple sensory streams does not fully capture the rich context for modeling maneuvers. They propose an Autoregression Input-Output Hidden Markov Model (AIO-HMM) which fuses sensory streams through a linear transformation of features. Based on the same intuitions, [9] propose a RNN-based architecture which learns rich representations for anticipation. Their approach learns how to optimally fuse information from different sensors. The architecture is trained in a sequence-to-sequence prediction manner such that it explicitly learns to anticipate given a partial context. More recently, the research proposed by [3] moves away from maneuver anticipation and attempts to anticipate accidents. The system is based on a dynamic-spatial attention recurrent neural network that learns to distribute attention to candidate objects dynamically while modeling temporal dependencies to robustly anticipate an accident.

III. The Dataset

The dataset being used in this research is a collection of outward-facing dashcam videos assembled by Chan *et al.* [3]. This dataset consists of 620 dashcam videos capture in six major cities of Taiwan. From these 620 videos, 1750 video clips were sampled where each snippet consists of 100 frames (five seconds). This dataset was originally used for accident prediction and thus, it was partitioned into two classes: 1) video snippets containing accidents and 2) video snippets containing no accidents. In total, there are 620 video snippets where the moment of accident occurs within the last thirty frames, and 1130 video snippets containing no accidents. One contribution of this research is to utilize this dataset to produce a new set of annotations corresponding to the level of risk in each visual scene viewed from the perspective of the driver's dashcam. To categorize the risk levels, a classic risk assessment approach is used.

Risk assessment is used to describe the overall process or method where you: 1) identify hazards and risk factors that have the potential to cause harm (hazard identification) and 2) analyse and evaluate the risk associated with these hazards (risk analysis or risk evaluation). Unfortunately, there is no simple or single way to determine the level of risk in a lot of situations, nor will a single technique apply to all circumstances. Determining risk requires the knowledge of domain-based activities, urgency of situations, and most importantly, objective judgment. Traffic risk assessment is a thorough look at various traffic situations in order to identify objects, situations, processes, *etc.*, that may cause harm to a vulnerable target, and to evaluate how likely and severe the risk is, *i.e* determining the risk level. The following risk levels are used to annotate the traffic risk dataset:

1) Low risk: visual scenes which do not include any hazards, resulting in a little-to-no probability of an incident (ideal driving situation).
2) Moderate risk: visual scenes which include hazards with a low-to-medium probability to cause an incident (normal driving situations).
3) High risk: visual scenes which include hazards with a high probability to cause an incident (unsafe driving situations).
4) Critical risk: visual scenes which include hazards with an almost certain probability to cause an incident (impending disaster).

The risk assessment dataset uses a dense set of annotations for each video snippet in the dataset, *i.e.* a risk level for every frame in the video. To create this dataset, each video snippet was displayed to a group of three subjects in segments consisting of ten frames (at 20 frames-per-second). The users were asked to annotated the short, half-second segment with a risk level. The annotated risk level is then used for all frames in the current segment. This process was completed for all segments in the video snippet and for all video snippets in the dataset. To ease the task for the annotators, the following set of criteria was used as a guideline for each risk level:

1) Low risk situations occur when there are no vehicles within a close proximity to one another in the visual scene and/or all vehicles in the visual scene are currently stopped. The visual scene must also contain no pedestrians and perfect weather conditions.
2) Moderate risk situations occur when normal driving conditions are present. These conditions can be described as visual scenes where there is no unsafe driving behaviour, *i.e.* all vehicles and pedestrians are following the traffic rules and regulations. The visual scene must also depict good weather conditions and a constant flow of traffic.
3) High risk situations occur when there are unsafe driving conditions. Examples of this include pedestrians

Figure 2: Sampled video sequences demonstrating the various risk levels associated with each visual scene. All labels were annotated by a collection of human viewers

on the road, speeding vehicles, drivers not following traffic rules and regulations, *etc*. High risk situations also occur when there is very high traffic congestion or poor weather conditions.

