IoT Motion Detection Surveillance System - Design Overview

1. Application Architecture

The IoT Motion Detection Surveillance System is a multi-threaded Golang application designed to run on a Raspberry Pi. It uses goroutines and channels to concurrently handle motion detection, video recording, and file transfer.

The application is composed of three main packages, each responsible for a specific part of the system's functionality:

- **`motion`**: Detects movement using a PIR sensor.
- **`camera`**: Records video when motion is detected.
- **`transfer`**: Transfers the recorded video to a remote server.

Communication between these components is managed through channels, which ensures safe and efficient data exchange between the concurrent processes.

2. Packages and APIs

`main` package

- **`main()`**: The entry point of the application. It initializes the channels, starts the goroutines for motion detection and video processing, and manages the application's lifecycle.

`motion` package

- **`DetectMotion(motionDetected chan<- bool, done <-chan struct{})`**: This function continuously monitors for motion. When motion is detected, it sends a boolean value to the `motionDetected` channel. It also listens on the `done` channel to gracefully shut down.

`camera` package

- **`RecordVideo() (string, error)`**: This function is responsible for recording video. It is called when a motion event is received. It saves the video to a local file and returns the file path.

`transfer` package

- **`TransferFile(filePath string) error`**: This function handles the transfer of the recorded video file to a remote server. It takes the file path as input and uses a transfer protocol (e.g., SCP) to send the file.

3. Component Interaction Diagram

The following diagram illustrates the relationship between the major components and the flow of data through the system.

```
graph TD
    subgraph "Main Application"
        A[main.go]
    end

    subgraph "Goroutine 1: Motion Detection"
        B[motion.DetectMotion]
    end

    subgraph "Goroutine 2: Video Processing"
        C[camera.RecordVideo]
        D[transfer.TransferFile]
    end

    A -- starts --> B
    A -- starts --> C
    B -- "motionDetected (channel)" --> C
    C -- "filePath (string)" --> D
```