NRC·CNRC CONSTRUCTION

Development of a Hybrid Algorithm to predict room fire flashovers based on Vision data

Author(s): Yuchuan Li and Yoon J. Ko, Ph.D
Report No.: A1-020368.1
Report Date: Aug. 13th, 2021
Project No.: A1-020368

National Research Council Canada
Conseil national de recherches Canada

Canada

© 2021 Her Majesty the Queen in Right of Canada,
as represented by the National Research Council Canada.

PDF: Cat. No. NR24-97/2021E-PDF
ISBN 978-0-660-40138-6

Development of a Hybrid Algorithm to Predict Room Fire Flashovers based on Vision data

PM

Ko, Yoon Digitally signed by Ko, Yoon
Date: 2021.09.03 16:54:57 -04'00'

Yoon J. Ko, Ph.D,

Approved

Kashef, AH
2021.09.13 16:31:48 -04'00'

Ahmed Kashef, Ph.D, P.Eng.
Program Leader
Research & Development, Fire Safety
NRC Construction

Report No: A1-020368.1
Report Date: 13, August, 2021
Project No: A1-020368
Program: Research & Development, Fire Safety

Copy no. 1 of 1

This report may not be reproduced in whole or in part without the written consent of the National Research Council Canada and the Client.

Table of Contents

List of Figures

List of Tables

Development of a Hybrid Algorithm to predict room fire flashovers based on Vision data

Yuchuan Li and Yoon Ko, Ph.D

Executive Summary

One of the most deadly situations that firefighters could face in firefighting is flashover, which is sudden fire propagation occurring in a room with all the items in the room bursting into the fire. In general, firefighters need years of training to identify and predict flashovers. Although the decades of experimental and numerical fire research shed light on the room fire dynamics, there are still gaps in transferring the fire science to the fire ground where innovative, yet simple solutions are needed to overcome the harsh environment.

This project is to develop a robust smart firefighting tool that can be easily deployed like cameras to the fire ground and provide effective assistance to firefighters. One critical ability of the smart fighting tool would be assisting firefighters in detecting impending deadly flashovers. An explorative study is conducted adopting deep learning methods in the processing of smoke and flame video images. Scientific knowledge of room fires is also coupled to build an algorithm that requires less hardware but produces high accuracy. The hybrid system combining deep learning methods and fire safety knowledge only requires RGB vision data for flashover prediction, which can be acquired by any camera used by firefighters. The system converts the RGB inputs to thermal images and processes the flashover analysis with images classified as smoke and flame.

The system was tested with video data obtained from various fire tests, and the performance was evaluated and compared with other existing models. The hybrid algorithm of the flashover prediction system demonstrated promising performance by surpassing other existing methods designed for similar tasks, with high prediction accuracy.

1 Introduction

A fire may cause a catastrophic impact on the economics due to the potential property loss/damage and, more importantly, on human life and safety. A report [1] shows that the total number of direct property losses due to fires is over $25 billion, and 1,318,500 fire cases were reported in 2018 in the US. In addition, home fires alone have caused 2,720 civilian fire deaths and 15,200 civilian fire injuries. Also, more than 30,000 firefighters are injured each year during firefighting operations [2,3].

One of the most deadly situations that firefighters could face in firefighting is flashover, which is sudden fire propagation occurring in a room with all the items in the room bursting into the fire, as shown in **Figure 1**. The risk of flashover comes mainly because it occurs very rapidly, and predicting its onset is very difficult in general. Besides, a compartment fire develops rapidly since modern furniture contains highly combustible/flammable materials, which tend to burn rapidly and release high heats [4], such as chemical fiber/plastic products.

a) b)

Figure 1: Examples of flashover. a) A flashover captured on GoPro, from [5]. b) Fire development of ceiling layer that shows flashover happening, from [6].

Researchers have poured much effort into room fire research to understand the phenomenon of flashover and the risk associated with it. The decades of experimental and modeling research have elucidated that there are typical indicators for flashovers, such as the smoke layer temperature in a typical room reaching approximately 550 °C to 600 °C [7], and the heat flux on the floor reaching 20 kW/m^2 to 25 kW/m^2 [8]. In addition, there are also modeling tools to simulate a room fire and predict the onset of flashover.

However, judgments for those indicators require measured thermal data captured by sensors in fire scenes, which are often difficult to obtain due to the harsh environment and lack of reliable sensors. Thus, firefighters are trained to identify and predict flashover based mainly on visual signs such as thick dark smoke build-ups near the ceiling, intense heat, active flame rolling across the ceiling, and smoldering of combustibles in a room. However, it is easy to miss these visual signs since a flashover occurs rapidly, and the visibility in a fire room is often very low due to the dark and smoke-filled environment. In addition, IR cameras have become prevalent tools to assist firefighters in locating hot spots or a point of egress. However, it is questionable whether the common firefighting IR cameras effectively assist accurate flashover prediction. Furthermore, accurate interpretation of the information received from these IR cameras is challenging because the thermal images and temperature readings vary depending on the different measurement wavelengths, temperature sensitivities, and configurations [9,10]. These problems cannot be easily solved by regular training of firefighters often conducted in simulated room fires.

Predicting a flashover under such harsh extreme conditions would be significantly supported by employing a smart tool for the automatic processing of visual information. With the advancement in image processing techniques, images of smoke and flame can be effectively analyzed using neural networks and deep learning methods. Recently, these techniques have been used in vision-based fire flame and smoke detection systems (VFSDS), which overcome the limitations of conventional spot-type smoke and flame detection systems. These

techniques can be applied to the analysis of visual signs of flashover, and when combined with Machine Learning (ML) and Artificial Intelligent (AI) technologies, predicting flashover would also be possible.

Therefore, the long-term objective of this project is to develop a robust smart firefighting tool that can be easily deployed to the fire ground and provide effective assistance to firefighters in actual building fire scenarios. One key ability of the smart fighting tool would be assisting firefighters in detecting impending deadly flashovers.

This report describes the preliminary study conducted to explore the feasibility of adopting Machine Learning (ML) and Artificial Intelligent (AI) technologies in developing a tool to assist firefighters in predicting impeding flashovers. Unlike previous studies focusing on point measurements (e.g., temperature and heat flux), this project explores the potential benefits of analyzing visual data (vison and thermal data) to observe incremental development of the smoke layer and the associated temperature rises in the room. The report discusses a hybrid algorithm employing both ML and pre-informed fire science knowledge, such as the flashover criteria of temperature and heat flux, and for validation, in-house archived image data are mainly used.

1.1 Objectives

The objectives of this explorative project are the following:

- To conduct a literature review to explore the feasibility of adopting ML and AL in predicting flashover based on image data.
- To design and develop a hybrid algorithm based on ML, which is suitable for processing fire/smoke vision data and thermal data.
- To test the hybrid algorithm and evaluate the performance

2 Literature Review

2.1 Flashover Prediction

Flashover is a complicated fire phenomenon observed in a compartment and is defined as a near-simultaneous ignition of all the combustible materials in the enclosure. Most materials undergo thermal decomposition when heated and release fuel vapors, and they burn while spreading out. Flashover would happen in a typical room when the hot smoke layer near the ceiling heats the exposed surface of furniture or items in the room to specific temperatures, which are also called autoignition temperatures. In general, the upper smoke layer in the room reaches the temperature around 500°C to 600°C, a flashover occurs with simultaneous ignition of all the items in the room. At the onset of the flashover, the heat flux measured on the floor reaches 20 kW/m^2 [11]. A typical example of flashover in fire experiments is shown in **Figure 2**. The most dangerous part of flashover is that it could spread rapidly, so the entire room would burst into fire. Consequently, it could kill the occupants and firefighters; and block the exit path for escape.

Figure 2: An example of flashover in experiments, from [11].

Room fires have been studied rigorously with mathematical models to predict parameters of temperature and heat release rate (HRR) as well as the onset of the flashover. One of the earliest technical definitions and analyses of flashover dates back to the 1970s [12]. Barbrausks et al. [13] introduced a mathematical model using regression to analyze the relationship between HRR, ventilation factor, and flashover. It discussed HRR and ventilation factors for different scenarios. In addition to experimental works, room fire modeling has helped to improve the understanding of flashover. Beshir et al. combined the benefit of simulations and actual tests and suggested semi-imperial flashover prediction models [14]. They used Fire Dynamic Simulator (FDS) developed by NIST (the National Institute of Standards and Technology) as their simulation engine and conducted sensitivity and parametric studies to understand the heat balance for under-ventilated compartments.

An example of temperature development for traditional compartment fire is shown in **Figure 3**. Once ignited, a room fire grows with the increase in the room temperature, and the growth rate depends on the combustibility/flammability of the fuels/items in the room. At the onset of flashover, the whole room will be involved in the fire, and the HRR is generally dependent on the amount of air supply through the windows or doors. With the parameters of temperature and HRR, a flashover could be forecasted.

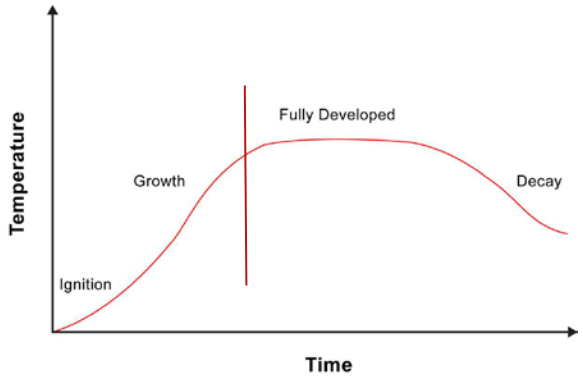


Figure 3: An example of temperature development for traditional compartment fire, from [11].

Experimental and numerical studies conducted for decades to understand flashovers found criteria to determine the onset of flashovers, such as smoke layer temperature and heat flux on the floor. Using these criteria, an algorithm and a device prototype were developed for flashover prediction in 2010 [15]. Their attempts to measure smoke layer temperatures (by thermocouples, multispectral IR optical sensors, and radiometric sensors) were partially successful due to the intense heat, so the heat received by a sentinel on the floor was used instead in the prediction. The effectiveness of the predictor in real scenarios was not thoroughly tested in the study. The major limitation of their predictor was relying on single-point measurement of the temperature of a sentinel since the radiative heat received by the sentinel from the smoke layer and fire varies significantly by its location on the floor.

2.1.1 Flashover Prediction with Machine Learning

Machine Learning (ML) methods are widely used in flashover prediction with classification tasks. A study proposed in [16] presented an ML approach for predicting flashover in a compartment fire. They used a traditional machine learning method: Support Vector Machine (SVM) as their core method and built a prediction model taking temperature and HRR as input. Their works provided flashover prediction based on a combination of HRR and temperature data. Another study proposed by [17] introduced an ML algorithm to predict flashover onset in archival experiments in a 1/5-scaled ISO enclosure. It used lasso regression to significantly reduce the amount of variance with a negligible increase in bias. The decision boundary is shown in **Figure 4** below. x_1 and x_2 are two input features from different dimensions of fire. y is the possibility of flashover occurrence (i.e., $y = 1$ for 100% possibility and $y = 0$ for 0% possibility). 'x' marks the test samples that flashover happened and 'o' marks the test samples that flashover never happened. Moreover, their algorithm showed a remarkable ability to make accurate predictions for unseen samples and test conditions. Their later work [18] conducted a similar study using a penalized logistic regression model, which could help identify factors that impact the flashover occurring.

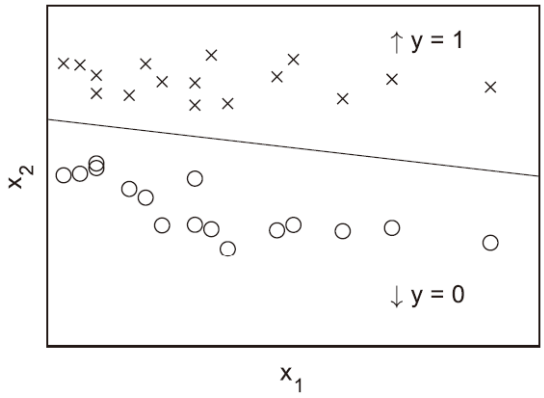


Figure 4: An illustration of decision boundary for flashover prediction, from [17].

2.1.2 Flashover Prediction with Deep Learning

The success of Deep Learning in recent years also drew the attention of fire researchers. With the help of Deep Learning, new solutions are sought for many classification and prediction problems in fire research, and the processing speed and accuracy have also been improved.

Fu et al. [18] built a flashover prediction model with deep neural networks, which can be used to warn firefighters before flashover occurs. They used CFAST developed by NIST as a simulation engine and used it to generate synthetic data. They then validated the fire simulations with full-scale fire experiments, and the overall results showed that their model's prediction accuracy was around 75%. Besides, generative models were also introduced in flashover prediction. Yun et al. applied conditional Generative Adversarial Network (cGAN) for image enhancement in [19] to enhance the dark video images of fire and smoke, which could be used to analyze and predict temperature variation for flashover. Their temperature analysis for flashover prediction is shown in **Figure 5**. This figure shows the variation of the number of pixels with a specific temperature range. They also used temperature as the criterion for flashover occurrence. The 'Flashover' at 190 seconds is the real flashover occurrence time and the 'Prediction' at 135 seconds is the prediction results of their system.

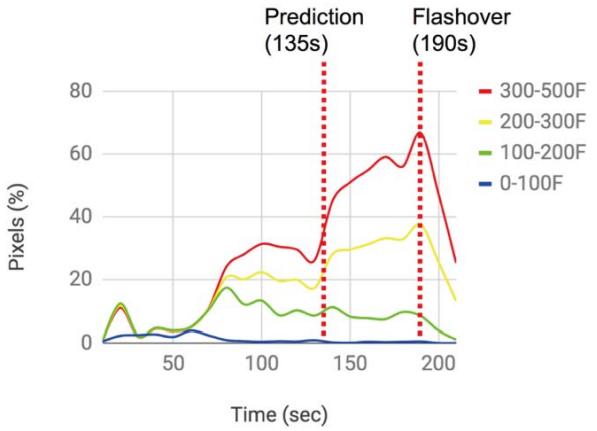


Figure 5: An illustration of temperature analysis for flashover prediction, from [19].

Besides, hybrid models of deep neural networks are also popular in flashover prediction to analyze various parameters. Yap et al. [20] introduced a model based on the Generalized Adaptive Resonance Theory (GART) neural network developed based on integrating Gaussian Supervised Real-Time Learning and Classification of Nonstationary Data by a Self-Organizing Neural Network (ARTMAP) and the Generalized Regression Neural Network. Their model demonstrated that it outperformed other networks and produced meaningful rules from data samples. In addition, Lee et al. introduced a network called GRNNFA, which is a fusion of the Fuzzy Adaptive Resonance Theory (FA) model and the General Regression Neural Network (GRNN) model in their work. They compared the performance of the GRNNFA with other published results, and it surpassed other models with deep neural networks.

Synthetic datasets were built using a fire modeling tool, *Fire Dynamics Simulator (FDS),* in many studies [21–23] for room fire research adopting ML or deep learning. Others used the established mathematical models to analyze an intermediate variable for flashover, such as HRR [24,25]. They further analyzed the data with fire research knowledge and experiences.

2.1.3 Limitations of the previous studies

Deep Learning is a hot topic in almost all current research areas. It has demonstrated capabilities surpassing traditional and human-level methods in recent years in image-related research, especially image recognition, segmentation, and classification. Utilizing the potential of Deep Learning, fire research also tried to integrate Deep Learning into fire research with image-related analysis, and it opened up new research ideas. Some researchers explored the possibility of combing temperature information with visual information in fire scenes to build their models for occluded object reconstruction and fire development analysis [19,26]. Meanwhile, some others brought Deep Learning models to fire simulation to improve the efficiency and accuracy of flashover analysis and prediction [14,18].

Table 1: A summary of classifications of existing models on flashover prediction.