4) Critical risk situations occur when it is almost certain an accident will occur, *i.e.* two vehicles heading towards one another without providing any cue as to stopping.

Because the annotation task is empirical and based on the observer's discretion, some video snippets are difficult to annotate into one category or may fall loosely into two or three categories. Therefore, video snippets containing a high standard deviation averaged across all annotated segments were removed. Generally, these video snippets correspond to situations that are not well-characterized in terms of the defined risk levels or may fall into multiple categories. This filtering process removed 347 video snippets. The dense labels used for the remaining 1403 video snippets were calculated from the average score across the three sets of annotations. Figure 2 demonstrates sampled video frames depicting each risk level while Figure 3 shows the probability distribution across the four risk levels in the newly annotated dataset.

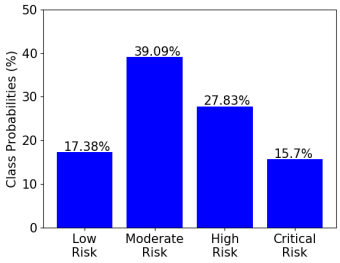

Figure 3: Class probabilities for the proposed traffic risk assessment dataset.

IV. Implementation

Given an input video sequence, the goal of this algorithm is to produce a categorical label at every timestep corresponding to the level of visual risk associated with the current frame. To achieve this goal, a two-stream dynamic-attention recurrent convolutional neural architecture (RCNN)

is used. The first stream, *i.e.* the spatial RCNN, analyses raw video frames and computes high-level appearance features, while the second stream, *i.e.* the temporal RCNN, analyses optical flow information between adjacent frames and computes high-level motion features. Both streams are then fed into separate dynamic-attention LSTM architectures that model both streams as hidden units in time. The additional attention mechanism allows the LSTM network to focus on relevant parts of each frame, thus providing a method to visualise what the network "sees". The output of both LSTM architectures are then combined at each timestep via late fusion to produce the corresponding risk level. The complete model is demonstrated in Figure 1. This research explores two approaches to the problem of traffic risk assessment, *i.e.* an online and offline method. The differences in both approaches are discussed below and will be highlighted in the following sections.

1) Offline: In offline video frame classification the full video sequence, *i.e.* the 100 frame sequence, is used as input into the two-stream network which allows all information in the video to be available at once. This process is beneficial as it permits the network to look forward and backward in time, throughout the full video sequence, in order to make accurate predictions at each timestep. Due to the network being used in an offline fashion, run time is not a constraint allowing the architecture to use deeper neural networks with a higher number of parameters.

2) Online: In online video frame classification, the algorithm must decide at every moment what the correct output label is. This decision cannot use information from future frames as this information is not yet available. The model is also required to deal with input data of an infinite length. To tackle this problem, it was decided to use a sliding window approach. This method was chosen as it is intuitive to understand and very easy to implement and test. A sliding window size of ten was justified through experimentation. The online model reads a series of ten sequential frames and produces an output label for the last timestep. The sliding window is then moved through time, with a stride of one, until the video sequence terminates. Because this model works in an online manner, the algorithm is required to run in real-time. This requirement constrains the system to use more shallow networks with fewer parameters compared to the offline implementation.

The complete algorithm is divided into five main stages: 1) the spatial stream RCNN, 2) the temporal stream RCNN, 3) the LSTM architecture, 4) the dynamic-attention mechanism, and 5) the training procedure. The following sections will describe each stage in greater detail.

A. Spatial Stream Recurrent Convolution Neural Network

The spatial stream RCNN operates on individual raw video frames, effectively performing object recognition from a still image, and outputs a sequence of high-level appearance features. Object recognition is an important task for traffic risk assessment as it provides information on surrounding vehicles, pedestrians, traffic signs, *etc*. The appearance features are computed using a pre-trained CNN that extracts a fixed 4096-dimensional feature vector from each video sequence (this vector sized was justified through experimentation). Because this spatial stream is essentially an image classification architecture, it was pre-trained on a large image classification dataset, *i.e.* the ImageNet [14] dataset. The spatial stream takes input images of size 224×224.

