

Capítulo 3

Diseño e implementación

En este capítulo se detallan los aspectos técnicos del desarrollo del trabajo. Incluye las consideraciones tomadas en cuenta durante el desarrollo, el detalle de las modificaciones realizadas al modelo de partida, así como el detalle del papel que juega el módulo en el sistema. De igual manera, se hace una descripción cronológica del desarrollo del módulo.

3.1. Consideraciones generales

El módulo de inteligencia artificial propuesto es una implementación del modelo YOLOv3, especialmente modificada para cumplir con las necesidades específicas del cliente, así como una serie de requisitos funcionales. Es por esto que fue necesario hacer modificaciones a los archivos originales de la biblioteca.

Como se puede observar en el capítulo 1, no todos los requerimientos funcionales propuestos para el inicio del trabajo pudieron ser cubiertos. Recordando que esta versión del trabajo representa un mínimo producto viable, a continuación se presenta una lista de todos los requerimientos funcionales que fueron cubiertos:

- El módulo se ejecuta en un computador de uso de hogar.
- El módulo es capaz de reconocer **n** clases como intruso.
- Se reporta al usuario de manera visual.
- Se reporta al usuario a través de un archivo de fácil análisis.
- Resulta muy sencillo iniciar la ejecución del módulo a través de su interfaz gráfica.

De igual manera, el módulo está construido de manera que realizar modificaciones, o bien, expandir su funcionalidad en un futuro resulte muy sencillo. Es de vital importancia recalcar que esto se hizo teniendo en cuenta que se trata de un módulo que, a fin de salir de este estatus de mínimo producto viable, deberá sufrir un proceso de optimización muy fuerte. Algunos ejemplos de puntos en los que se pueden aplicar estos trabajos son:

- El módulo está diseñado para filtrar las etiquetas que no hacen parte de las clases dadas en la definición de intruso. Esto implica que, a pesar de que no se reporte en ninguna de las dos metodologías, el módulo aún reconoce otras clases. En otras palabras, si se puede observar un modelo en el *frame*, este será satisfactoriamente detectado y son los métodos encargados del reporte los que no lo entregarán como un resultado positivo.

Capítulo 3

Diseño e implementación

En este capítulo se detallan los aspectos técnicos del desarrollo del trabajo. Incluye las consideraciones tomadas en cuenta durante el desarrollo, el detalle de las modificaciones realizadas al modelo de partida, así como el detalle del papel que juega el módulo en el sistema. De igual manera, se hace una descripción cronológica del desarrollo del módulo.

3.1. Consideraciones generales

El módulo de inteligencia artificial propuesto es una implementación del modelo YOLOv3, especialmente modificada para cumplir con las necesidades específicas del cliente, así como una serie de requisitos funcionales. Es por esto que fue necesario hacer modificaciones a los archivos originales de la biblioteca.

Como se puede observar en el capítulo 1, no todos los requerimientos funcionales propuestos para el inicio del trabajo pudieron ser cubiertos. Recordando que esta versión del trabajo representa un mínimo producto viable, a continuación se presenta una lista de todos los requerimientos funcionales que fueron cubiertos:

- El módulo se ejecuta en un computador de uso de hogar.
- El módulo es capaz de reconocer **14** clases como intruso.
- Se reporta al usuario de manera visual.
- Se reporta al usuario a través de un archivo de fácil análisis.
- Resulta muy sencillo iniciar la ejecución del módulo a través de su interfaz gráfica.

De igual manera, el módulo está construido de manera que realizar modificaciones, o bien, expandir su funcionalidad en un futuro resulte muy sencillo. Es de vital importancia recalcar que esto se hizo teniendo en cuenta que se trata de un módulo que, a fin de salir de este estatus de mínimo producto viable, deberá sufrir un proceso de optimización muy fuerte. Algunos ejemplos de puntos en los que se pueden aplicar estos trabajos son:

- El módulo está diseñado para filtrar las etiquetas que no hacen parte de las clases dadas en la definición de intruso. Esto implica que, a pesar de que no se reporte en ninguna de las dos metodologías, el módulo aún reconoce otras clases. En otras palabras, si se puede observar un modelo en el *frame*, este será satisfactoriamente detectado y son los métodos encargados del reporte los que no lo entregarán como un resultado positivo.