Algorithm basis		**Source of dataset**			**Types of data capture device**	
Deep Learning	Fire knowledge	Real-world	Synthetic	Real-world + Synthetic	Sensors	Video cameras
27.3%	72.7%	9.1%	72.7%	18.2%	88.9%	11.1%

However, flashover analysis and prediction are very challenging tasks. Many of the existing flashover forecasting research methods and models have problems and cannot be put into practical use. **Table 1** summarizes the survey conducted under this project on the previous flashover prediction studies using data

science. A total number of 22 studies conducted between 1995 and 2020 are reviewed for the classifications of their methods (Deep Learning-based or fire knowledge-based) and the source of data set (real-world data or synthetic data or combined) and the types of data (sensor data or image data). The survey results indicate that one of the weaknesses of the models from the previous studies is that their datasets are not representative of real scenarios since many studies used synthetic data due to the lack of real-world data. Also, the types of data used in the previous studies are primarily sensor data, which are difficult to be obtained from real fire scenes unless they are readily installed in the fire scene and high temperature/harsh environment endurable. For example, the upper limit of the temperature recorded by an existing handheld IR camera is mostly about 500K, and the temperature of an indoor fire can reach 800K [27]. It leads to insufficient accuracy of real-world data and range of existing equipment suitable to collect data from fire scenes [28]. For the reasons, many of the previous studies used synthetic or simulated data from tools, such as FDS, which has been proved to be an accurate and effective fire simulation tool.

Another crucial problem is the lack of evaluation metrics. The prediction results from the previous studies for flashover are limited to binary outputs, whether it will happen or not [29,30]. In an actual fire scene, firefighters need to know how much time is available prior to a flashover. In addition, most of the existing flashover prediction methods that could predict the onset of flashovers are based on post-fire analysis because they need to analyze the entire information from start to end. It makes their methods not applicable to real-time analysis.

Thus, to overcome the limitation of the previous studies, it is suggested to develop a smart firefighting tool to predict flashover using real-world data, based on image data rather than sensor data since vision cameras and IR cameras are currently used by firefighters. Vison RGB images could be used as inputs for the prediction system as long as real-time image data processing is possible.

2.2 Deep Learning Algorithms

There are several deep learning algorithms reviewed and used in the present study. This section describes the algorithms adopted in the system that is developed in the present study and other algorithms reviewed for comparisons with the system.

Generative Adversarial Networks (GANs), which are a group of unsupervised Deep Learning frameworks, are reviewed for data generation required in the training process. For effective data generation and learning, two existing methods are selected and explored in the present study since they are designed for tasks like image translation and augmentation. They are Cycle Generative Adversarial Network (CycleGAN) [31] and Attention-Guided Generative Adversarial Network (AGGAN) [32].

There are three critical components in these algorithms: cycle structure, foreground attention, and background attention. The cycle structure is a basic structure for un-paired image conversion tasks, and both CycleGAN and AGGAN are equipped with the cycle structure to deal with un-paired images as input for the neural network.

The foreground attention aims to provide clear and better quality for foreground contents of images, such as flame and smoke of a fire. While CycleGAN has the foreground attention feature, AGGAN does not have it. So, this structure makes AGGAN generate images with better quality than CycleGAN. The background attention could improve image quality by separating the foreground and background areas in image conversion. However, both CycleGAN and AGGAN do not have the background attention feature. As described in the later chapter 3, the present study proposed a new algorithm called Dual Attention Generative Adversarial Network (DAGAN) with both the foreground and background attention features. The comparisons of the performances of these algorithms are discussed in Chapter 4.2.1.

One of the fundamental tasks in the Computer Vision field is segmentation, especially semantic segmentation. Sematic segmentation plays a broad role in various applications such as processing medical image analysis, robotic perception, video surveillance, augmented reality, and image compression.

Several deep learning algorithms are reviewed in the present study for Video Semantic Segmentation. Pyramid Scene Parsing Network (PSPNet) [33] and Deep learning Lab V3 (DeepLab V3) [34] are studied since they are designed for image semantic segmentation algorithms. Also, this study explored video semantic segmentation algorithms: Reference-Guided Mask Propagation (RGMP) [35], Semantic Video Convolutional Neural Network (SV-CNN) [36], Semantic Video Segmentation (SVS) [37], and Temporal Distributed Network (TD-Net) [38].

PSPNet and DeepLab V3 are designed for image semantic segmentation, which means that they do not use inter-frame information. However, as our input would be video, the inter-frame information is vital to form a segmentation result with high quality and high consistency.

The four algorithms designed for video semantic segmentation tasks use residual feature extraction methods for the in-frame information, which is a good base for segmentation performance. As for the structure for inter-frame information, RGMP uses reference-guided masks, which have been popular in the past few years, to improve pixel-wise recognition ability. Both SV-CNN and SVS use Dual-CNN for inter-frame calculation, while SV-CNN focuses on optical flow and SVS focuses on overall pixel intensity. TD-Net is the method that is chosen for our system. It has knowledge distillation and group convolution, which are an excellent boost for scene understanding for segmentation and high segmentation speed. It is hard to tell which one of the four would better perform for flame and smoke segmentation tasks before evaluation.

Video prediction is necessary for a self-supervised learning task to develop a smart tool to predict future events based on video feeds. The video prediction algorithms are capable of extracting meaningful representations of the patterns in input videos. Although video prediction tasks would be easy for humans with additional physical knowledge, deep learning algorithms are still highly challenging. Factors contributing to such complexity are occlusions, camera movement, lighting conditions, clutter, or object deformations. For the video prediction, the present study reviewed Convolutional Neural Network with Laplacian pyramid (CNN-LP) [39], Convolutional Long Short Term Memory (Conv-LSTM) [40], Stochastic variational video prediction (SVVP)[41], Appearance and Motion Conditions Generative Adversarial Network (AMC-GAN) [42] and Stochastic Video Prediction (SAVP) [43].

CNN-LP and Conv-LSTM use Convolutional Neural Network and Recurrent Neural Network, respectively, for prediction tasks, both of which are traditional network structures making their performance stable but not very high in prediction quality. Variational Autoencoder adopted in SVVP is a conditioned generative model, which means that it would generate prediction with good plausibility but lack of variety. On the other hand, AMC-GAN uses Generative Adversarial Network, an unconditioned generative model that would produce prediction with diversity. The SAVP uses both Variational Autoencoder and Generative Adversarial Network, which would provide prediction with diversity and visual plausibility. For video prediction, diversity guarantees the number of choices that a user could choose for fine-tuning, and plausibility makes sure that data generation is good in video quality. As a result, SAVP, which is chosen by this study for our module, has the potential for the best performance for prediction in the flashover prediction asks.

3 Design and Methodologies

3.1 Design of Entire System

A smart firefighting tool is designed to predict room fire flashover based on images analysis to process input image data in real-time and provide a prediction for the onset of flashovers. This tool is an end-to-end system that takes an RGB image as input and returns flashover prediction results as output.

The overall process design of the tool is depicted in **Figure 6**. For the input, the system reads RGB video data obtained from a vision camera. A frame from the video input is generated and processed as designed for flashover analysis at each time step. The system aims to analyze those frames in real-time and predict an occurrence of flashover based on the analysis of information from input frames.

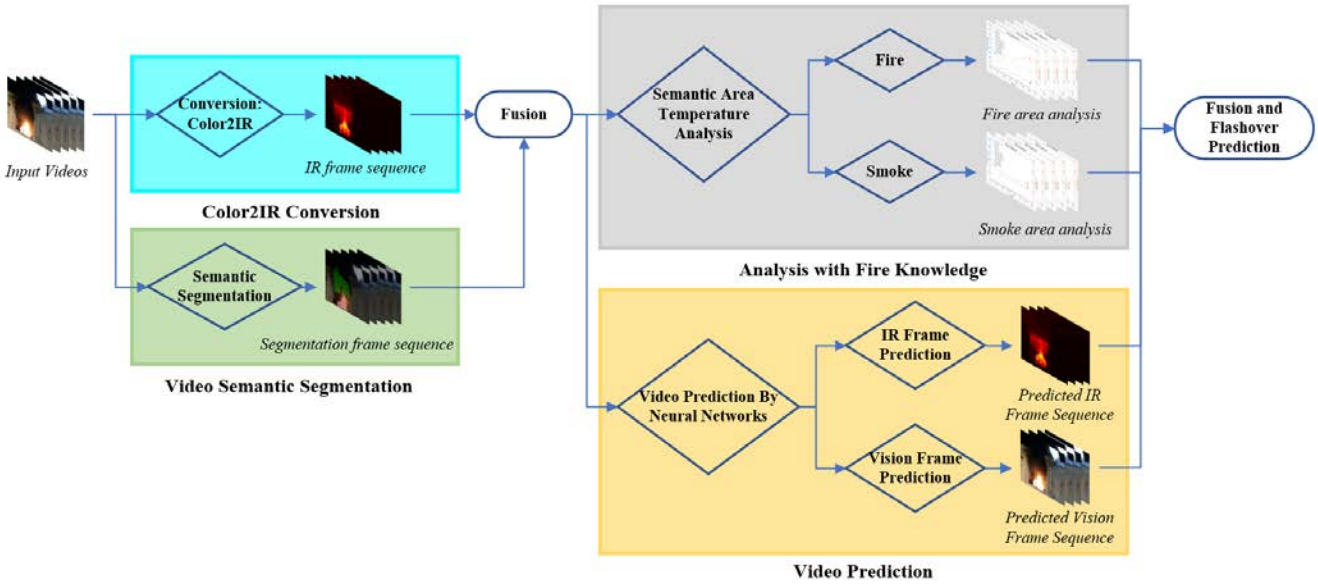


Figure 6: Overview of our system.

The system consists of four modules: the Color2IR Conversion Module, Video Semantic Segmentation Module, Analysis with Fire Knowledge Module, and Video Prediction Module. A fusion part is also proposed in order to maximize the integration analysis of that information.

As the first module in our system, the Color2IR Conversion Module processes thermal information from input vision frames by converting them to IR frames. It is a core part of the entire system, as it provides thermal information by transferring color images to IR images. Knowing the temperature/thermal condition of the entire room provides much better information in predicting flashover than working with limited point measurements of temperature. Although some other parameters could also be suitable information bases for fire analysis, like HRR and flame height, the temperature is first explored in this study. The crucial part in the IR conversion is a deep neural network by DAGAN, which is evolved from famous existing cross-domain image conversion methods such as the DiscoGAN [44] and DualGAN [45]. In addition, self-attention in the Computer Vision areas, like Self-Attention [46] and loop structure used in CycleGAN [31] are also incorporated. These features are optimized to work together and contribute to a powerful model for the Color-to-IR image conversion algorithm.

The input data also goes to Video Semantic Segmentation Module, which produces semantic information for flame and smoke areas. The core part of this module is also a deep neural network called TD-Net [38]. Due to its excellent performance in segmentation, accuracy and speed is the best choice for our module in detecting smoke and flame patterns/areas. It also incorporates the Knowledge Distillation [47] for the speedy and accurate transformation of knowledge from a deep teacher network.

The processed data from the Color2IR Conversion Module and Video Semantic Segmentation Module is fused to flow into the analysis with fire knowledge, which processes a pair of the converted IR frame sequence and corresponding segmented flame and smoke patterns in vision frame based on a mathematical model and

statistical analysis. This hybrid approach employs the experience and knowledge gained through years of room fire research, validated and reliable [13]. Besides, the hybrid approach is suitable to explore the feasibility of image processing of smoke and flame; and predicating impeding flashover adopting Deep Learning methods.

Simultaneously with the analysis with fire knowledge, the Video Prediction Module also takes the fused data to produce possible future information in image formats. The direct source of the video prediction is the collected frames of the input vision frames and the converted IR images. The core part of this module is a deep neural network called Stochastic Adversarial Network (SAN) [43]. It takes advantage of combining the high-quality output without blurry and diverse predictions. These are important to our system since future information in clear image format would result in high accuracy predictions in the next step of the system. It becomes important for dynamic/diverse images, such as images of fire phenomena.

In the end, our system will make a decision of flashover prediction based on the analysis results from the Analysis with Fire Knowledge Module and Video Prediction Module. Our system is not only capable of providing both binary prediction of flashover occurrence but also an *Estimated Time Arrival (ETA)* as a countdown of flashover occurrence for the future. Those two results would assist the users in making decisions for escaping from flashover or firefighting tactics.

3.2 Design of sub-modules

The system adopts a modular design aiming to split specific functions and characters, making real-time prediction feasible. The system is designed to generate the final prediction results via fusion of the results from each step and linear mathematical analysis. This section provides brief descriptions of each sub-module, and more detailed processes of each sub-module are provided in Appendix, which also provides parameter settings in loss function for our deep neural networks.

As one of the most crucial sub-modules in the system, Color2IR Conversion aims to provide corresponding IR images that could tell the temperature of each pixel from a visual image captured from a standard camera that could be taken into fire rescue with firefighters. The input videos would be cut into independent frames in this module and processed as a single unit. Besides, it is a kind of cross-domain image transfer task in the Computer Vision field as the images of input and output are from different types.

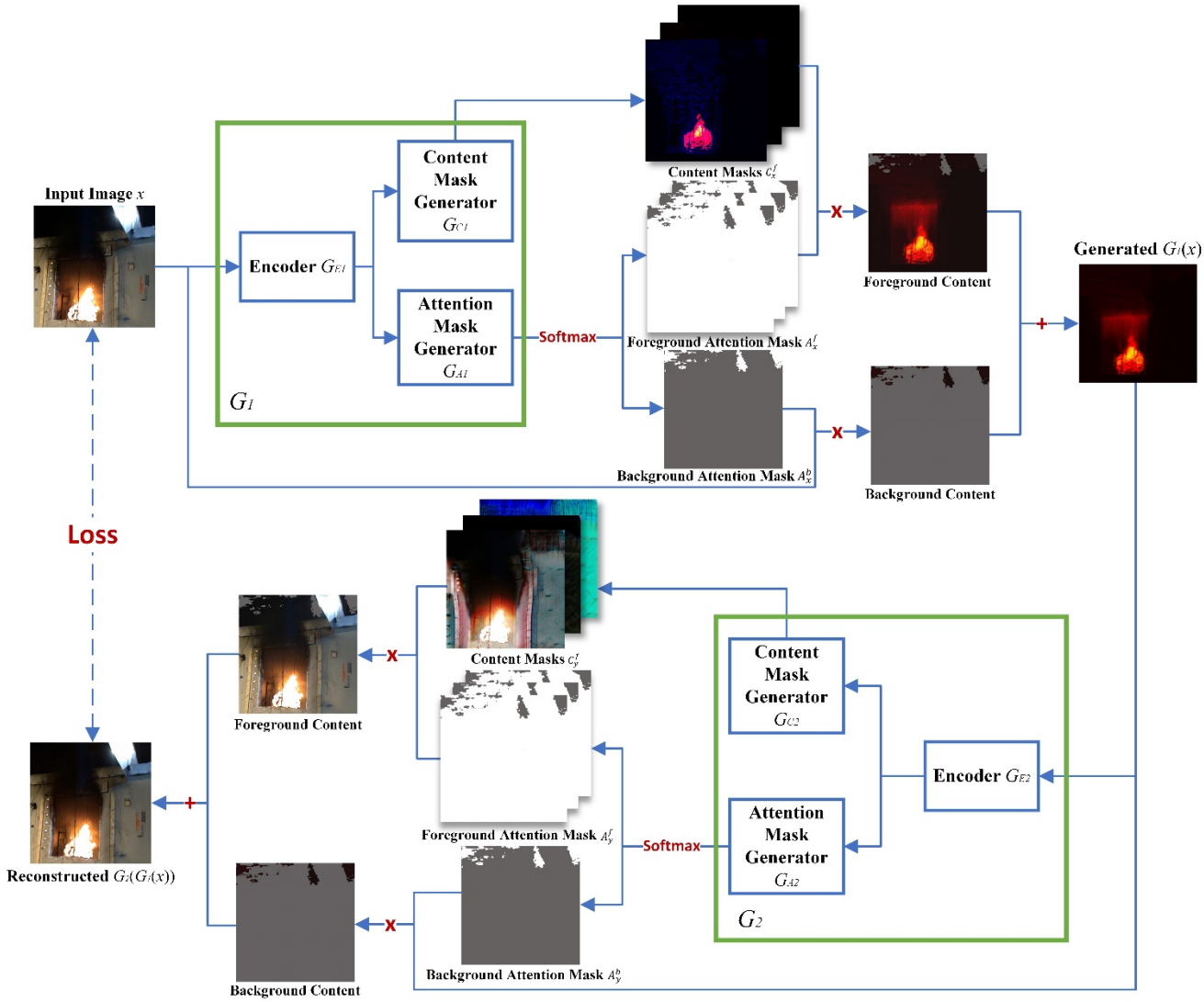


Figure 7: An illustration of DAGAN architecture. ('×' stands for multiplication of matrices, '+' denotes the sum of matrices, and 'Softmax' is the Softmax activation function. ' Loss' is the Cycle-Consistency Loss inspired by CycleGAN.)