The offline model utilizes a VGG network [26] consisting of 19 weight layers, *i.e.* convolutional and fully-connected layers. This model was found to provide the best results. Between each convolutional layer there is a combination of batch normalization and rectified linear units (ReLU). Batch normalization was added to the model as it was found to aid in the stability of the network. Each sequence of convolution, batch normalization, and ReLU ends with a max-pooling layer. At the end of the network there are three fully-connected layers with a combination of ReLU and dropout. Using dropout was found to help improve generalization and was set to 50% during training. To compute the final appearance features, the last two fully-connected layers were removed from the network. Through testing, this process was shown to help improve generalization and thus, increase performance. The resultant spatial stream has a dimensionality of 100×4096.

The online model utilizes an AlexNet [13] consisting of 8 weight layers. This model provided the best speed/accuracy trade-off and was justified through testing. Unlike the VGG network, the AlexNet does not include batch normalization between each convolutional layer. Batch normalization was removed as this process is computationally expensive and slow to run. Similar to the VGG network, a dropout of 50% was used during training and the last two fully-connected layers were removed. The resultant spatial stream has a dimensionality of 10×4096 (the dimension of 10 is a result of the sliding window size).

B. Temporal Stream Recurrent Convolutional Neural Network

Temporal information encodes the pattern of apparent motion of various objects in a visual scene and is a crucial component to any video classification problem. For the problem of traffic risk assessment, gathering knowledge about temporal features can help provide important information such as the relative speed or trajectory of a driver and surrounding vehicles. The motion estimation used in this approach is based on a dense optical flow field. This dense flow field computes a flow vector for all points in a pair of frames using the algorithm of [5]. The flow algorithm of [5] was used as there is an open-source implementation,

and harnessing the power of a GPU, the algorithm can run faster than real-time.

To compute flow fields, motion estimates were computed between each pair of consecutive frames in the video sequence. The resultant flow fields consist of two channels corresponding to flow vectors in both the x and y directions. During testing, it was shown that pre-training on a larger image classification dataset, *i.e.* the ImageNet dataset, provided the network with a strong initialization to facilitate faster training and prevent overfitting on the relatively small risk dataset. To utilize the exiting network pre-trained on raw video frames, the flow fields were transformed into "flow images" by centering the x and y values around 128 and multiplying by a scalar such that the flow values fall between the range of 0 and 255. A third channel for the flow image was created by calculating the flow magnitude. Similar to the spatial stream, the "flow images" were resized to 224×224.

To compute high-level motion features from the input "flow image", the offline model uses a pre-trained VGG network. The VGG network used is smaller than the spatial CNN as it only consists of 16 weighted layers. The final motion features are extracted by removing the last two fully-connected layers of the network. After processing the complete sequence of "flow images", the resultant temporal stream has a dimensionality of 99×4096. The online architecture utilizes the same pre-trained AlexNet model as the spatial stream. The resultant online temporal stream has a dimensionality of 9×4096.

C. Long Short-Term Memory Architecture

The complete two-stream RCNN architecture uses an RNN to explicitly model sequences of CNN activations in time. Since most videos contain dynamic content, the variations between frames encode additional information that is useful in making more accurate predictions. A multilayer LSTM architecture is used in which the output from the previous LSTM layer is used as input to the next layer.

