Instituto Tecnológico de Costa Rica
Escuela de Ingeniería en Computación

"Data Analytics, procesamiento de grandes volúmenes de información para generar inteligencia de negocios"

Proyecto de Graduación para optar por el título de
Bachillerato de Ingeniería en Computación

Marco Antonio Sánchez Sánchez

San Carlos Noviembre, 2012

Tabla de Contenido

Resumen

En el siguiente documento se explica primeramente el contexto del proyecto y el organigrama de la empresa y se describe el problema que se intenta dar solución, las metodologias y tecnologias con las que se va a trabajar el proyecto y el personal involucrado.
Se abarca el objetivo general y los específicos, de describen los productos esperados y los requerimientos necesarios para la elaboración del proyecto, los requerimientos no funcionales y un análisis de riesgos que pueden o llegan a afectar el proyecto.

Se describe un modelo de la solución del problema, a modo de diagrama de funcionalidad, con una explicación de cada uno de los pasos que sigue la aplicación para llegar a la solución, una descripción de los Modelos de subsistemas, como lo es el de guardar imágenes directamente en el HDFS(sistema distribuido de Hadoop).
Una explicación de los componentes y sistemas, utilizados en la solución del problema, o que ayudan directa o indirectamente a esta.

Por último una sección con el trabajo a futuro para el proyecto, recomendaciones y sugerencias para la empresa sobre los temas del big data y hadoop. Conclusiones del proyecto y experiencias laborales del desarrollador.

Contexto del proyecto

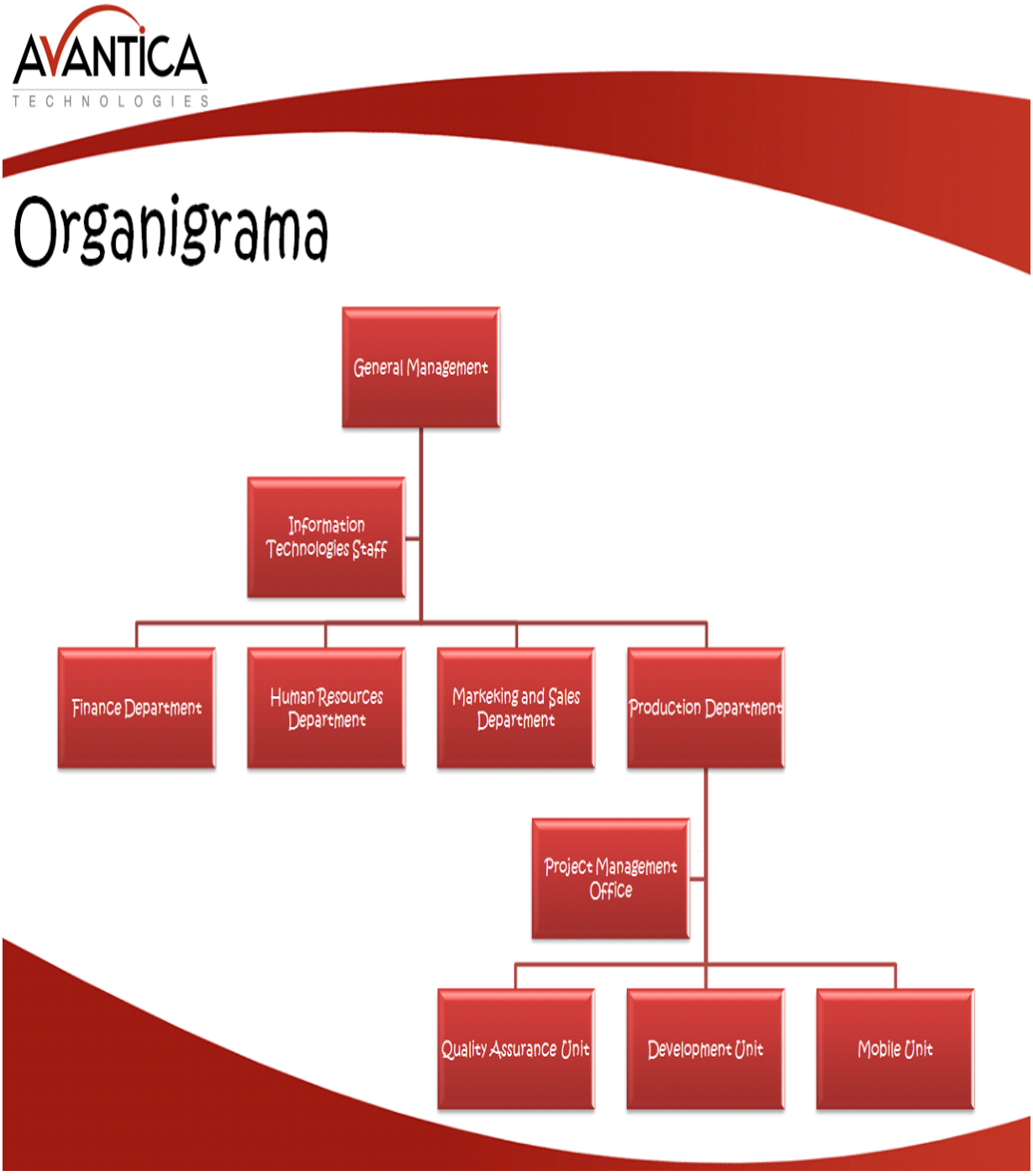
La empresa se especializa en desarrollo de software a la medida. Esta dividida en Departamento de Finanzas, de Recursos Humanos, Ventas y Mercadeo, Producción y Soporte (TI).