Figure 7 shows the overall structure of Dual-Attention GAN (DAGAN), a novel deep neural network used in the system, which is inspired by the success of CycleGAN in un-paired image conversion. The input images of DAGAN are from the visual videos of fire scenes, which are denoted as x in **Figure 7**. The input x will be fed into the generator G_1, which consists of an encoder G_{E1} and two mask generators: G_{C1} and G_{A1}. G_{E1} is a parameter-sharing encoder which could generate low-level feature maps. While G_{C1} is a content mask generator that could generate a set of masks C_x^f, which contains sets of the content feature captured from the encoder G_{E1}.

G_{A1} is a generator for attention mechanism providing attention-level feature maps from the encoded information. The direct output of G_{A1} is processed by a Softmax activation function, and it would produce two types of attention masks: A_x^f and A_x^b.The foreground attention mask and background attention mask enable DAGAN to differentiate the foreground and background images. Then, the foreground information and background information extracted from input x will be processed independently. The final generated image $G_1(x)$ would be the sum of them.

That is the end of the generation process and the start of the reconstruction process. The reconstruction is inverse to the generation process in structure, while the training process would be independent.

In addition, vanilla discriminators are used to distinguishing the generated images $G_1(x)$ and real images y or $G_2(y)$ and x. The system also employs loss functions proposed by us. There are several loss functions in DAGAN: an adversarial loss that same as vanilla GAN, and a loop loss or cycle loss (as shown in the dotted line in DAGAN between original input x and reconstruction result $G_1(G_2(x))$, as in **Figure 7).** DAGAN also employs pixel loss (to constrain the generator without discriminator information at pixel level), identity loss for pixel-level measurement in CycleGAN, and Attention Adversarial loss in AGGAN (to form a stable attention mask in the training process without any annotations on the image pairs in the training set) as well as a pure attention loss (to improve the stability and performance of attention masks). These loss functions used in DAGAN are optimized by piecing them all together with weights.

In this way, a closed-loop for the DAGAN process is finally formed, starting from the input x to the reconstruction of G_2.

There is another sub-module that takes the original input images, which is **Video Semantic Segmentation** Module generating semantic information for fire scenes. Real-time video semantic segmentation results are required for accurate and speedy processing of the input images. The system adopts TD-Net [39], which is a type of neural network for video semantic segmentation. The basic idea of TD-Net is Group Convolution, which extracts features with separated filter groups instead of only one guaranteed model parallelization and representations. The sub-networks design and Attention Propagation Module (APM) contribute to fast and consistent segmentation.

TD-Net conducts the Encoding Phase, first, where the network generates path-specific feature maps and Query and Key maps for across-frames correlating between pixels. Then, it calculates the attention from Value, Query, and Key as a self-attention mechanism. These feature maps are merged to effectively capture non-local correlations between pixels across frames with the help of this self-attention mechanism. After that, there is a down sampling process to reduce the computation costs. The second phase of TD -Net is the segmentation phase, which includes a propagation approach that measures the attention of neighboring frames. Then, it finally computes the final feature representative at each time frame, generating segmentation maps. To enhance the sub-feature maps in the entire feature space, it adopts Grouped Knowledge Distillation mechanism.

As shown in Figure 6, after the Color2IR Conversion Module and Video Semantic Segmentation Module, **Video Prediction Module** runs the subsequent processing using the power of neural networks to provide reliable visual results for fire scenes. Generative models (the state-of-the-art methods) are employed for Encoder-Decoder models, which provide predictions with diversity, and GANs models are used for naturalistic predictions. Thus, a combination of them is Stochastic Adversarial Video Prediction (SAVP), which provides a prediction with stochasticity as well as plausibility. It consists of two parts. The first part is a Variation Autoencoder (VAE) that also acts as a generator. The generator predicts the future frames with the previous ones and latent codes, which specifies a distribution based on a fixed variance Laplacian distribution. The second part is GAN, where a generator provides a prediction of future frames. With a discriminator distinguishing the generated frames from original ones, the generator would be trained using binary cross-entropy loss. Compared with VAE, GAN could generate predictions with higher diversity. The results of VAE would be more visual plausibility. As a result, the combination of VAE and GAN in SAVP would make our Video Prediction Module produce predictions with high diversity and visual plausibility.

Another essential step in the present Flashover Prediction System is Fire Knowledge Module. While Deep learning is a popular choice for many research fields, yet a hybrid approach is taken in the present explorative study by employing Fire Knowledge Module. Many previous fire safety research studies show that linear mathematical models are still influential in processing conventional measurement data, such as temperature.

Thus, taking a similar approach, the Analysis with Fire Knowledge Module in our Flashover Prediction System is proposed to extract and process the data acquired from the input images using statistical analysis methods. There are several choices for parameters (e.g., temperature, HRR, and flame height) for the statistical and mathematical models in flashover analysis and prediction. As the temperature distribution information could be directly extracted from the output of the previous modules in our system, the temperature is chosen for a flashover criterion. As widely accepted, a typical flashover happens when the upper layer of smoke in the room is approaching or above 600℃ for normal conditions in a typical room [11]. Using the temperature data extracted from the input, a statistical graph for the real-time analysis of flashover occurrence.

In addition, for forecasting the onset of flashover, the system combines the prediction frames from the Video Prediction Module and extracts information in the converted IR and visual domain. The detailed solution proposed for this problem only produces a limited number of prediction frames, like 5 or 10 frames. And, it would make them visually plausible and full of contextual information from the original input. It is called applicable predictions. Then, the system combines data and generates the graph of statistical information; thus, it could be regarded as a tangent of specific points representing the original input, shown as in the figure below.

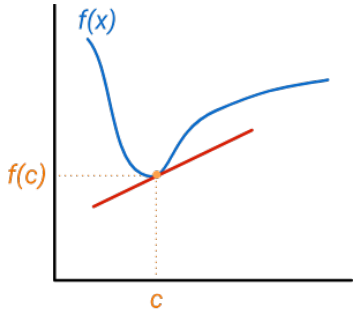


Figure 8: An illustration of applicable predictions and the temperature data curve.

Then, following the definition of derivatives, we could approximate the future point on the graph with current data and the tangent, formulated as the equation below.

$$f(c_f) = f(c) + \sigma \cdot (c_f - c) \qquad \text{eq. } 1$$

Where c is the point of original frames.
c_f denotes the points for the future.
σ is the tangent value.

The statistical temperature graph is in a discrete domain, it needs to be linked and form a continuous curve. The tangent is updated with every input frame. It could also help prevent collapse problems in fluctuation points or spikes in the temperature curve, shown in **Figure 9**.

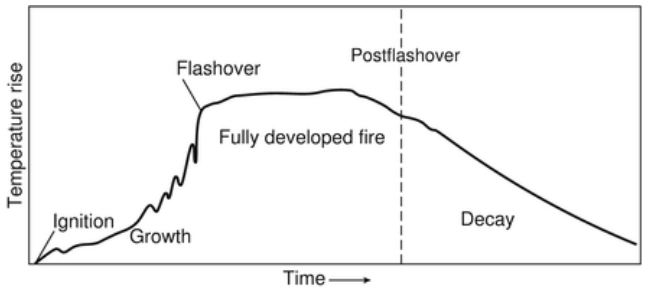
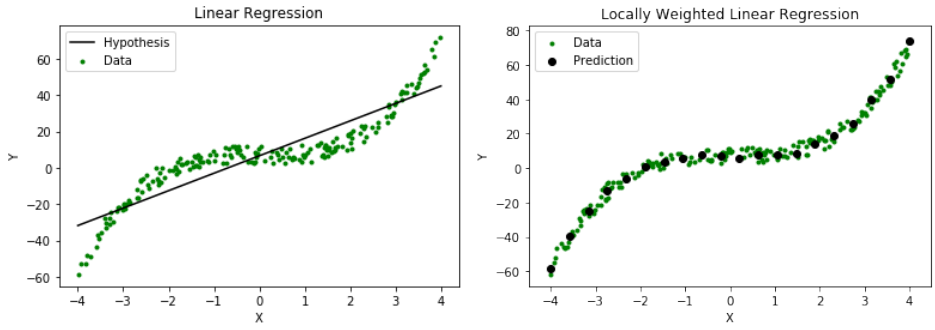


Figure 9: An illustration of temperature variation with time. The fluctuation point is between flashover and the growth stage of fire development.

Furthermore, locally weighted linear regression is introduced as the core mathematical model in this sub-module. Locally weighted linear regression is a supervised, non-parametric learning algorithm. The model does not learn a fixed set of parameters as it is done in ordinary linear regression. Parameters θ are computed individually for each query point x. While computing θ, a higher "preference" is given to the points in the training set lying in the vicinity of x than the points lying far away from x. There is no training phase in the overall process of this algorithm, and all the work is done during the testing phase or while making predictions. Compared with ordinary linear regression, it could prevent overfitting and underfitting problem and give a better regression model for prediction. An example of linear regression and locally weighted linear regression is shown in the figure below.



a) b)

Figure 10: An example of the comparison of ordinary linear regression and locally weighted linear regression in prediction. a): ordinary linear regression. b): locally weighted linear regression.

For ordinary linear regression, it needs to minimize a target in the equation as follow:

$$J(\theta) = \sum_{i=1}^{m} \left(\theta^T x^{(i)} - y^{(i)}\right)^2 \qquad \text{eq. 2}$$

And the prediction for query point x will be $\theta^T x$.

While the locally weighted linear regression assigns weights $w^{(i)}$ to each regression point, as shown in equation 3.

$$J(\theta) = \sum_{i=1}^{m} w^{(i)} \left(\theta^T x^{(i)} - y^{(i)}\right)^2 \qquad \text{eq. 3}$$

If the $x^{(i)}$ is lying closer to the query point x, the value of $w^{(i)}$ will be larger. Otherwise, it will be smaller.

A typical choice for the weights $w^{(i)}$ is:

$$w^{(i)} = exp\left(\frac{-\left(x^{(i)} - x\right)^2}{2\tau^2}\right) \qquad \text{eq. 4}$$

Where τ is the bandwidth parameter and controls the rate at which $w^{(i)}$ falls with distance from x.

For the FPS setting, we set the FPS in Analysis with Fire Knowledge module to 1, matching the settings in previous modules, which could reduce computation cost and remain precision analysis. The number of predicted frames is set to 3 to match the FPS for a better prediction result.

With the locally weighted linear regression and tangent prediction model set for our system, this sub-module could smoothly proceed the data from previous sub-modules and do the analysis and prediction of flashover with fire knowledge criterion of flashover.

4 Evaluation

4.1 Experimental Setup

4.1.1 Software and Hardware Setup

The hardware is a workstation computer with Intel and NVIDIA chips. For specs, it has Intel Core I9-10900K, 64GB system memory, and 3TB Solid-State-Drive (SSD). For the GPU part, which is essential for deep learning research, we use dual RTX2080 Ti with 11GB GDDR6 dedicated video memory and an RTX3090 with 24GB GDDR6X dedicated video memory.

For the software on the computer, we use Windows 10 build 19041.985 and Ubuntu 18.04 for the operating system. We chose *Pytorch* as our deep learning framework and CUDA 11.1 and CUDA 10.2 for the GPU driver framework. Besides, there are also some other packages, such as *NumPy*, *matplotlib*, *OpenCV*, *tensorboardX*, etc.

4.1.2 Dataset Preparation

For training, testing, and validation of the Deep Neural Networks studied in the current project, image datasets are gathered and prepared. The data sets were prepared for the sub-module training/testing and the entire system testing/validation. A well-designed model alone cannot achieve good performance in real-world situations without extensive training and testing over data sets sufficiently representative to the problem of interest. Therefore, fire test data were collected and sorted by the types of images (e.g., vision and IR), and the collected video data were also identified with the corresponding test data (e.g., temperature and HRR) for the temporal fire development and flashover, if available.

4.1.2.1 Sub-module Dataset Preparation

The sub-module of the Color2IR is tested with 1800 image pairs from 17 fire experiments conducted under the CFMRD (Characterization of Fires in Multi-Suite Residential Dwellings) project [48] by the *National Research Council (NRC)*, Canada. In the project, single and multiple household furniture and items were burned in a typical room with heavy instrumentations to characterize fires in residential dwellings. The selected 17 fire experiments used in this study are listed in **Table 3**. The tests with the name starting with 'SI' were individual furnishing tests where a mattress, bed assembly, workstation, or upholstered furniture was placed in the test room (with dimensions 3.8 m wide x 4.2 m long x 2.4 m high) with a window (1.5 m x 1.5 m). The tests with the name starting with 'RBF' tested a set of fully furnished living rooms (RBF-12) or bedrooms (RBF-07). The living room (3.8 m wide x 4.2 m long x 2.4 m high) had a window of 1.5 m x 1.5 m, and the bedroom (3.2 m wide x 3.5 m long x 2.4 m high) had a window of 1.4 m x 1.2 m. The test rooms were instrumented to measure HRR, room temperature, heat flux on the floor. A vision and IR camera[1] were also placed outside the test room to capture the temporal fire development. The tests with the name starting with 'M' were burning tests in a metal box. A vision camera and an IR camera were also placed outside the test room to capture the temporal fire development for this test.

---

[1] A VarioCAM infrared camera with the specifications of 384x288 pixels, the IR images were pre-processed with Cubic spline interpolation to match the resolution with vision images.
Spectral range: 7.5 to 14 µm, Temperature measurement range: -40°C to 1200°C and measurement accuracy: ± 2 K, ± 2 %*.

The image data are cut from the video recording of the vision and IR sources. All the image pairs have been verified and synchronized according to the official record of NRC test reports. A detailed description of the Color2IR dataset is shown in the table below. The ratio of training and testing part is 9:1. An overview of the Color2IR dataset is shown in **Figure 11** below. IR images use a 1024-level constant Colorbar that ranges from $(280, 1400)$ *Kelvin (K)* to transform the temperature information to the color domain. The Colorbar is shown on the right side of IR images.

Table 2: Description of the Color2IR dataset.

Name of NRC test	Burning items	Resolution of image pairs	Total number of images
PRF-07	-	400×400	120
PRF-12	-	400×400	90
M-1	metal box	640×480	230
05-SI-03	Mattress	400×400	90
08-SI-04	Mattress	400×400	100
14-SI-06	Mattress	400×400	80
31-SI-13	Bed assembly	400×400	100
16-SI-16	Wardrobe	400×400	140
33-SI-21	workstation	400×400	80
22-SI-22	Toys	400×400	90
10-SI-24	Chair	400×400	100
09-SI-25	Chair	400×400	60
26-SI-26	Chair	400×400	70
15-SI-27	Chair	400×400	80
23-SI-76	Bed assembly	400×400	80
19-SI-83-1	Bed clothes	400×400	60
19-SI-83-2	Bed clothes	400×400	80

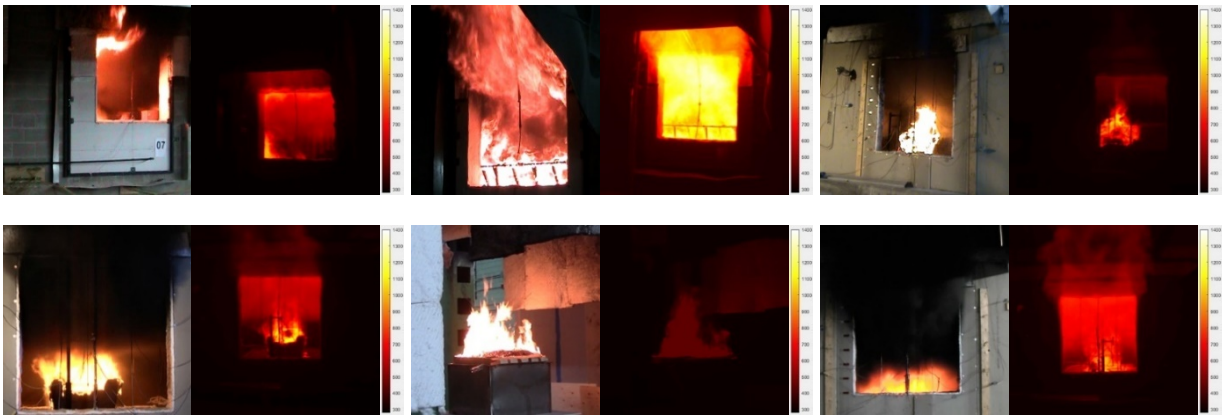


Figure 11: Samples from Color2IR dataset.