The offline model uses a bidirectional RNN (BRNN) [23]. BRNNs are beneficial as they increase the amount of input information available to the network by including future and past input states. The online model uses a standard unidirectional RNN. As previously mentioned, the final output of the offline model's spatial and temporal streams have a dimensionality of 100×4096 and 99×4096, respectively, while the output of online model has a dimensionality of 10×4096 and 9×4096. It is important that both output streams have the same sequence length, and thus, the first sequence of appearance features are dropped from both the offline and online spatial streams resulting in a dimension of 99×4096 and 9×4096 respectively. These streams are then fed into their respective LSTM architectures. The offline spatial stream RCNN consists of two hidden layers each containing 512 hidden cells, while the offline temporal stream RCNN contains the same number of layers but with 256 hidden cells. The online spatial and temporal stream RCNNs both consist of one layer and 512 hidden cells. These parameters were selected based on the highest validation classification accuracy.

D. Dynamic-Attention Mechanism

The dynamic-attention mechanism allows the network to selectively focus on parts of an image when performing image classification. The attention model learns which parts in the image are relevant for the task at hand and attaches a higher attention to them. This process is important as it not only improves classification accuracy and allows for a faster training process, it also adds a method to visualise where the network "looks" when performing a particular task. In the case of traffic risk assessment, this attention mechanism adds an extra dimension of interpretability as it implicitly learns to focus on hazardous objects in the visual scene when an at-risk scenario is present. The approach used is similar to the method previously described in [3].

To learn the dynamic-attention model, candidate objects corresponding to vehicles, pedestrians, traffic signs, *etc.*, at specific spatial locations are extracted from each frame. To extract objects from each frame, a state-of-the-art, real-time object detection system proposed by [20] is used. This object detector extracts the class, spatial location, and probability score for each detected object. A dynamic-attention mechanism is used for both the spatial and temporal streams. From each input image frame, ten of the highest-scoring objects are extracted and concatenated with the full image frame to use as input into the dynamic-attention model. For the spatial and temporal streams, each object is extracted from either the raw image frame or the "flow image" frame. In situations where there are less than ten candidate objects, the remaining images are filled with zeros. The zero-filled object images were found to produce attention scores approximately equal to zero, corresponding to the network ignoring these objects.

To provide a dynamic-attention level for each of the ten candidate objects, each object is first resized to 224×224 and passed through their respective CNN architectures to produce a high-level observation $\hat{x}_t^j$. This process was also repeated for the full image frame and resulted in an output matrix with a dimensionality of $99 \times 11 \times 4096$ and $9 \times 11 \times 4096$ for the offline and online models respectively. This matrix is then used as input into the dynamic-attention model. On a per-frame basis, each object feature is given an attention level based on the soft-attention method proposed by [1]. The attention model consists of a single fully-connected layer, U, that transforms each high-level feature $\hat{x}_t^j$ into the same feature-space as h_t. The matrix U is jointly learned with the full two-stream architecture. Instead of using a complete feed-forward neural network to model the relevance between the previous hidden state h_{t-1} and each transformed observation $U\hat{x}_t^j$, as proposed by [3], a simple

Table I: Average classification test accuracy for the offline two-stream model. The spatial stream and temporal stream utilize raw image frames and "flow images" respectively.

Spatial Stream		**Temporal Stream**		**Two-Stream**	
Single-Frame CNN	59.06%	Single-Frame CNN	61.65%	Two-Stream without attention	78.45%
3D-CNN [11]	69.42%	3D-CNN [11]	67.77%		
Spatial RCNN without attention	74.29%	Temporal RCNN without attention	76.60%	Two-Stream with attention	**84.89%**
Spatial RCNN with attention	**78.91%**	Temporal RCNN with attention	**82.11%**		

weighted dot-product proved to yield better results:

$$e_t^j = h_{t-1} U \hat{x}_t^j \tag{1}$$

where t and j represent each timestep and object, respectively. U is a matrix of weights corresponding to the attention model parameter, and e_t^j are the unnormalized attention scores. These weights are then normalized using a softmax function to produce an output on the scale of zero to one, thus, each observation is provided with a normalized attention score stating how relevant that particular observation is when predicting the risk level.

