

# Obstructed Camera View

Keep your camera view clear with our obstructed camera detection model.

## Overview

The obstructed camera detection model uses computer vision techniques to analyze the camera's video feed and identify if there is any obstruction present in the camera's field of view. The model utilizes deep learning techniques to learn the features of an unobstructed camera view and detects the presence of obstructions by analyzing the changes in the image features.

The obstructed camera detection model can operate in real-time, providing continuous monitoring of the camera's field of view. This model can detect various types of obstructions, including partial obstructions, and can provide an alert when an obstruction is detected.

## Vision AI based monitoring

The aim of an obstructed camera detection model is to detect whether a camera's field of view is obstructed or not. This model is designed to detect various types of obstructions such as fingers, tape, post-it notes, or other physical objects that might block the camera view. The obstructed camera detection model can be useful in different settings such as surveillance systems, video conferencing, or any other applications that require a clear and unobstructed camera view.

In summary, the obstructed camera detection model is a computer vision-based algorithm that uses deep learning techniques to detect obstructions in a camera's field of view. This model can be useful in various settings, providing real-time monitoring of camera views and alerting when obstructions are present.

## Model Details

### Dataset

The dataset consists of images and videos collected from various sources.

### Model

The model to detect obstructed camera event is in progress and it will be released soon.

## Scenario details

The business logic for this scenario is as follows:

- We use existing camera feeds from the premises to monitor whether the camera view is obstructed or not. We detect partial obstructions that may only partially block the camera view, such as fingers partially covering the lens.
- VisionAI system is able to run on edge devices. It uses camera feeds for processing.
- When an instance of obstructed camera is detected, an alert will be raised.

### Test now with online Web-Cam

To test this model & scenario, you can use the following steps:

- Install the visionai package from PyPI

```
$ pip install visionai
```

- Test the scenario from your local web-cam

```
$ visionai scenario test obstructed-camera-detection

Downloading models for scenario: obstructed-camera-detection
Model: obstructed-camera-detection:
https://workplaceos.blob.core.windows.net/models/yolov5s-obstructed-
camera-detection/yolov5s-obstructed-camera-detection-0.0.1.zip

Starting scenario: obstructed-camera-detection..
```

- You should be able to see the events generated on your console window with the detections of obstructed cameras.

### With RTSP Camera - Pipelines

[TODO]

### With Azure Setup

VisionAI app is available at a Azure Market place, one can download and use it by

following steps mentioned [here](#)

## Features

Here are the features of the obstructed camera detection model:

- Real-time monitoring: The model can monitor the camera's field of view in real-time, providing immediate detection of any obstructions.
- Detection of various obstructions: The model can detect various types of obstructions such as fingers, tape, post-it notes, or other physical objects that might block the camera view.
- High accuracy: The model uses deep learning techniques and can achieve high accuracy in detecting obstructions.
- Easy integration: The model can be integrated with different applications such as video conferencing, surveillance systems, or any other system that requires an unobstructed camera view.

## Training with custom data

The scenario is provided as part of our GPL-v3 package for VisionAI. If you wish to train this with custom datasets, please contact us and we can provide you with the training code. You can do custom training with your own datasets for free, as long as it complies with GPLv3 license (you give back the code to the community). If you are interested in a custom license, please [contact us](#).

## Contact Us

- For technical issues, you can open a Github issue [here](#).
- For business inquiries, you can contact us through [our website](#).