For the Video Semantic Segmentation Module, which is required to be trained and tested for classifications of smoke and flame patterns, a new dataset of FS segmentation was prepared. The dataset contains 40 image sequences collected from various sources. The selected data from the CFMRD project conducted by NRC (introduced in the Color2IR dataset) were used, and video data publicly available were also included. These data are videos captured by firefighters' equipment [5] and fire rescue videos posted on YouTube [49]. Also, video data from one of the room fire tests conducted by NIST is also included in the dataset [50]. In addition, some of the synthetic fire images generated by Blender are also included in the FS Segmentation dataset. Shown in the seconds' row of **Figure 12 is** a sample image of the synthetic smoke and flame pattern is generated using Blender[2], a free and open-source 3D computer graphics software for computer animation. It builds life-like smoke and fire patterns and merges them into real scenes, such as the room shown in **Figure 12**. Thus, the Video Semantic Segmentation Module was trained and tested with various fire and smoke scenes captured in a fire room, through the window of a fire room, and outside a building in fire.

Each sequence in the dataset contains images in 2 seconds of the original videos. The number of images in each sequence depends on the Frame Per Second (FPS) of its original video. The ratio for training and testing is 9:1. A description of the FS Segmentation dataset is listed in **Table 3** below. An overview of samples with their annotations from the FS Segmentation dataset is shown in **Figure 12** below.

Table 3: Description of the FS Segmentation dataset

Source and sequence name	Numbers of images in each sequence	FPS of the original video	Number of sequences	Total number of images
Firefighters, Firerescue-1	48	24	2	96
Firefighters, Firerescue-2	48	24	3	144
NRC, PRF-07	60	30	3	180
NRC, PRF-12	60	30	4	240
NRC, M-1	60	30	4	240
NRC, 16-SI-16	60	30	4	180
NRC, 26-SI-26	60	30	4	180
NRC, 23-SI-76	60	30	4	240
YouTube, NISTvideo-1	48	24	4	192
YouTube, NISTvideo-2	48	24	4	192
Synthetic, Blender-1	60	30	2	120
Synthetic, Blender-2	60	30	2	120

[2] We use α-channel edge processing for blending.

Figure 12: Samples and their annotations (R: flame, G: smoke) from the FS Segmentation dataset. The Upper left image pair is Christmas Tree tests from NIST. The upper right one is image pair from a Fire rescue video posted on YouTube [49]. The lower left image pair is synthetic images generated in this study using Blender. The lower right image pair is from the NRC PRF-07 test.

Next, for the Video Prediction, a new dataset of video fire scenes was prepared for training and testing. This dataset, Fire-Smoke Video Prediction (FSVP), consists of two parts: visual image sequences (FSVP-V) and IR (FSVP-IR) image sequences. We prepare two independent groups of images to train our Video Prediction Module to predict IR images and visual images. For FSVP-V, 60 image sequences were collected from the same four sources used for the FS Segmentation dataset: the videos captured firefighters' equipment, the NRC fire safety tests, and the fire rescue video of YouTube. Each sequence in the dataset contains images in 20 seconds of the original videos. The number of images in the sequence depends on the FPS of its original video. For FSVP-IR, 20 image sequences were collected from the NRC fire safety tests. Each sequence in the dataset contains 120 images of the original videos. The ratio of training and testing part is 9:1. A description of the FSVP dataset is listed in **Table 4** below. An overview of sample sequences from the FSVP dataset is shown in **Figure 13** below.

Table 4: Description of the FSVP dataset.

Partition of dataset	Source and sequence name	Numbers of images in each sequence	FPS of the original video	Number of sequences	Total number of images
FSVP-V	Firefighters, Fireresc ue-1	480	24	4	1920
	Firefighters, Fireresc ue-2	480	24	4	1920
	NRC, PRF-07	600	30	4	2400
	NRC, PRF-12	600	30	6	3600
	NRC, M-1	600	30	6	3600
	NRC, 16-SI-16	600	30	6	3600
	NRC, 26-SI-26	600	30	6	3600
	NRC, 23-SI-76	600	30	6	3600
	YouTube, NISTvideo-1	480	24	6	2880
	YouTube, NISTvideo-2	480	24	6	2880
	Synthetic, Blender-1	600	30	3	1800
	Synthetic, Blender-2	600	30	3	1800
FSVP-IR	NRC, PRF-07	120	1/2	4	480
	NRC, PRF-12	120	1/2	4	480

	NRC, M-1	120	1/10	3	360
	NRC, 16-SI-16	120	1/3	3	360
	NRC, 26-SI-26	120	1/3	3	360
	NRC, 23-SI-76	120	1/3	3	360

Figure 13: Samples from FSVP dataset (First row: FSVP-V, Second row: FSVP-IR).

4.1.2.2 Full System Dataset Preparation

For testing and validation of the entire system, a dataset was prepared. It is essential to have datasets that recorded actual fire scenes together with recorded flashover times for the Flashover Prediction System to provide a fast and precise prediction for flashover. A dataset of FP was prepared using the videos recorded from the NRC CFMRD project, selecting only the data from the fire tests where a flashover occurred. Also, fire video data from a room fire test conducted by NIST was also included together with their analysis for flashover time by the HRR and temperature measured in the test. Thus, the entire system was tested against the eight individual fire scenes. This hybrid system does not need to be trained on those samples, all of them were used for testing purposes. A description of the FP dataset is listed in **Table 5** below. Samples from the FP dataset are shown in **Figure 14** below.

Table 5: Description of the FP dataset.

Source of video	Sequence name	Sequence length (s)	Flashover time (s)
NRC	PRF-07	250	185
	PRF-12	150	94
	08-SI-04	250	169
	14-SI-06	250	157
	21-SI-10	150	95
	23-SI-76	200	113
	31-SI-13	300	227
NIST	NISTtest-1	50	22

Figure 14: Part of samples from the FP dataset (5th and 6th of each row is the start of flashover).

4.1.3 Sub-module Parameters Settings

To successfully run the training and testing with the datasets prepared, sub-module parameters need to be set properly for secure organic cooperation between the submodules. As illustrated in **Chapter 2**, each sub-module of the present system is designed with a specific target, whether to provide analysis or prediction for the flashover, and the sub-modules are configured to flow the image data in the system to yield a solid prediction as output finally.

There are generally two steps in the overall process of our Flashover Prediction System.

The first step in the overall process of the present Flashover Prediction System is generating temperature distribution features and semantic segmentation features for flame and smoke independently by the two sub-modules of the Color2IR Conversion and Video Semantic Segmentation. What becomes vital in this step is the synchronization of the two modules since they do not exchange shared information while processing the incoming data. When the input video feeds in, it will first be segmented into a sequence of frames, the selected of which go to the next step. The selection rate can be set by defining the FPS, and FPS=1 is used as it can provide continuous visual information over time. Then, the selected frames will be sent into the subsequent two sub-modules for processing. Since the two modules use different network architectures, the processing time of each frame is different, even with the same input with the same resolution. To solve the issues, FPSs are required to be set for each module. The Video Semantic Segmentation Module uses TD^2-PSP50, which is designed for fast and accurate processing and could achieve a processing speed of up to 10 FPS for an HD video. On the other side, the Color2IR Conversion Module, a speed of about 1.7 FPS, is suitable for an HD video. Considering the above, the FPS was set to 1 so as to match the speed of both sub-modules while reducing computation costs at the same time.

At the final step of the analysis and prediction for flashovers, the Video Prediction Module produces future frames in both vision and converted IR formats. The IR prediction is particularly important for predicting the temperature development in the test room.

The visual prediction is generally for the purpose of future reference and real-time evaluation of prediction results.

Due to the processing time differences between the two modules of the Video Prediction Module and the Analysis with Fire Knowledge, the overall processing speed is largely limited. The speed of IR frame prediction is 7.6 FPS, which means it takes less than 0.4 seconds to provide a 3-frames prediction for the future. Although the Video Prediction Module itself could achieve a high process speed, the four modules in our system need to match the speed with each other for cooperation. Thus, considering the FPS limitation in the previous step, the FPS is set as 1 for the final step.

Therefore, the entire system runs the Flashover Prediction System with all the sub-modules and provides a real-time prediction for flashover occurrence at 1 FPS for an input video at HD resolution.

4.2 Evaluation of sub-modules

The performance of each of the sub-modules was first evaluated as an independent task. Then, in order to evaluate the performance results from each sub-module, we not only introduced evaluation metrics for each of them but also evaluated the performance on our custom dataset.

4.2.1 Color2IR Module

The evaluation of the Color2IR module adopting the algorithm DAGAN was conducted with the Color2IR dataset. It is an unpaired dataset for image conversion. As a result, a qualitative evaluation study is conducted with the dataset. The performance of DAGAN was compared with other existing algorithms: CycleGAN and AGGAN. TABLE compares the features of the two algorithms with DAGAN used in this study. Chapter 2.2 provides the details of these features. As summarized in Table 6, both CycleGan and AGGAN do not have the background attention feature, while DAGAN is designed for both background and foreground attention features. With the Background attention feature improving the image quality by separating the foreground and background areas, DAGAN demonstrated better performance than the other algorithms.

Table 6: A comparison of the structure and components of algorithms for the Color2IR Conversion Module.

Name of methods	Cycle Structure	Foreground Attention	Background Attention
CycleGAN	True	False	False
AGGAN	True	True	False
DAGAN	True	True	True

Some sample images generated by DAGAN, CycleGAN, and AGGAN are shown in **Figure 15** below. As indicated by the labels above the images, each input is compared with the images generated by the deep neural networks of CycleGAN, AGGAN, and DAGAN, as well as the Ground Truth (GT) images (i.e., actual IR data).

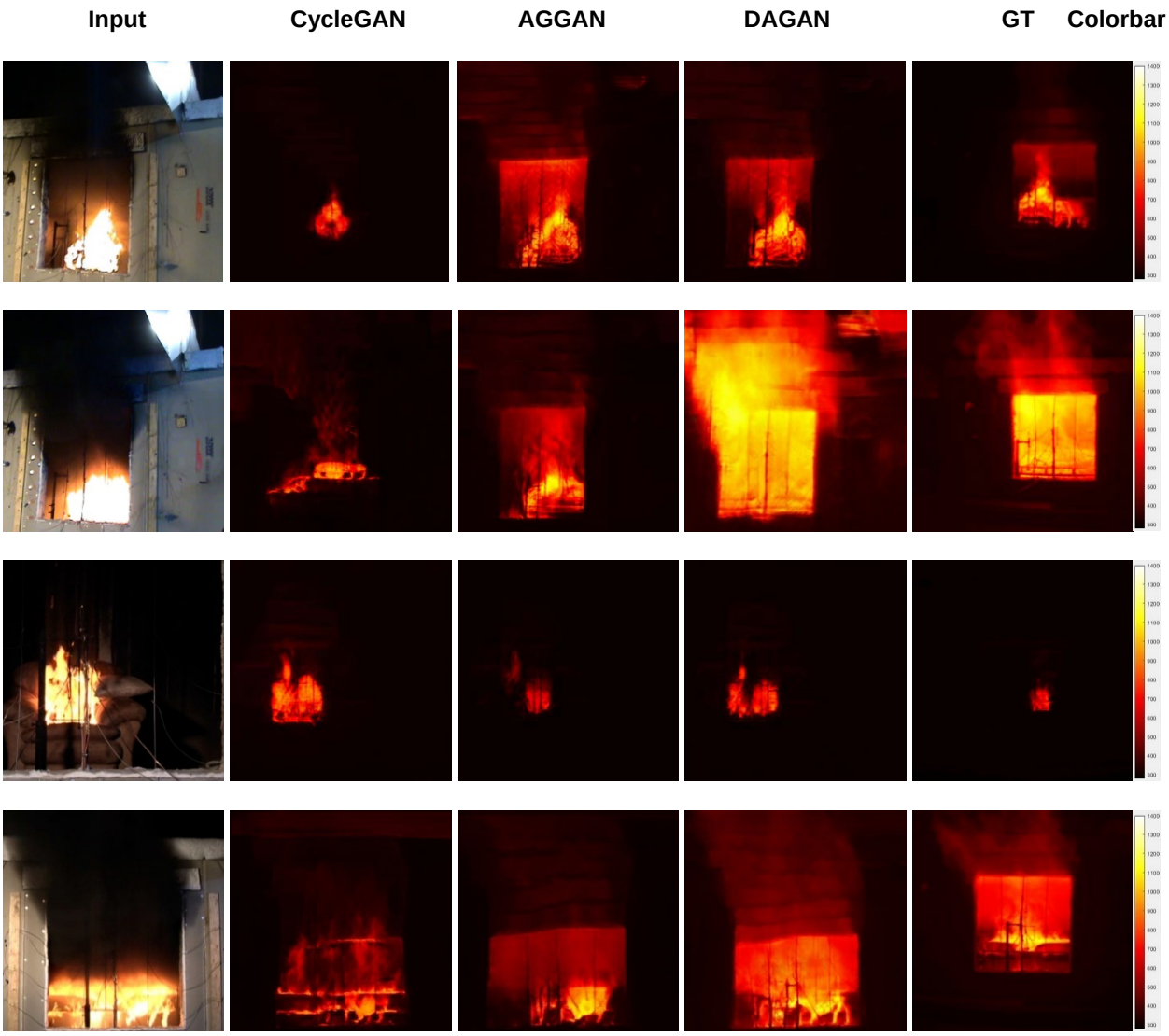


Figure 15: Samples of images, the label above denotes the source of each column.

From the samples presented in **Figure 15**, the quality of the generated images from DAGAN is consistently better than CycleGAN, which seems not to capture the 'conversion principle' between visual images and IR images. When compared to DAGAN and AGGAN, the performance of CycleGAN varies with the state of the fire development. For an early stage of fire, such as the images in the third row, AGGAN and DAGAN have similar conversion results of the background, while DAGAN could capture more details for the flame. When the fire goes on and a smoke layer appears, like the images in the first and fourth rows, DAGAN gives a more apparent transformation of the smoke layer and better temperature accuracy, compatible with the corresponding GT. For a fully-developed fire, like the images in the second row, DAGAN is dominantly better than AGGAN in producing the thermal image with higher accuracy, especially in the hot area.

However, DAGAN is not fully compatible with the GT even though it performed better than AGGAN and CycleGAN. A possible reason is that the Color2IR dataset is challenging for image conversions since it contains image pairs with different view angles in various test conditions. However, the Color2IR module contributed significantly to successful flashover prediction by the system.

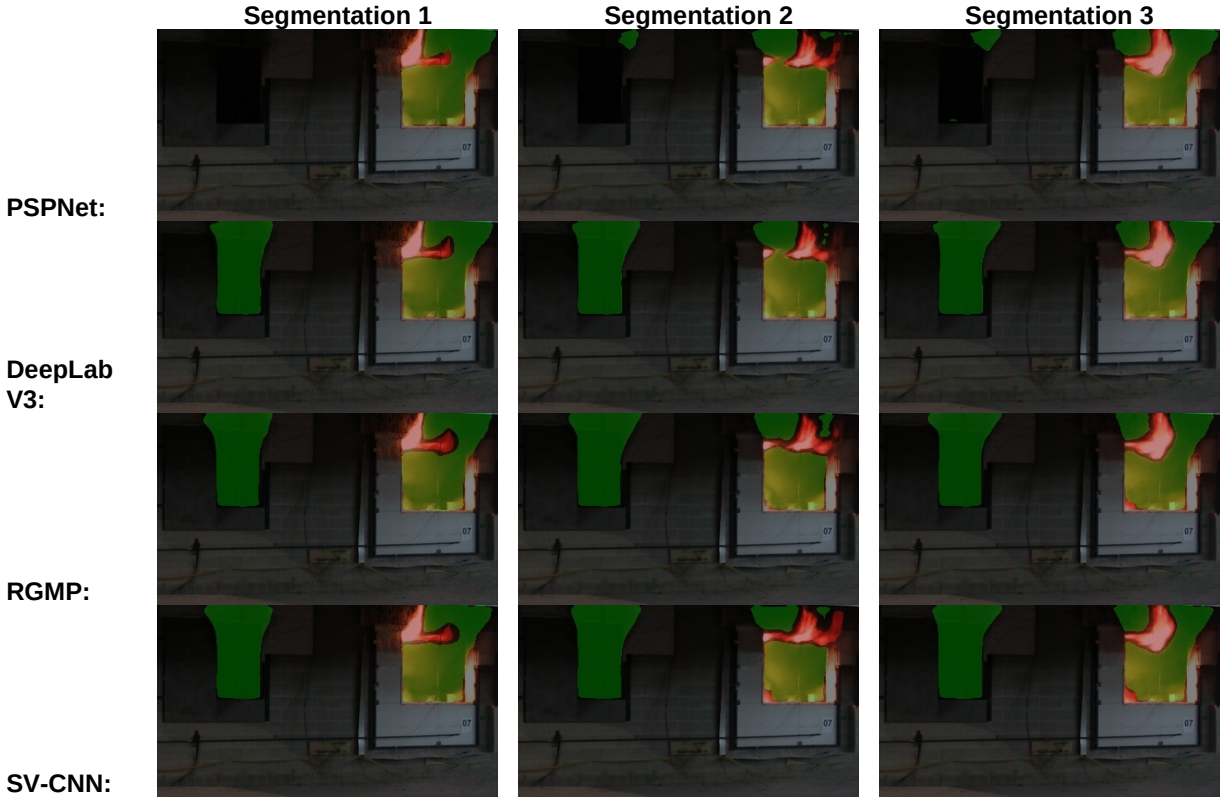
4.2.2 Video Semantic Segmentation Module

The Video Semantic Segmentation Module is evaluated with the FS Segmentation dataset. As discussed in CHAPTER 3, TD-Net is used in the module. TD-Net's performance is evaluated by comparing it to the other 5 existing state-of-the-art methods for accurate and speedy image/video segmentation, as shown in TABLE. The existing methods tests with the FS Segmentation dataset can be divided into two categories, depending on the purpose of their usage. The first is image semantic segmentation: PSPNet and DeepLab V3. The other is video semantic segmentation: RGMP, SV-CNN, SVS and TD-Net (used in our module). TABLE summaries the feature of each method, and detailed descriptions of the method are in CHAPTER#.