V. Results

The following section will discuss and evaluate the results on the traffic risk assessment dataset for both the offline and online model. A training, validation, and testing split of 70%, 10%, and 20% was used, resulting in 982, 281, and 140 videos respectively.

A. Offline

To evaluate the offline model, each stream was individually evaluated with and without the use of dynamic-attention. In addition, a baseline single-frame CNN and a 3D-CNN [11] classifier were used for comparison. These classifiers were adjusted to provide the best classification accuracy for each stream. Table I demonstrates the results. As expected, the single-frame CNN classifier performed very poorly in both the spatial and temporal streams (59.06% and 61.65% respectively) as it does not take into consideration the temporal changes of a given feature stream. The 3D-CNN performed slightly better (69.42% and 67.77%) as this approach attempts to implicitly model temporal changes in fixed-size clips containing small portions of the video. Unfortunately, this method is difficult to train and takes considerable time due to the high number of parameters involved. The spatial and temporal stream RCNNs greatly outperformed (74.29% and 76.60% respectively) both aforementioned methods as these approaches explicitly model the temporal changes of features as hidden units in time. Adding the dynamic-attention model was found to improve these results greatly. It is worth noting that the dynamic-attention temporal stream (82.11%) significantly outperformed the dynamic-attention spatial stream (78.91%). This process demonstrates the important of paying particular attention to various objects' speed, trajectory, and other information captured via motion features when categorizing risk levels. The complete two-stream model was then tested without (78.45%) and with (84.89%) the use of dynamic-attention. The two separate streams were combined via a simple average across class scores as this method was found to provide the best results. Additionally, this process demonstrates how each stream contributes its own information to the task of traffic risk assessment and combining these streams allowed the network to capture an overview of both streams of information. Table II displays the confusion matrix for the complete offline two-stream with dynamic-attention while Figure 4 demonstrates a sampled output. The low accuracy of the low risk class is a result of falsely detected objects in low risk scenes (which happens frequently). The categories on the right display the output risk level (low risk: beige, moderate risk: yellow, high risk: orange, and critical risk: red) and the plot on the bottom window displays the probability of each class as a function of time (the same colour-scheme is used). In addition to the risk level, each frame contains bounding boxes locating various detected objects. Each bounding box consists of a colour to represent the level of attention that particular object receives. The attention scale is demonstrated on the left-side of the each figure. To display the dynamic-attention from both streams, a simple average across attention weights was computed. It is worth noting in Figures 4c- 4d, the dynamic-attention model is focusing its attention on the hazardous object in each visual scene.

Table II: Classification confusion matrix for the offline two-stream with dynamic-attention.

	Low Risk	**Moderate Risk**	**High Risk**	**Critical Risk**
Low Risk	$\mathbf{77.48}\%$	20.69%	1.83%	0%
Moderate Risk	8.42%	$\mathbf{91.03}\%$	0.55%	0%
High Risk	1.32%	9.65%	$\mathbf{82.39}\%$	6.64%
Critical Risk	0.04%	5.63%	12.04%	$\mathbf{82.29}\%$

B. Online

For online use, one of the main requirements is for the model to run in real-time. To address this constraint, a sliding window approach was used. Various window sizes were tested for each stream without the use of dynamic-attention. It was found that a sliding window of size 10 provided