El proyecto va a ser realizado dentro del departamento de producción de la empresa Avantica San Carlos y para uso interno de la empresa.

Permitir generar conocimiento para el desarrollo de nuevos proyectos que surjan con Big Data, Hadoop, OpenCV & JavaCV, actualmente son tecnologías nuevas y que tienen un gran auge en el mercado mundial.

Consiste en generar inteligencia de negocio a partir de los reportes generados por el software, y mediante el uso de las cámaras de vigilancia como fuentes de información, para la toma de decisiones.

Organigrama



Descripción del problema.

El proyecto se basa en la idea de generar conocimiento para la toma de decisiones, a través del procesamiento de datos no estructurados para obtener de información relevante para la toma de decisiones y a su vez generar conocimiento acerca las nuevas tendencias tecnológicas de la industria, como Big Data, datos no estructurados, librerías hadoop (mapReduce, HDFS), librería openCV.

Metodologías / tecnologías de trabajo

Scrum: el cual consiste en una metodología ágil de trabajo, con entregas parciales de un producto, priorizadas según los beneficios para el cliente con el fin de mostrar resultados pronto.

Hadoop: Librería que permite junto con un algoritmo MapReduce, mapear y reducir la cantidad de información. Así también HDFS que consiste en parte de la librería que permite el procesamiento solo o multinodos de los datos.
OpenCV: Librería y algoritmo para la detección de rostros.
JavaCV: Librería para realizar las funciones de OpenCV en Java.
Lenguajes de programación: Java, conocimiento de la Librería OpenCV en C y Java.

Personal involucrado

Nombre	Departamento	Labores que realiza	Responsabilidades
Rodrigo Vargas	Gerente de Producción	Coordinar las unidades de desarrollo, QA, y móvil.	Dar seguimiento al proyecto y garantizar los recursos para que llegue a su completitud.
Marco Sanchez	Desarrollador	Investigación de las tecnologías y desarrollo de la aplicación siguiendo estándares de la industria y las mejores prácticas	Investigación y Desarrollo de la aplicación

Necesidades y expectativas

Las expectativas del proyecto son generar inteligencia de negocios a partir del procesamiento de datos no estructurados que generan las cámaras, así también de obtener conocimiento sobre nuevas tecnologías del mercado, crear una aplicación de calidad, asi tambien si el proyecto es exitoso llevarlo a otras áreas de la empresa en donde se pueda generar información, así también de ofrecer dicho conocimiento sobre las tecnologías para proyectos con futuros clientes.

Las necesidades de la empresa para desarrollar este proyecto consiste en querer generar conocimiento para la empresa sobre tecnologías nuevas en el mercado y a su vez obtener un beneficio a la hora de la toma de decisiones, para eso se decidió probar con las cámaras de vigilancia

Requerimientos no funcionales

La aplicación será de procesamiento distribuido utilizando la herramienta hadoop HDFS por lo que los tiempos de respuesta varían.
El idioma de la aplicación será español.

Servidor(es) o equipos para el procesamiento distribuido del gran volumen de información así también se debe contar con una buena resolución de las cámaras de vigilancia, ya que con imágenes borrosas o confusas la detección de los rostros se complica, o puede conllevar a resultados imprecisos.

Análisis de los Riesgos:

Riesgos que ya no representan efectos importantes para el proyecto

- Experiencia de los programadores, ya que se tuvo una buena capacitación y ambientacion con las herramientas.
- Mal diseño debido a que ya se diseñó acorde a lo que se puede cumplir y se tiene bien claro cada uno de los pasos para el desarrollo del proyecto.