Table 7: A comparison of the structure and components of algorithms for Video Semantic Segmentation Module.

Name of methods	Usage of frame information	Usage of inter-frame information
PSPNet	True, Pyramid feature extraction	False
DeepLab V3	True, Dual feature extraction	False
RGMP	True, Residual feature extraction	True, reference-guided masks
SV-CNN	True, Residual feature extraction	True, Dual-CNN for Netwarp calculation
SVS	True, Residual feature extraction	True, Dual-CNN for transform flow calculation
TD-Net	True, Residual feature extraction	True, Knowledge distillation and group convolution

Figure 16 shows segmented images results, which compare the performance of each method. The labels at the left of the images are the method used in each row of images. The images in the same column are segmented frames simultaneously (Segmentation 1, Segmentation 2, and Segmentation 3). The video semantic segmentation methods show better performance in the segmentation accuracy. TD-Net employed in our module is one of the best among them, contributing significantly to the stable prediction of flashover.



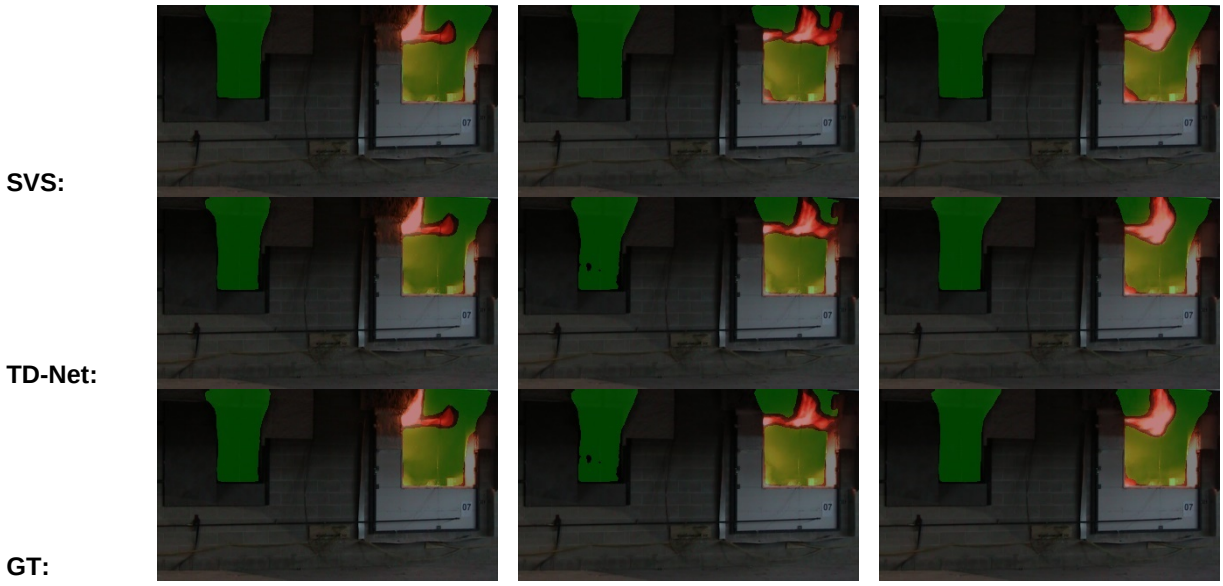


Figure 16: Images samples for accuracy, labels at left denotes the source of each row.

The performance of the methods studies is further analyzed adopting five different evaluation metrics, they are Intersection of Union (IoU), mean Intersection of Union (mIoU), Accuracy (Acc), mean Accuracy (mAcc), and Speed scores.

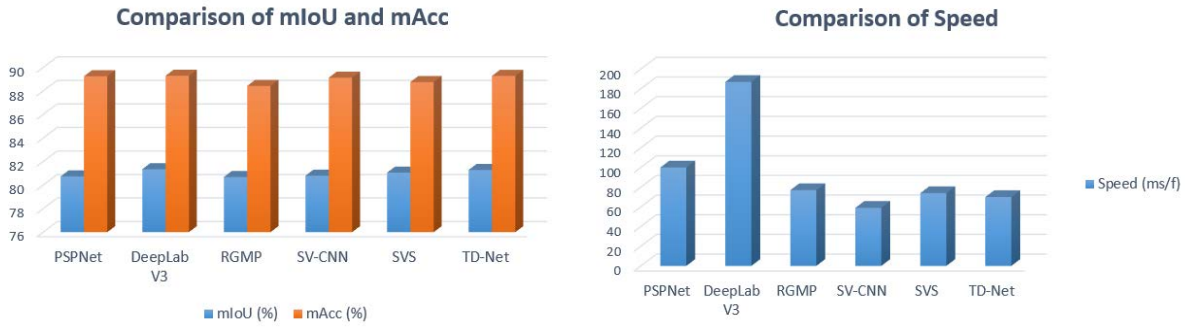
IoU and mIoU are popular metrics for segmentation tasks. They are both defined as the ratio of intersection and union of ground truth and predictions, while mIoU evaluates several classes. They are defined as the ratio of area size that has successful segmentation.

Acc and mAcc are defined as the accuracy in pixel level, while mAcc evaluates on several classes. In detail, it is the ratio of the number of pixels.

Compared with IoU and mIoU, Acc and mAcc calculate the accuracy on the pixel level, which could be a lot better for comparison among different classes having a big difference in the area. Besides, Acc and mAcc metrics only measure the impact of False Positive (FP). For example, IoU and mIoU calculate both FP and False Negative (FN).

Unlike the metrics listed above, speed is a unique evaluation metric for video semantic segmentation. It measures the ability of models to provide real-time and instant segmentation results. It is defined as the ratio of the number of frames and the time for processing.

The results from the comparison based on mIoU, mAcc, and Speed are shown in the figure below. TD-Net shows the dominant performance both in mIoU and mAcc metrics. However, some methods demonstrate better performance than TD-Net in specific metrics (e.g., DeepLab V3 showing higher mIoU scores than TD-Net), but TD-Net has the best overall performance, both accuracy and speed.



a) b)

Figure 17: Quantitatively study for methods on the FS Segmentation dataset. a) Comparison of mIoU and mAcc. b) Comparison of Speed.

Figure 18 shows the extended information for flame and smoke segmentation. Although calculated for different classes (smoke or fire), TD-Net resulted in excellent performance depicting the balance between the accuracy and speed for segmentation.

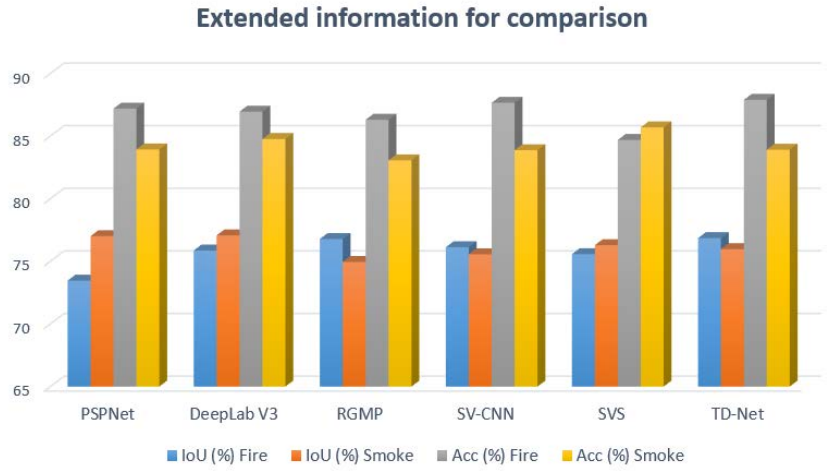


Figure 18: Extended information of quantitatively study for methods on FS Segmentation dataset.

4.2.3 Video Prediction Module

For the quantitative evaluation of the Video Prediction Module, two different metrics are introduced. They are the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM). PSNR is one of the most popular metrics in image conversion tasks as it has reconstruction loss of images. It is defined as the equation below.

$$PSNR = 10 \times log_{10}\left(\frac{L^2}{\frac{1}{N}\sum_{i=1}^{N}\left(I_i - \hat{I}_i\right)^2}\right) \quad \text{eq. 5}$$

Where I_i is the GT image.
$\hat{I}_i$ is the result of conversion or reconstruction.
N is the number of pixels in them.
L is the maximum pixel value. It is measured in dB via the log_{10} function.

For the quality, the higher the $PSNR$ is, the better quality of images is.

On the other hand, SSIM measures the structural similarity between images in terms of independent comparison with luminance, contrast, and structures from the Hue Saturation Value (HSV) color space. It is defined as the equation below.

$$SSIM = \frac{2(\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad \text{eq. 6}$$

Where $\mu_x$and μ_y are local means for image x and image y.
σ_x and σ_y are the standard deviation of image x and image y.
σ_{xy} is the cross-covariance between them. It is ranged in $(0,1)$.

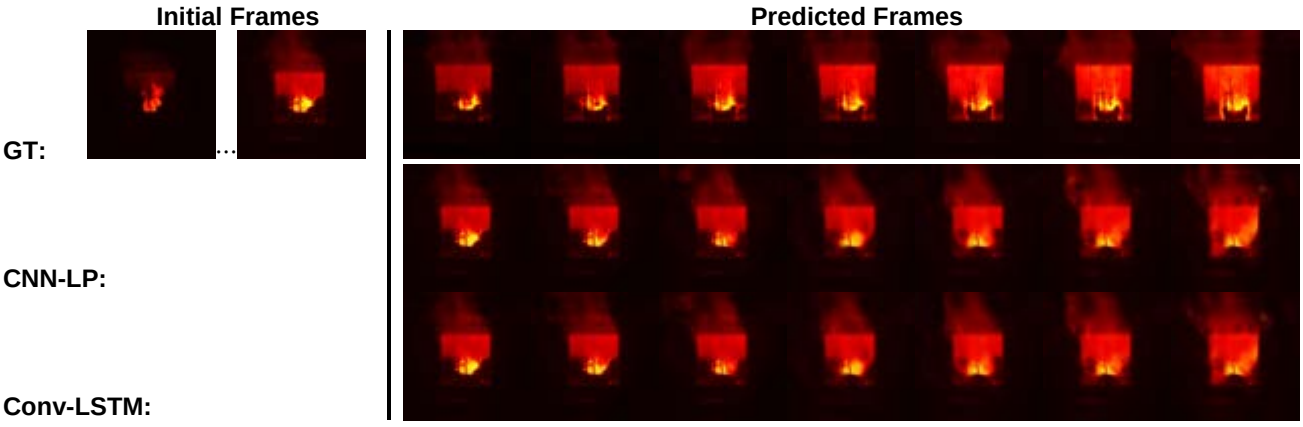
For the quality, the higher the $SSIM$ is, the better the quality of images is.

To conclude, PSNR is a metric that counts on the Mean Square Error (MSE) of pixel-level, indicating that a high PSNR score would make the generated images similar to the GT in corresponding pixel values; however, the visual perception is not guaranteed. SSIM is similar to the evaluation system of human vision. So, a high SSIM score guarantees that the generated images and GT are visibly similar in human eyes. To some extent, these two evaluation metrics have a complementary relationship, which is also why both of them are chosen in our evaluation.

With those metrics, the video prediction methods, as listed in **Table 8**, are evaluated with the FSVP dataset.

Table 8: A comparison of the structure and components of algorithms for the Video Prediction Module.

Name of methods	Structure Basis
CNN-LP	Convolutional Neural Network
Conv-LSTM	Recurrent Neural Network
SVVP	Variational Autoencoder
AMC-GAN	Generative Adversarial Network
SAVP	Variational Autoencoder **&** Generative Adversarial Network



SVVP:

AMC-GAN:

SAVP:

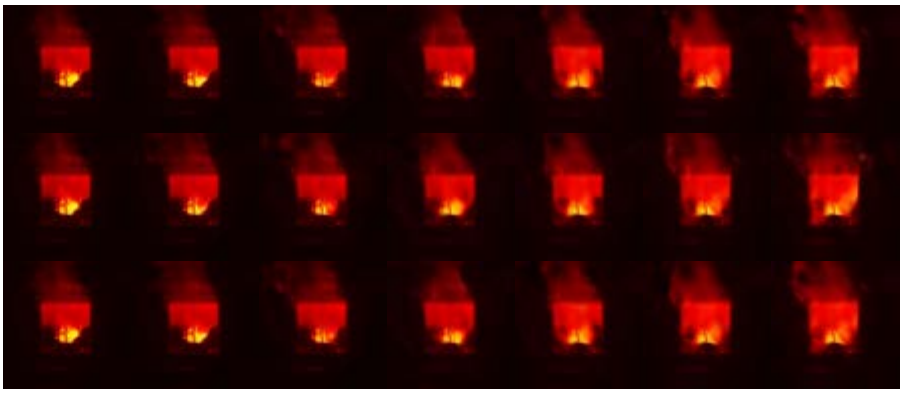


Figure 19: Samples of predicted images, the label at left denotes the source of each row.

The qualitative results from the evaluation study are shown in **Figure 19**. The quantitative evaluation results in terms of PSNR and SSIM are plotted in **Figure 20**.

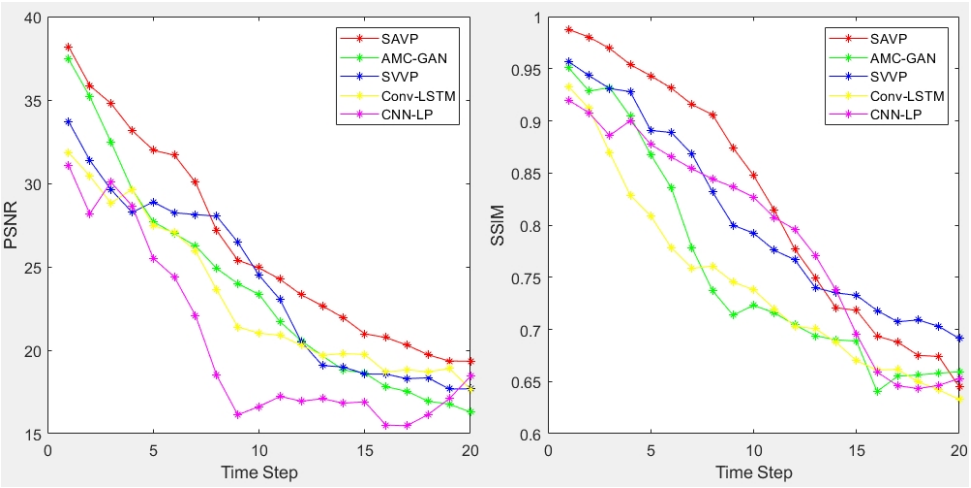


Figure 20: Plots of PSNR and SSIM scores with prediction time variation.

Overall, in the evaluation of the FSVP-IR dataset, SAVP is the best performer among the methods tested, and it delivers a better shape of the upper hot layer and flame area, while other methods fail on one aspect. For example, AMC-GAN brought a good prediction for the background and upper layers while its prediction for the flame area was poor.

As for the quantitative study on PSNR and SSIM scores, it somehow proves that the visual results from the qualitative study can be considered as quick evaluation methods. Moreover, SAVP shows leading performance among them.