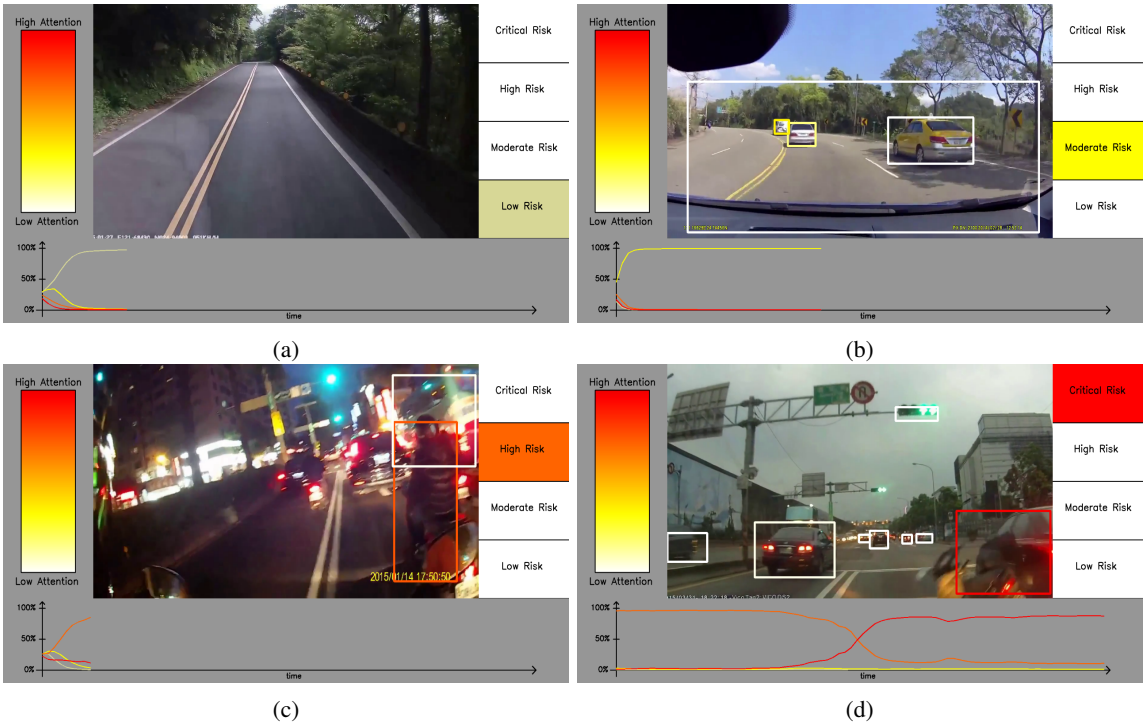

Figure 4: Output images displaying the results from the offline two-stream model.

the best trade-off between speed and accuracy. Doubling the window size to 20 increased the computation time two-fold while only increasing the accuracy by less than one percent. Using the window size of 10, the spatial stream RCNN received an accuracy of 69.94% while the temporal stream RCNN scored an accuracy of 71.90%. Using the dynamic-attention mechanism improved these scores to 73.28% and 75.66% respectively. The complete online two-stream with dyanmic-attention was then evaluated producing an overall classification accuracy of 79.90%. Although the online model suffers from a lower accuracy compared to the offline method, it provides the main benefit of running in real-time on an infinite stream of data. It is worth noting that each individual stream stream runs at approximately 13 frames-per-second (FPS) on a TITAN X GPU. The complete two-stream model runs at approximately 7 FPS. Table III displays the confusion matrix for the complete online two-stream with dynamic-attention. The online model's main performance loss is a result of misclassifying high and critical risk levels. It was found that the smaller window size and the removal of the bidirectional RNN primarily affected reasoning on these two risk classes.

Table III: Classification confusion matrix of the online two-stream with dynamic-attention.

	Low Risk	**Moderate Risk**	**High Risk**	**Critical Risk**
Low Risk	**78.17%**	18.0%	3.83%	0%
Moderate Risk	7.37%	**90.83%**	0.60%	1.20%
High Risk	8.93%	15.59%	**71.25%**	4.24%
Critical Risk	14.08%	11.96%	3.63%	**70.33%**

VI. Conclusion

This research presents the problem of traffic risk assessment from outward-facing dashcam videos. The proposed algorithm uses a two-stream dynamic-attention recurrent convolutional neural network to extract a set of high-level appearance and motion features in two separate streams. An LSTM architecture is used to model each separate stream in time. The addition of a dynamic-attention mechanism allows the model to learn to focus on relevant objects in the visual scene. The results show that these objects generally correspond to hazardous objects, thus demonstrating the network's ability to implicitly learn to identify hazardous behaviour. The final output of the algorithm is a risk level for each frame of an input video stream. The four risk levels are as follows: low risk, moderate risk, high risk, and critical risk. Both an offline and online model are proposed where the offline demonstrates a high classification accuracy across all risk classes, while the online model demonstrates a lower accuracy but runs on an infinite stream of data in near real-time (7 FPS).