4.3 Evaluation of Entire System

The entire system's performance is evaluated for flashover prediction with the 8 fire cases in the FP dataset. The raw performance of our system over the 8 cases is presented in **Figure 21**, which plots the prediction time of flashover by our system (in the gray bar), the offset between the real flashover time and prediction (in yellow bar), and the forecast time (i.e., the ability to predict flashover in advance, in the dark blue bar).

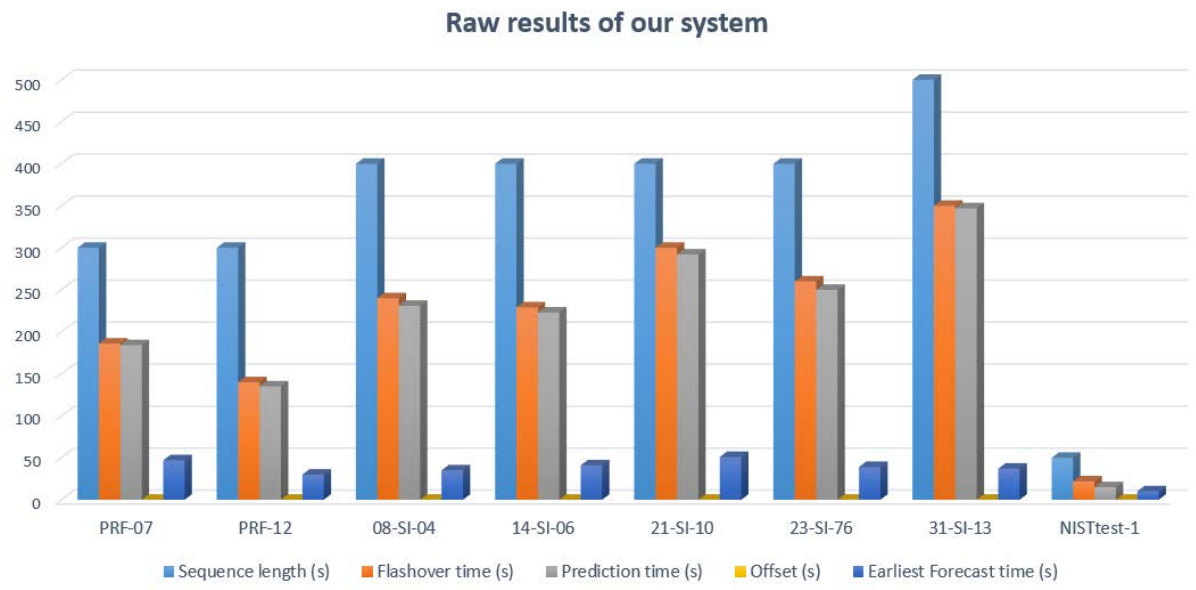


Figure 21: Raw statistics of flashover prediction performance of our system on the FP dataset.

In **Figure 21**, our system results in accurate prediction and early forecast in all 8 cases. In fact, accurate and early warning is crucial to firefighters responding to compartment fires. Among all the cases, the largest offset comes from the evaluation of the case of NIST test-1 since the fire scene was different from the other scenes from the NRC fire tests, and the system's sub-modules, including the Color2IR, were not trained with such scene including corresponding IR scene (due to the absence of the IR data). The view angle of the NIST Test-1 contained illumination both from the fire and lighting fixture in the room, for which our system is not yet trained.

Figure 22 shows a real-time analysis and prediction demo snapshot of the system. It presents several system components, including IR conversion, video semantic segmentation, predictions of visual and IR frames for the next few seconds, as well as statistical analysis of smoke and flame temperatures from the segmentation. The judgment of flashover occurrence is made based on the fusion of the predicted future frames and the temperature analysis among them. The system can provide a warning and *Estimated Time Arrival (ETA)* for the flashover in real-time.

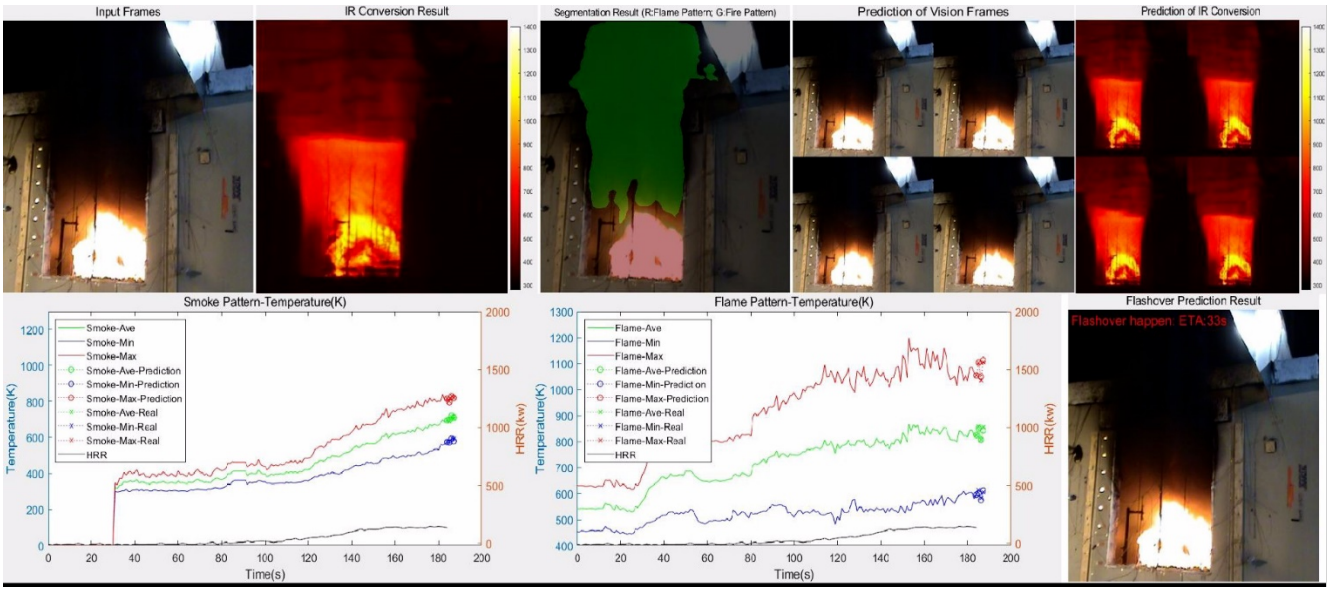


Figure 22: Sample of flashover prediction demo by our system.

Furthermore, new evaluation metrics are proposed for quantitative evaluations of flashover prediction enabling comparisons of different models with various cases. Applying the concept of the observation ratio widely used in the field of action prediction, a new evaluation metric of observation ratio for flashover prediction (r_f) is proposed. It is defined as the equation below;

$$r_f = \frac{t_c}{t_F} \qquad \text{eq. 7}$$

Where r_f is the observation ratio defined in flashover prediction.
t_c and t_F are the current time and the ground truth of flashover occurrence, shown in **Figure 23**.

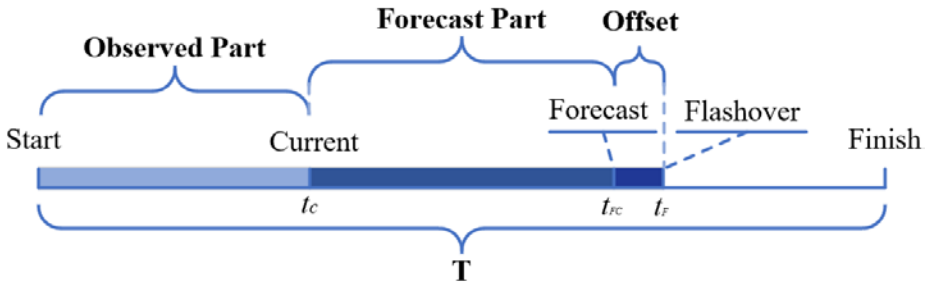


Figure 23: An illustration of time and period in a sequence of flashover predictions.

Accuracy (Acc) aims to measure the ability of a flashover prediction system in producing binary prediction (that is, happen or not) of flashover occurrence. Here, we use an average of all accuracy. It would be regarded as a successful prediction as long as the flashover prediction system identifies a happening of flashover in the future. Accuracy could be combined with observation ratio r_f to give evaluation over the predictions made with different observed frames in a sequence. Thus, the accuracy of flashover prediction would be measured at different observation ratio r_f.

Similar to the idea of accuracy at different observation ratio r_f, a new Forecast Accuracy (*FA*) score is formulated. Unlike Acc, *FA* aims to measure the system's ability to predict flashover in terms of the time

forecast of flashover occurrence. It would be regarded as a successful forecast only when the time forecasted by a flashover prediction system matches the real happening time of flashover. It could be formulated as equation 8.

$$FA = 1 - \frac{|t_F - t_{FC}|_{abs}}{t_F} \quad \text{eq. 8}$$

Where t_{FC} is the predicted time of flashover, shown in **Figure 23**.
t_F is the real time of flashover, shown in **Figure 23**.

FA with observation ratio r_f would allow evaluations of different models can be achieved without the influence of sequence lengths.

Finally, using the two new evaluation metrics, the performance of our system is compared with other state-of-the-art flashover prediction systems proposed in recent years. The results are shown in **Table 9**.

Table 9: Comparison of flashover prediction performance with other models.

Name of model	Acc@r_f		FA@r_f	
	Acc@0.5	**Acc@1**	**FA@0.5**	**FA@1**
Dexters et al. [17]	-	0.91	-	-
Fliszkiewicz et al.[16]	-	0.6569	-	-
Yap et al. [20]	-	0.94	-	-
Lee et al. [51]	-	0.92	-	-
Fu et al. [18]	0.761	1	0.681	0.813
Yun et al. [19]	-	1	-	0.92
ours	0.875	1	0.813	0.94

Most of the models only provide accuracy at $r_f = 1$, which means those models conduct only a 'classification' task given the entire sequence. Some of them provided the forecast time and prediction time so that both accuracy and *FA* could be measured at different observation ratio r_f.

As shown in TABLE, the prediction performance of our system is in the top place among all other models in accuracy and FA at $r_f = 0.5$ or $r_f = 1$. While some methods might show comparable results at one particular metric (e.g., *Yun et al.* [19] got 0.92 in **FA@1**), our model shows a much powerful forecast ability for flashover as shown in metrics **Acc@0.5 and FA@0.5**. It points out that our system has a high flashover prediction accuracy and high flashover forecast ability.

5 Conclusions

A smart firefighting tool is developed to predict the flashover occurrence in compartment fire only based on visual images and videos. The tool adopts deep learning neural networks structuring 4 sub-modules for image conversion of vision data to thermal data, segmentation for smoke/flame, video predictions, and determination for flashover based on pre-defined fire science knowledge. This hybrid system using not only Deep Learning methods but also fire research knowledge showed successful flashover predictions.

Color2IR module, as the most important module in the system, adopts a novel deep neural network, DAGAN, which is capable of producing both foreground attention masks and background attention masks. The conversion to IR data performed a stable foreground conversion and a clear background with high quality. The other three modules also demonstrated the state-of-the-art performance in their individual tasks and fit specific conditions in flashover prediction. Instead of using image semantic segmentation models, a video semantic segmentation method of TD-Net is used for smoke/flame segmentation accuracy as well as processing speed. For the video prediction module, SAVP is used to take advantage of the visually plausible results from GAN and diverse output from VAE. Also, statistical models are used in applying fire experience and knowledge to improve prediction accuracy. To further make these sub-modules collaborate as a system, the parameters are optimized within each module to provide a real-time prediction of flashover and maintain high accuracy.

The hybrid system resulted in a promising performance on flashover prediction. The performance of the system was evaluated in comparison with other existing models using a new metric inspired by the action prediction evaluation. The overall comparison shows that our system delivers not only high accuracy in top-tier but also is capable of giving early and accurate forecasts at the same time.

Finally, some suggested future works are as follows;

- It is suggested to further evaluate the system and improve its design.
- In particular, the conversion to IR images would require evaluations and validation by training the conversion module with various data obtained from different types of IR cameras.
- It is also necessary to consider using IR images directly instead of using the converted IR images (i.e., due to lack of IR data) to improve the prediction quality of IR future frames.
- Instead of the hybrid approach, pure deep learning could be adopted in the system.
- It is suggested to build a universal dataset for benchmarking in the flashover prediction field.

6 References

[1] M.J. Karter, Fire Loss in the United States During, (1998).

[2] M.E. Bowyer, V. Miles, T.N. Baldwin, T.R. Hales, Preventing deaths and injuries of fire fighters during training exercises, (2016).

[3] C. Butler, S. Marsh, J.W. Domitrovich, J. %J J. of occupational Helmkamp, environmental hygiene, Wildland firefighter deaths in the United States: A comparison of existing surveillance systems, 14 (2017) 258–270.

[4] A.C.Y. Yuen, G.H. Yeoh, R. Alexander, M. %J C.S. in F.S. Cook, Fire scene reconstruction of a furnished compartment room in a house fire, 1 (2014) 29–35.

[5] F. Hobbyist, Flashover Fire Caught on GoPro, (2015). https://www.youtube.com/watch?v=k-3UCGGizgc.

[6] B. of A. Steven J. Avato Tobacco, Ceiling layer development, (2017). https://www.sciencedirect.com/topics/physics-and-astronomy/flashover.

[7] R.D. Peacock, P.A. Reneke, R.W. Bukowski, V. %J F.S.J. Babrauskas, Defining flashover for fire hazard calculations, 32 (1999) 331–345.

[8] W.D. Walton, P.H. Thomas, Y. Ohmiya, Estimating temperatures in compartment fires, in: SFPE Handbook of Fire Protection Engineering, Springer, 2016: pp. 996–1023.

[9] C. Nix, Do Thermal Imaging Cameras Help During a Flashover ?, Fire Apparatus & Emergency Equipment. 21 (2016).

[10] B.A. Starnes, Thermal Imaging Cameras in the Fire Service : Asset or Detriment ? You Decide, Fire Apparatus & Emergency Equipment. (2018).

[11] NIST, Fire Dynamics, (2018). https://www.nist.gov/el/fire-research-division-73300/firegov-fire-service/fire-dynamics#:~:text=Heat%20Release%20Rate%20(HRR)%20is,to%20one%20Joule%20per%20second.

[12] T. Harmathy, T. Lie, Fire test standard in the light of fire research, in: Fire Test Performance, ASTM International, 1970.

[13] V. %J F.T. Babrauskas, Estimating room flashover potential, 16 (1980) 94–103.

[14] M. Beshir, Y. Wang, F. Centeno, R. Hadden, S. Welch, D. %J F.S.J. Rush, Semi-empirical model for estimating the heat release rate required for flashover in compartments with thermally-thin boundaries and ultra-fast fires, (2020) 103124.

[15] D. Cyganski, R.J. Duckworth, K.A. Notarianni, Development of a portable flashover predictor (Fire-ground environment sensor system), Worcester Polytechnic Institute, 2010.

[16] M. Fliszkiewicz, A. Krasuski, K. Krenski, Evaluation of a Heat Release Rate based on Massively Generated Simulations and Machine Learning Approach, in: FedCSIS (Position Papers), 2014: pp. 45–52.

[17] A. Dexters, R.R. Leisted, R. van Coile, S. Welch, G. %J F. Jomaas, Materials, Testing for knowledge: Application of machine learning techniques for prediction of flashover in a 1/5 scale ISO 13784 - 1 enclosure, (2020).

[18] E.Y. Fu, W.C. Tam, J. Wang, R. Peacock, P. Reneke, G. Ngai, H.V. Leong, T. Cleary, Predicting Flashover Occurrence using Surrogate Temperature Data, (2021).

[19] K. Yun, J. Bustos, T. %J E.I. Lu, Predicting rapid fire growth (flashover) using conditional generative adversarial networks, 2018 (2018) 127-1-127–4.

[20] K.S. Yap, C.P.L.E.W.M. Lee, J.M. Saleh, Development and Application of An Enhanced ART-Based Neural Network, in: The International Conference on Man-Machine Systems, 2009.

[21] W.C. Tam, E.Y. Fu, R. Peacock, P. Reneke, J. Wang, J. Li, T. %J F.T. Cleary, Generating Synthetic Sensor Data to Facilitate Machine Learning Paradigm for Prediction of Building Fire Hazard, (2020) 1–22.

[22] T. Buffington, J.-M. Cabrera, A. Kurzawski, O.A. %J F.T. Ezekoye, Deep-Learning Emulators of Transient Compartment Fire Simulations for Inverse Problems and Room-Scale Calorimetry, (2020) 1–27.

[23] K. Yun, K. Yu, J. Osborne, S. Eldin, L. Nguyen, A. Huyen, T. Lu, Improved visible to IR image transformation using synthetic data augmentation with cycle-consistent adversarial networks, in: Pattern Recognition and Tracking XXX, International Society for Optics and Photonics, 2019: p. 1099502.

[24] Z. Wang, C. Song, T. %J E. Chen, Deep learning based monitoring of furnace combustion state and measurement of heat release rate, 131 (2017) 106–112.

[25] L. Kou, X. Wang, H. Zhang, R. Yang, Y. Liu, Inverse Model for Fire Heat Release Rate Using Deep Neural Networks, in: ASME 2020 Heat Transfer Summer Conference Collocated with the ASME 2020 Fluids Engineering Division Summer Meeting and the ASME 2020 18th International Conference on Nanochannels, Microchannels, and Minichannels, American Society of Mechanical Engineers Digital Collection, 2020.

[26] K. Yun, T. Lu, E. Chow, Occluded object reconstruction for first responders with augmented reality glasses using conditional generative adversarial networks, in: Pattern Recognition and Tracking XXIX, International Society for Optics and Photonics, 2018: p. 106490T.

[27] A. Starnes, Thermal Imaging Cameras in the Fire Service: Asset or Detriment? Your Decide, (2018). https://www.fireapparatusmagazine.com/technology/thermal-imaging-cameras-in-the-fire-service-asset-or-detriment-you-decide/#gref.

[28] V. Bianco, M. Paturzo, M. Locatelli, E. Pugliese, A. Finizio, A. Pelagotti, P. Poggi, L. Miccio, R. Meucci, P. Ferraro, Seeing People Moving Behind Smoke and Flames by Lensless Digital Holography at Long Infrared Wavelength, (n.d.).

[29] J.-H. Cho, C.-H. Hwang, J. Kim, S. %J J. of the K.S. of S. Lee, Sensitivity analysis of FDS results for the input uncertainty of fire heat release rate, 31 (2016) 25–32.

[30] F. Zhang, P. Habisreuther, H. Bockhorn, H. Nawroth, C.O. %J A.A. united with A. Paschereit, On prediction of combustion generated noise with the turbulent heat release rate, 99 (2013) 940–951.

[31] J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017: pp. 2223–2232.

[32] H. Tang, D. Xu, N. Sebe, Y. Yan, Attention-guided generative adversarial networks for unsupervised image-to-image translation, in: 2019 International Joint Conference on Neural Networks (IJCNN), IEEE, 2019: pp. 1–8.

[33] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: pp. 2881–2890.

[34] L.-C. Chen, G. Papandreou, F. Schroff, H. %J arXiv preprint arXiv:. 05587 Adam, Rethinking atrous convolution for semantic image segmentation, (2017).

[35] S.W. Oh, J.-Y. Lee, K. Sunkavalli, S.J. Kim, Fast video object segmentation by reference-guided mask propagation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: pp. 7376–7385.

[36] R. Gadde, V. Jampani, P. v Gehler, Semantic video cnns through representation warping, in: Proceedings of the IEEE International Conference on Computer Vision, 2017: pp. 4453–4462.

[37] D. Nilsson, C. Sminchisescu, Semantic video segmentation by gated recurrent flow propagation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: pp. 6819–6828.

[38] P. Hu, F. Caba, O. Wang, Z. Lin, S. Sclaroff, F. Perazzi, Temporally distributed networks for fast video semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: pp. 8818–8827.

[39] E. Denton, S. Chintala, A. Szlam, R. %J arXiv preprint arXiv:. 05751 Fergus, Deep generative image models using a laplacian pyramid of adversarial networks, (2015).

[40] J. Cheng, L. Dong, M. %J arXiv preprint arXiv:. 06733 Lapata, Long short-term memory-networks for machine reading, (2016).

[41] Y. Jang, G. Kim, Y. Song, Video prediction with appearance and motion conditions, in: International Conference on Machine Learning, PMLR, 2018: pp. 2225–2234.

[42] M. Babaeizadeh, C. Finn, D. Erhan, R.H. Campbell, S. %J arXiv preprint arXiv:. 11252 Levine, Stochastic variational video prediction, (2017).

[43] A.X. Lee, R. Zhang, F. Ebert, P. Abbeel, C. Finn, S. %J arXiv preprint arXiv:. 01523 Levine, Stochastic adversarial video prediction, (2018).

[44] T. Kim, M. Cha, H. Kim, J.K. Lee, J. Kim, Learning to discover cross-domain relations with generative adversarial networks, in: International Conference on Machine Learning, PMLR, 2017: pp. 1857–1865.

[45] Z. Yi, H. Zhang, P. Tan, M. Gong, Dualgan: Unsupervised dual learning for image-to-image translation, in: Proceedings of the IEEE International Conference on Computer Vision, 2017: pp. 2849–2857.

[46] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. %J arXiv preprint arXiv:. 03762 Polosukhin, Attention is all you need, (2017).

[47] G. Hinton, O. Vinyals, J. %J arXiv preprint arXiv:. 02531 Dean, Distilling the knowledge in a neural network, (2015).

[48] A. Bwalya, E. Gibbs, G. Lougheed, A. Kashef, Characterization of fires in multi-suite residential dwellings: final project report: part 1-A compilation of post-flashover room fire test data, National Research Council of Canada, 2014. https://doi.org/10.4224/21275340.

[49] OakRidgeFD, Flashover Demonstration, (2014). https://www.youtube.com/watch?v=BtMmymOxdjc&t=170s.

[50] R. and B. Bryant M., The NIST 20 MW Calorimetry Measurement System for Large-Fire Research, Technical Note (NIST TN) 2077. (2019). https://doi.org/https://doi.org/10.6028/NIST.TN.2077.

[51] E.W.M. Lee, Y.Y. Lee, C.P. Lim, C.Y. %J A. engineering informatics Tang, Application of a noisy data classification technique to determine the occurrence of flashover in compartment fires, 20 (2006) 213–222.

7 Appendix:

Designs of sub-modules

The prediction of flashover is complex. Other currently available methods are either limited to the success of synthetic simulation test data or limited to post-fire analysis. Thus, it often becomes difficult to make real-time predictions. In order to better solve this problem, we use a modular design in our system. Each sub-module has its specific function and character. The different sub-modules combine the results through image fusion or mathematical analysis to generate the entire system's flashover prediction. This section provides detailed descriptions of each sub-module, including parameter settings in loss function for our deep neural networks.

As one of the most crucial sub-modules in the system, Color2IR Conversion aims to provide corresponding IR images that could tell the temperature of each pixel from a visual image captured from a standard camera that could be taken into fire rescue with firefighters. The input videos would be cut into independent frames in this module and processed as a single unit. Besides, it is a kind of cross-domain image transfer task in the Computer Vision field as the images of input and output are from different types. For the IR conversion, a novel structure of deep neural networks: Dual-Attention GAN (DAGAN), is designed. The architecture of DAGAN is illustrated in **Figure 7** in Chapter 3, and it is inspired by the success of CycleGAN in un-paired image conversion. The input images of DAGAN are the sequence cut from the visual videos of fire scenes in the dataset, which is denoted as x in **Figure 7**. There is no restriction for the Frame Per Second (FPS), as someone might have relatively low computational capability hardware and would like to convert them for real-time usage. The input x will be fed into our generator G_1, which consists of an encoder G_{E1} and two mask generators: G_{C1} and G_{A1}. G_{E1} is a parameter-sharing encoder which could generate low-level feature maps. While G_{C1} is a content mask generator that could generate a set of masks C_x^f, which contains sets of the content feature captured from the encoder G_{E1}.

Unlike G_{C1}, G_{A1} is a generator for attention mechanism providing attention-level feature maps from the encoded information. The direct output of G_{A1} is processed by a Softmax activation function to change the scope of mapping. Then, it would produce two types of attention masks: A_x^f and A_x^b. We used the definition of the self-attention mechanism proposed by others to build a simple but super effective one to be used in our system. To be specific, the foreground attention mask and background attention mask enable DAGAN to differentiate the foreground and background images, which could help solve background blurry and foreground color drift. In total, the two attention mask generators would produce a total number of N attention masks: $[\{A_x^f\}_{f=1}^{N-1}, A_x^b]$ in the generation process. There is only one background attention mask A_x^b , and there is a set of $(N-1)$ foreground masks $\{A_x^f\}_{f=1}^{N-1}$. This is because the foreground is rather important than the background information to provide contextual information. The amount of information in the foreground is also more than that of the background, as defined by the attention mask generator. This also allows the process of foreground context and background context independently.

Then, the foreground information and background information extracted from input x will be processed independently. For the foreground information, foreground attention masks $\{A_x^f\}_{f=1}^{N-1}$ would be used to generate the foreground content by combing the set of content masks in earlier steps. At the same time, the background attention mask would help keep a clean and tidy background of generated images by combing it with the original input x. The final generated image $G_1(x)$ would be the sum of the two content images selected from extracted feature maps by our attention mechanism. It is, which could be calculated as the formula below.

$$G_1(x) = \sum_{f=1}^{N-1} \left(A_x^f \times C_x^f\right) + x \times A_x^b \quad \text{eq. 9}$$

That is the end of the generation loop and also the start of the reconstruction loop. The basic idea of loop structure is that we should be back to where we start if we walk in a loop. It also works for the image conversions as the loop conversion should make the reconstruction back into the same domain as the input x. The reconstruction is inverse to the generation process in structure, while the training process would be independent. Let

$$y = G_1(x) \quad \text{eq. 10}$$

And we will have an equation that describes the calculation of the corresponding attention masks and content masks in the reconstruction process in a way that is similar to the generation process. That equation is shown as equation 12 below.

$$G_2(y) = \sum_{f=1}^{N-1} \left(A_y^f \times C_y^f\right) + y \times A_y^b \quad \text{eq. 11}$$

The only difference between them is that the foreground and background areas for x and y would be different as they are from different domains.

In this way, a closed-loop for the DAGAN process could be finally formed a closed-loop for the DAGAN process, starting from the input x to the reconstruction of $G_2(y)$ or, in other words, $G_2(G_1(x))$ if we take equation 9 into it. The process of the loop is shown below.

$$x \to G_1(x) \to G_2(G_1(x)) \approx x \quad \text{eq. 12}$$

Where x stands for the input image in the vision domain.
G_1 and G_2 are generators mentioned above.

Then, if we could bring the process denoted in equation 11 into it, the detailed calculation process should be:

$$G_2(G_1(x)) = \sum_{f=1}^{N-1} \left(A_y^f \times C_y^f\right) + G_1(x) \times A_y^b \approx x \quad \text{eq. 13}$$

For another direction of the loop that starts from the image in the IR domain:

$$y \to G_1(y) \to G_2(G_1(y)) \approx y$$
$$G_2(G_1(y)) = \sum_{f=1}^{N-1} \left(A_x^f \times C_x^f\right) + G_2(y) \times A_x^b \approx y \quad \text{eq. 14}$$

Where y stands for the input image in the IR domain.
G_1 and G_2 are generators mentioned above.

In addition, for the discriminators in DAGAN, there are two types of discriminators. The first type is the discriminators D_{Y1} and D_{Y2}, which are vanilla discriminators used to distinguish the generated images $G_1(x)$ and real images y or $G_2(y)$ and x.

Besides, we also proposed a brand-new type of discriminator, which is the second type of discriminator. They are D_{YA1} and D_{YA2}, which are attention discriminators that capable of taking both images and feature maps

generated by the attention mask generator as input. As we have generated a total number of N attention masks in the generation process. Let

$$A_x = \left[\{A_x^f\}_{f=1}^{N-1}, A_x^b\right] \quad \text{eq. 15}$$

And we make a concatenation of it with the generated images $G_1(x)$ And real images y. So, it should be

$$M_{1y} = [A_x, y], \quad M_{1x} = [A_x, G_1(x)] \quad \text{eq. 16}$$

After that, the attention discriminator would take M_{1y} or M_{1x} as input and will also try to distinguish the generated images with attention masks M_{1x} and the real images with attention masks M_{1y}.

Building a brand-new neural network is just half of success, even if it has excellent design. Another part of our contribution to DAGAN is the design of loss functions for it. There are several parts of loss function for DAGAN, and we are going to introduce them one by one.

The first part of the loss function is an adversarial loss that same as vanilla GAN, which is formulated as the equation below.

$$\mathcal{L}_{GAN}(G_1, D_{Y1}) = \mathbb{E}_{y\sim p_{data}(y)}[\log(D_{Y1}(y))] + \mathbb{E}_{x\sim p_{data}(x)}\left[\log\left(1 - D_{Y1}\left(G_1(x)\right)\right)\right] \quad \text{eq. 17}$$

In this equation, generator G aims to minimize the adversarial loss: $\mathcal{L}_{GAN}(G_1, D_Y)$, while D_{Y1} tries to maximize it at the same time. The target of G_1 is to generate an image $G_1(x)$ that is similar to the images from domain Y, while D_{Y1} aims to distinguish between the generated images $G_1(x)$ and the real images y.

Similar to the relationship between equation 12 and equation 14 that lasted above, there is a similar process for the generator G_2 and discriminator D_{Y2}. Their adversarial loss is defined as the equation below.

$$\mathcal{L}_{GAN}(G_2, D_{Y2}) = \mathbb{E}_{x\sim p_{data}(x)}[\log(D_{Y2}(x))] + \mathbb{E}_{y\sim p_{data}(y)}\left[\log\left(1 - D_{Y2}\left(G_2(y)\right)\right)\right] \quad \text{eq. 18}$$

Where D_{Y2} tries to distinguish between the generated image $G_2(y)$ and the real image x.

As a network with the loop structure, there is also a loop loss or cycle loss in DAGAN between original input x and reconstruction result $G_1(G_2(x))$. The cycle-consistency loss in DAGAN is formulated as the equation below.

$$\mathcal{L}_{Cycle}(G_1, G_2) = \mathbb{E}_{x\sim p_{data}(x)}[\|G_2(G_1(x) - x)\|_1] + \mathbb{E}_{y\sim p_{data}(y)}[\|G_1(G_2(y) - y)\|_1] \quad \text{eq. 19}$$

Where the reconstruction result $G_2\left(G_1(x)\right)$ is closely related to input x in pixel level.
$G_1\left(G_2(y)\right)$ should match the input of y under similar circumstances.
Here, the L1 loss is used to measure the image difference in pixel level.

Besides, we also use pixel loss in DAGAN in order to constrain the generator without discriminator information at the pixel level. It could be formulated as follow.

$$\mathcal{L}_{Pixel}(G_1, G_2) = \mathbb{E}_{x\sim p_{data}(x)}[\|G_1(x) - x\|_1] + \mathbb{E}_{y\sim p_{data}(y)}[\|G_2(x) - x\|_1] \quad \text{eq. 20}$$

Here, we also use the *L*1 loss for pixel-level measurement. It is also called identity loss in CycleGAN.

Another type of loss that we also introduce is Attention Adversarial loss in AGGAN. We brought the idea of the formation of adversarial loss shown in equations 15 and 16. Also, We made a few modifications so that it could fit the dual-attention mechanism in DAGAN. While the original idea is similar to the formation of adversarial loss

shown in equations 12 and 13, we also made a modification to fit the dual-attention mechanism in DAGAN. Thus, this loss comes from the attention discriminator D_{YA1} and D_{YA2} and the generator G_1 and G_2. It could be formulated as the equation below.

$$\mathcal{L}_{AGAN}(G_1, D_{YA1}) = \mathbb{E}_{y\sim p_{data}(y)}\left[\log(D_{YA1}(M_{1y}))\right] + \mathbb{E}_{x\sim p_{data}(x)}\left[\log(1 - D_{YA1}(M_{1x}))\right] \quad \text{eq. 21}$$

Where $M_{1y} = [A_x, y]$, $M_{1x} = [A_x, G_1(x)]$ as illustrated in equation 16.

This loss helps form a stable attention mask in the training process without any annotations on the image pairs in the training set.

Furthermore, we also propose a pure attention loss to improve the stability and performance of attention masks. This loss only uses the information in generated attention masks that aims to solve the problem that attention masks saturation problem. The attention loss is shown in the equation below.

$$\mathcal{L}_{At}(A_x) = \sum_{w,h=1}^{W,H} |A_x(w+1,h,c) - A_x(w,h,c)| + |A_x(w,h+1,c) - A_x(w,h,c)| \quad \text{eq. 22}$$

Where A_x is the attention mask for calculation.
W and H is the width and height dimension of A_x.