Acknowledgment

This work is supported by NSERC. Additionally, we gratefull acknowledge Brisk Synergies for providing funding for this research.

References

[1] D. Bahdanau, K. Cho, and Y. Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". In: *CoRR* abs/1409.0473 (2014). arXiv: 1409.0473. URL: http://arxiv.org/abs/1409.0473.

[2] *Canadian Motor Vehicle Traffic Collision Statistics: 2015*. https://www.tc.gc.ca/eng/motorvehiclesafety/tp-tp3322-2015-1487.html. Accessed: 2018-02-08. 2017.

[3] F.-H. Chan et al. "Anticipating Accidents in Dashcam Videos". In: *Computer Vision – ACCV 2016*. Ed. by S.-H. Lai et al. Cham: Springer International Publishing, 2017, pp. 136–153. ISBN: 978-3-319-54190-7.

[4] A. Doshi, B. Morris, and M. Trivedi. "On-road prediction of driver's intent with multimodal sensory cues". In: *IEEE Pervasive Computing* 10.3 (July 2011), pp. 22–34. ISSN: 1536-1268. DOI: 10.1109/MPRV.2011.38.

[5] G. Farnebäck. "Two-Frame Motion Estimation Based on Polynomial Expansion". In: *Image Analysis*. Ed. by J. Bigun and T. Gustavsson. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 363–370. ISBN: 978-3-540-45103-7.

[6] A. Gee and R. Cipolla. "Determining the gaze of faces in images". In: *Image and Vision Computing* 12.10 (1994), pp. 639–647. ISSN: 0262-8856. DOI: https://doi.org/10.1016/0262-8856(94)90039-6. URL: http://www.sciencedirect.com/science/article/pii/0262885694900396.

[7] K. Haring, M. Ragni, and L. Konieczny. "A Cognitive Model of Drivers Attention". In: *Proceedings of the 11th International Conference on Cognitive Modeling, ICCM 2012*. Jan. 2012.

[8] A. Jain et al. "Know Before You Do: Anticipating Maneuvers via Learning Temporal Driving Models". In: *CoRR* abs/1504.02789 (2015). arXiv: 1504.02789. URL: http://arxiv.org/abs/1504.02789.

[9] A. Jain et al. "Recurrent Neural Networks for Driver Activity Anticipation via Sensory-Fusion Architecture". In: *CoRR* abs/1509.05016 (2015). arXiv: 1509.05016. URL: http://arxiv.org/abs/1509.05016.

[10] Q. Ji and X. Yang. "Real Time Visual Cues Extraction for Monitoring Driver Vigilance". In: *Computer Vision Systems*. Ed. by B. Schiele G. and Sagerer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 107–124. ISBN: 978-3-540-48222-2.

[11] S. Ji et al. "3D Convolutional Neural Networks for Human Action Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.1 (Jan. 2013), pp. 221–231. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2012.59.

[12] D. J. King, G. Siegmund, and D. T. Montgomery. "Outfitting a Freightliner Tractor for Measuring Driver Fatigue and Vehicle Kinematics During Closed-Track Testing". In: *SAE 942326*. Nov. 1994.

[13] A. Krizhevsky. "One weird trick for parallelizing convolutional neural networks". In: *CoRR* abs/1404.5997 (2014). arXiv: 1404.5997. URL: http://arxiv.org/abs/1404.5997.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*. NIPS'12. Lake Tahoe, Nevada: Curran Associates Inc., 2012, pp. 1097–1105. URL: http://dl.acm.org/citation.cfm?id=2999134.2999257.