Finally, we could finally get our loss function for DAGAN optimization by piecing them all together with weights. The loss function of DAGAN is formulated as follows;

$$\begin{aligned}\mathcal{L}_{DAGAN} &= \lambda_{Cycle} \times \mathcal{L}_{Cycle} + \lambda_{Pixel} \times \mathcal{L}_{Pixel} + \lambda_{At} \times \mathcal{L}_{At} + \lambda_{GAN} \times (\mathcal{L}_{AGAN} + \mathcal{L}_{GAN}) \\ &= \lambda_{Cycle} \times \mathcal{L}_{Cycle}(G_1, G_2) \\ &+\lambda_{Pixel} \times \mathcal{L}_{Pixel}(G_1, G_2) \\ +\lambda_{GAN} \times &\left(\mathcal{L}_{GAN}(G_1, D_{Y1}) + \mathcal{L}_{GAN}(G_2, D_{Y2}) + \mathcal{L}_{AGAN}(G_1, D_{YA1}) + \mathcal{L}_{AGAN}(G_2, D_{YA2})\right) \\ &+\lambda_{At} \times \left(\mathcal{L}_{At}(A_x) + \mathcal{L}_{At}(A_y)\right)\end{aligned} \quad \text{eq. 23}$$

Where $\lambda_{Cycle} = 10$.
$\lambda_{Pixel} = 1$.
$\lambda_{GAN} = 0.5$.
$\lambda_{At} = 1 \times 10^{-6}$ in our setup.

In this way, a closed-loop for the DAGAN process is finally formed, starting from the input x to the reconstruction of G_2.

There is another sub-module that takes the original input images, which is **Video Semantic Segmentation** Module. This is used to generate semantic information for fire scenes. Firefighters usually face a super dark room with hot gases around them without good lighting conditions, which will dramatically increase the difficulty of flame and smoke recognition. Thus, one of the most demanding needs for our system is real-time video semantic segmentation. The system adopts TD-Net [38], a type of neural network for video semantic segmentation made of efficient networks, smaller than most of the existing networks. It could also provide segmentation results as accurate as those 'big' networks. The basic idea of TD-Net is Group Convolution, which extracts features with separated filter groups instead of only one guaranteed model parallelization and representations. The sub-networks design and Attention Propagation Module (APM) contribute to fast and consistent segmentation.

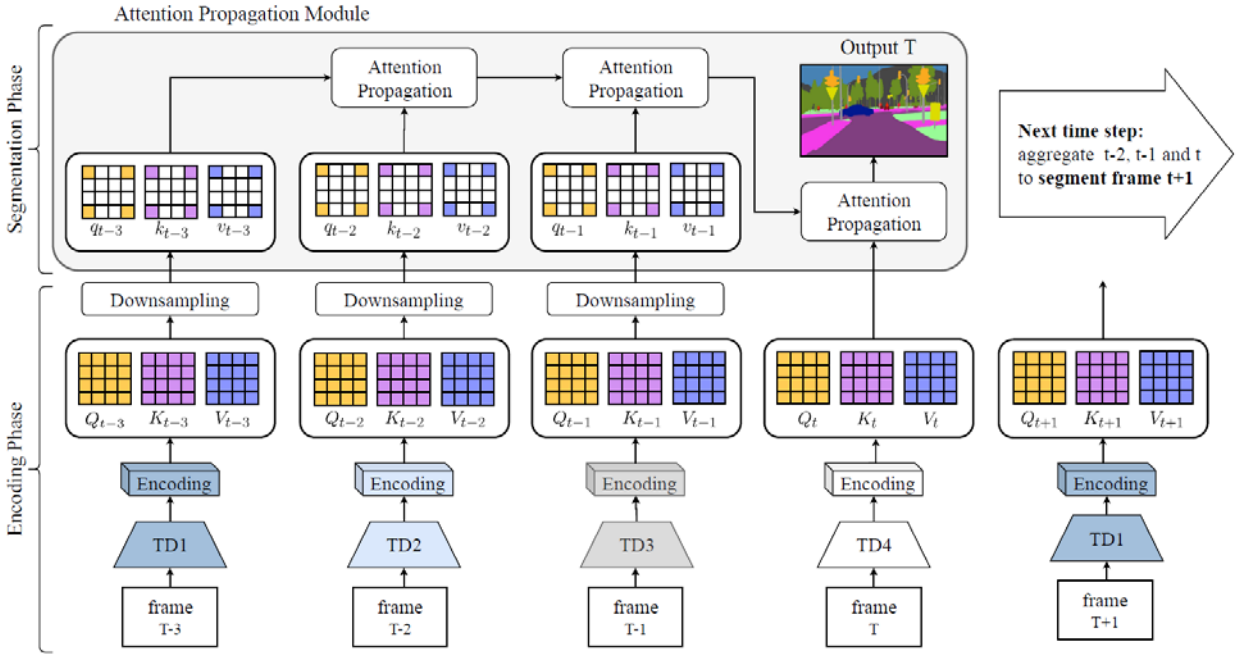


Figure 24: An illustration of the detailed structure of TD-Net, from [38]

The detailed structure of TD-Net is shown in **Figure 24**. The first phase of TD-Net conducts the Encoding Phase. The network generates feature maps, which are path-specific, and Query and Key maps for cross-frames correlating between pixels. After that, it calculates the attention from Value (V), Query (Q), and Key (K) as a self-attention mechanism formulated as the equation below.

$$\boldsymbol{Aff}_p = Softmax\left(\frac{Q_t K_p^T}{\sqrt{d_k}}\right) \quad \text{eq. 24}$$

Where d_k is the dimension of Q_t and K_p.

Then, those feature maps are merged together at current frames, and previous $(m-1)$ frames as follow:

$$V_t' = V_t + \sum_{p=t-m+1}^{t-1} \phi\left(\boldsymbol{Aff}_p V_p\right) \quad \text{eq. 25}$$

Those feature maps could effectively capture non-local correlations between pixels across frames with the help of this self-attention mechanism. After that, there is a downsampling process to reduce the computation costs.

The second phase of TD -Net is the segmentation phase, which includes a propagation approach that measures the attention of m neighboring frames, which is formulated as the following.

$$v_p' = \phi\left(Softmax\left(\frac{q_t k_p^T}{\sqrt{d_k}}\right) v_{p-1}'\right) + v_p \quad \text{eq. 26}$$

Where q_t is the downsampled version of Q_t.
k_p is the downsampled version of K_p.
v_p is the downsampled version of V_p.

Then, it finally computes the final feature representative at each time frame, computed as:

$$V_t' = \phi\left(Softmax\left(\frac{Q_t k_{t-1}^T}{\sqrt{d_k}}\right)v_{t-1}'\right) + V_t \qquad \text{eq. 27}$$

And the segmentation maps are generated by the equation as follows.

$$S_m = \pi_m(V_m') \qquad \text{eq. 28}$$

Where π_m is the final prediction layer of sub-networks m.

There is also a Grouped Knowledge Distillation mechanism in order to enhance the sub-feature maps in the full feature space. The loss function is illustrated in equation 26.

$$\mathcal{L} = CE\big(\pi_S(V_i', gt)\big) + \alpha \cdot KL(\pi_S(V_i')||\pi_T(\sum f)) + \beta \cdot KL(\pi_S(V_i)||\pi_T(f_i)) \qquad \text{eq. 29}$$

Where CE denotes $CrossEntrophy$ loss.
KL is the KL-divergence.
π_S is the prediction of student network.
π_T is that of teacher network.

In our system, we set the m, which is the number of sub-networks to 2.

After the Color2IR Conversion Module and Video Semantic Segmentation Module, **Video Prediction Module** runs the subsequent processing using the power of neural networks to provide reliable visual results for fire scenes. The module that is directly related to prediction purposes in our system is the Video Prediction Module.

For existing methods on video prediction tasks, generative models are the state-of-the-art methods now. Encoder-Decoder models provide predictions with diversity, and GANs models are capable of giving naturalistic predictions. Thus, a combination of them will promise a prediction with stochasticity as well as plausibility. Stochastic Adversarial Video Prediction (SAVP) is used in our system. An illustration of the detailed structure of the SAVP model is shown in **Figure 25**.

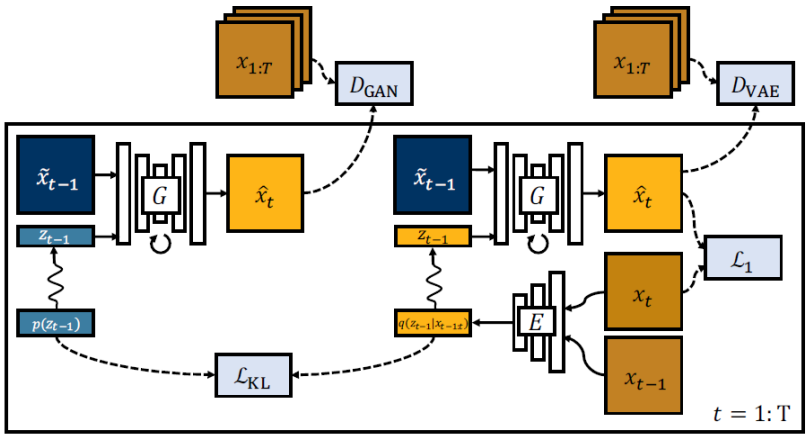


Figure 25: An illustration of the detailed structure of SAVP, from [42]

It consists of two parts. The first part is a Variation Autoencoder (VAE) that also acts as a generator. The generator G predicts the future frames with the previous ones $\tilde{x}_{t-1}$ and latent codes z_{t-1}, thus it actually specifies a distribution $p(x_t|x_{0:t-1}, z_{0:t-1})$, based on a fixed variance Laplacian distribution with mean as $\hat{x}_t = G(x_0, z_{0:t-1})$. For the VAE part, there is a conditional VAE which has a conditional encoder and decoder on the previous frames $\hat{x}_t$ or x_t. Then, they rewrite the reconstruction term to allow the backpropagation through the encoder. That term is formulated as follows.

$$\mathcal{L}_1(G, E) = \mathbb{E}_{x_{0:T}, z_t \sim E(x_{t:t+1})|_{t=0}^{T-1}} \left[\sum_{t=1}^{T} \left\| x_t - G\left(x_0, z_{0,t-1}\right) \right\|_1 \right] \quad \text{eq. 30}$$

Where $\hat{x}_t$ denotes reconstructed frames.
x_t is the ground truth frames.
z_t is the latent variables.

Besides, there is a regularization term for the encoder to approach the prior distribution. It is shown as equation 31.

$$\mathcal{L}_{KL}(E) = \mathbb{E}_{x_{0:T}} \left[\sum_{t=1}^{T} \mathcal{D}_{KL}(E(x_{t-1:t}) || p(z_{t-1})) \right] \quad \text{eq. 31}$$

Where *KL* is the KL-divergence.

So, for the optimization of VAE, it involves minimizing the objects listed above in equation 30 and equation 31. It is shown in the equation below.

$$G^* E^* = \arg\min_{G,E} \lambda_1 \mathcal{L}_1(G, E) + \lambda_{KL} \mathcal{L}_{KL}(E) \quad \text{eq. 32}$$

The second part is GAN, which is shown in the left part of **Figure 25**, where a generator G provides a prediction of future frames $\hat{x}_{1:T}$. The discriminator D distinguishes the generated frames $\hat{x}_{1:T}$ from the original ones $x_{1:T}$. Thus, the generator would be trained using binary cross-entropy loss, formulated as follow.

$$\mathcal{L}_{GAN}(G, D) = \mathbb{E}_{x_{1:T}}[\log D(x_{0:T-1})] + \mathbb{E}_{x_{1:T}, z_t \sim p(z_t)|_{t=0}^{T-1}} \left[\log\left(1 - D\left(G(x_0, z_{0:T-1})\right)\right) \right] \quad \text{eq. 33}$$

For the generator, it could be learned with an adversarial process, formulated as follow.

$$G^* = arg \min_G \max_D \mathcal{L}_{GAN}(G, D) \quad \text{eq. 34}$$

The optimization objective with VAE and GAN part could be put together for the final loss, shown in the equation below.

$$G^*, E^* = arg \min_{G,E} \max_{D, D^{VAE}} \lambda_1 \mathcal{L}_1(G, E) + \lambda_{KL} \mathcal{L}_{KL}(E) + \mathcal{L}_{GAN}(G, D) + \mathcal{L}_{GAN}^{VAE}(G, E, D^{VAE}) \quad \text{eq. 35}$$

For the detail of the structure, they use Conv-LSTM in generators G and the discriminator in SNGAN as discriminator D.

Another critical step in the present Flashover Prediction System is **Fire Knowledge Module**. While Deep learning is a popular choice for many research fields, yet a hybrid approach is taken in the present explorative study by employing Fire Knowledge Module. Many previous fire safety research studies show that linear mathematical models are still influential in processing conventional measurement data, such as temperature. Thus, taking a similar approach, the Analysis with Fire Knowledge Module in our Flashover Prediction System

is proposed to extract and process the data acquired from the input images using statistical analysis methods. There are several choices for parameters (e.g., temperature, HRR, and flame height) for the statistical and mathematical models in flashover analysis and prediction. As the temperature distribution information could be directly extracted from the output of the previous modules in our system, the temperature is chosen for a flashover criterion. Research has found the relationship between temperature and flashover. As widely accepted, a typical flashover happens when the upper layer of smoke in the room is approaching or above 600℃ for normal conditions in a typical room [11]. Temperature data are extracted from the input through the IR conversion and the segmentation of smoke and flame areas, and the data are analyzed for flashover occurrence.

In addition, for forecasting the onset of flashover, the system combines the prediction frames from the Video Prediction Module and extracts information both in the converted IR and visual domain. The detailed solution proposed for this problem is only producing a limited number of prediction frames, like 5 or 10 frames. It would make them not only visually plausible but also full of contextual information from the original input. Hence, it is called applicable predictions. Then, the system combines data and generates a graph of statistical information. Thus, it could be regarded as a tangent of specific points representing the original input, as shown in the figure below.

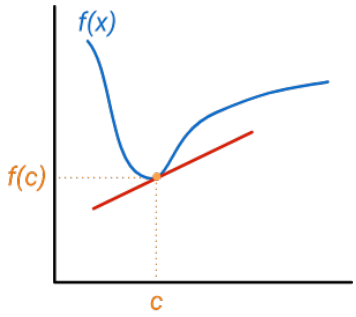


Figure 26: An illustration of applicable predictions and the temperature data curve.

Then, following the definition of derivatives, we could approximate the future point on the graph with current data and the tangent, formulated as equation 36.

$$f(c_f) = f(c) + \sigma \cdot (c_f - c) \qquad \text{eq. } 36$$

Where c is the point of original frames.
c_f denotes the points for the future.
σ is the tangent value.

Since the time domain of the statistical temperature graph is discrete, we need to link them together to form a continuous curve. The tangent is updated with every input frame. It could also help in preventing collapse problems in fluctuation points or spikes in the temperature curve, shown in **Figure 27**.

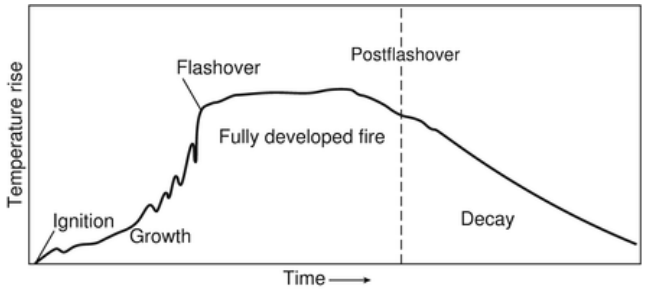


Figure 27: An illustration of temperature variation with time. The fluctuation point is between flashover and the growth stage of fire development.

Furthermore, locally weighted linear regression is introduced as the core mathematical model in this sub-module. It is a type of supervised, non-parametric learning algorithm. The difference between it and the ordinary linear regression model is that it does not learn a fixed set of parameters. It would contribute to better performance on the data curve with more fluctuation and spikes, which is typical for temperature data. is

For the FPS setting, we set the FPS in Analysis with Fire Knowledge module to 1, matching the settings in previous modules, which could reduce computation cost and remain precision analysis. In addition, the number of predicted frames is set to 3 to match the FPS for a better prediction result.

With the locally weighted linear regression and tangent prediction model set for our system, this sub-module could smoothly proceed the data from previous sub-modules and do the analysis and prediction of flashover with fire knowledge criterion of flashover.