[15] W. Kwon and S. Lee. "Performance evaluation of decision making strategies for an embedded lane departure warning system". In: *Journal of Robotic Systems* 19.10 (2002), pp. 499–509. ISSN: 1097-4563. DOI: 10.1002/rob.10056. URL: http://dx.doi.org/10.1002/rob.10056.

[16] Y. Matsumoto and A. Zelinsky. "An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement". In: *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*. 2000, pp. 499–504. DOI: 10.1109/AFGR.2000.840680.

[17] B. Morris, A. Doshi, and M. Trivedi. "Lane change intent prediction for driver assistance: On-road design and evaluation". In: *2011 IEEE Intelligent Vehicles Symposium (IV)*. June 2011, pp. 895–901. DOI: 10.1109/IVS.2011.5940538.

[18] R. Onken. "DAISY, an adaptive, knowledge-based driver monitoring and warning system". In: *Intelligent Vehicles '94 Symposium, Proceedings of the*. Oct. 1994, pp. 544–549. DOI: 10.1109/IVS.1994.639576.

[19] A. Palazzi et al. "Predicting the Driver's Focus of Attention: the DR(eye)VE Project". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2018).

[20] J. Redmon and A. Farhadi. "YOLOv3: An Incremental Improvement". In: *CoRR* abs/1804.02767 (2018). arXiv: 1804.02767. URL: http://arxiv.org/abs/1804.02767.

[21] A. Sahayadhas and K. Sundaraj. "Detecting Driver Drowsiness Based on Sensors: A Review". In: *Sensors* 12.12 (2012), pp. 16937–16953. ISSN: 1424-8220. DOI: 10.3390/s121216937. URL: http://www.mdpi.com/1424-8220/12/12/16937.

[22] D. D. Salvucci. "Inferring Driver Intent: A Case Study in Lane-Change Detection". In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 48.19 (2004), pp. 2228–2231. DOI: 10.1177/154193120404801905. eprint: https://doi.org/10.

1177/154193120404801905. URL: https://doi.org/10.1177/154193120404801905.

[23] M. Schuster and K. K. Paliwal. “Bidirectional Recurrent Neural Networks”. In: *Trans. Sig. Proc.* 45.11 (Nov. 1997), pp. 2673–2681. ISSN: 1053-587X. DOI: 10.1109/78.650093. URL: http://dx.doi.org/10.1109/78.650093.

[24] T. Selker, A. Lockerd, and J. Martinez. “Eye-R, a Glasses-mounted Eye Motion Detection Interface”. In: *CHI ’01 Extended Abstracts on Human Factors in Computing Systems*. CHI EA ’01. Seattle, Washington: ACM, 2001, pp. 179–180. ISBN: 1-58113-340-5. DOI: 10.1145/634067.634176. URL: http://doi.acm.org/10.1145/634067.634176.

[25] S.-W. Shih, Y.-T. Wu, and J. Liu. “A calibration-free gaze tracking technique”. In: *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*. Vol. 4. 2000, 201–204 vol.4. DOI: 10.1109/ICPR.2000.902895.

[26] K. Simonyan and A. Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *CoRR* abs/1409.1556 (2014). arXiv: 1409.1556. URL: http://arxiv.org/abs/1409.1556.

[27] A. Suzuki et al. “Lane recognition system for guiding of autonomous vehicle”. In: *Proceedings of the Intelligent Vehicles ‘92 Symposium*. June 1992, pp. 196–201. DOI: 10.1109/IVS.1992.252256.

[28] J.-D. Wu and T.-R. Chen. “Development of a drowsiness warning system based on the fuzzy logic images analysis”. In: *Expert Systems with Applications* 34.2 (2008), pp. 1556–1561. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2007.01.019. URL: http://www.sciencedirect.com/science/article/pii/S0957417407000401.