Escala de evaluación de riesgos

Valor de Riesgo	Descripción
1	Despreciable
2	Marginal
3	Critico
4	Catastrofico

Descripción del riesgo	Mala administración del tiempo
Categoría del riesgo	Técnicos
Posible causa del riesgo	Cálculo del tiempo optimista
El impacto para el proyecto.	3
La probabilidad de ocurrencia	0.15
La exposición ante el riesgo	0.45
La estrategia de evasión	Asignar las tareas con más tiempo del estimado

La estrategia de mitigación	Realizar un análisis más exhaustivo tomando en cuenta situaciones imprevistas
La estrategia de contingencia	Conversar con el cliente(Empresa) para llegar a un acuerda en tiempo de entrega y así poder entregar un producto completo y probado.

Descripción del riesgo	Complejidad del Proyecto
Categoría del riesgo	Proyecto
Posible causa del riesgo	Tecnologías nuevas en la industria y algoritmos complicados de IA.
El impacto para el proyecto.	3
La probabilidad de ocurrencia	0.10
La exposición ante el riesgo	0.3
La estrategia de evasión	Delimitar el proyecto la fase de detección de caras, no de reconocimiento.
La estrategia de mitigación	Reuniones con el PM, aclarando todas las dudas sobre las tecnologías, y tareas a realizar.
La estrategia de contingencia	Invertir más tiempo de investigación.

Descripción del riesgo	Calidad de los datos, imágenes o video

Categoría del riesgo	Técnicos
Posible causa del riesgo	Imágenes o video de las cámaras muy borrosas.
El impacto para el proyecto.	4
La probabilidad de ocurrencia	0.5
La exposición ante el riesgo	2
La estrategia de evasión	Analizar solo imagenes y video con buena calidad
La estrategia de mitigación	Asegurarse de la calidad de imágenes o video que están almacenando las cámaras.
La estrategia de contingencia	Tomar en cuenta que entre los resultados existe un margen de error y considerarlo a la hora de tomar decisiones.

Objetivo general

Crear una aplicación de software que genere información a partir del análisis de grandes volúmenes de información, mediante el uso de las cámaras de vigilancia y las librerías hadoop, openCV & JavaCV, para el área de producción de la empresa Avantica San Carlos.

Objetivos especificos

1. Investigar sobre las librerías hadoop, openCV & JavaCV y los algoritmos de detección.
2. Desarrollar una aplicación que procese la información en un cluster de al menos dos nodos utilizando el procesamiento distribuido en hadoop.
3. Procesar y generar información a partir de los datos obtenidos por imágenes de las cámaras de vigilancia.

Productos esperados

1. Documento de herramientas y procedimientos de instalación, configuración de hadoop para el procesamiento de grandes volúmenes de información.
2. Una aplicación software que permita procesar los datos de las cámaras de seguridad de manera distribuida entre los equipos y que genere información útil para la toma de decisiones.
3. Documento de herramientas, configuración y utilización de los algoritmos de detección de rostros.

Requerimientos

Configuración de SSH y perfiles de usuario en los equipos que forman parte del procesamiento distribuido.

Consiste en la configuración del SSH de todas las máquinas esto para lograr que la aplicación realice el procesamiento distribuido.

Configuracion de HDFS en los equipos.

HDFS es parte de hadoop y debe ser configurado como master y slave a todas las máquinas para que conozcan que rol van a desempeñar en el procesamiento distribuido.

Implementar y configurar librería OpenCV & JavaCV para la detección de rostros.

Consiste en incorporar código dentro de la aplicación en Java queimplementae la librería de OpenCV(JavaCV) para la detección de los rostros, la cual se va a servir de comunicador entre Java y OpenCV en el reconocimiento de los rostros y va a retornar el resultado del análisis .

Procesamiento de imágenes en el entorno distribuido.

Consiste crear una aplicación en Java que permite utilizar los datos de la cámara y analizarlos mapearlos y reducirlos, mediante el procesamiento distribuido de los equipos.

Crear conector en java (hadoop) para ejecutar las librerías de C(OpenCV).

Consiste en utilizar la librería JavaCV para realizar la detección de los rostros en las imágenes.

Crear archivos con los datos mapeados y reducidos.

Una vez generado el mapeo y reducción de los datos se viene a realizar un archivo con toda la información obtenida de la búsqueda como un estilo log.

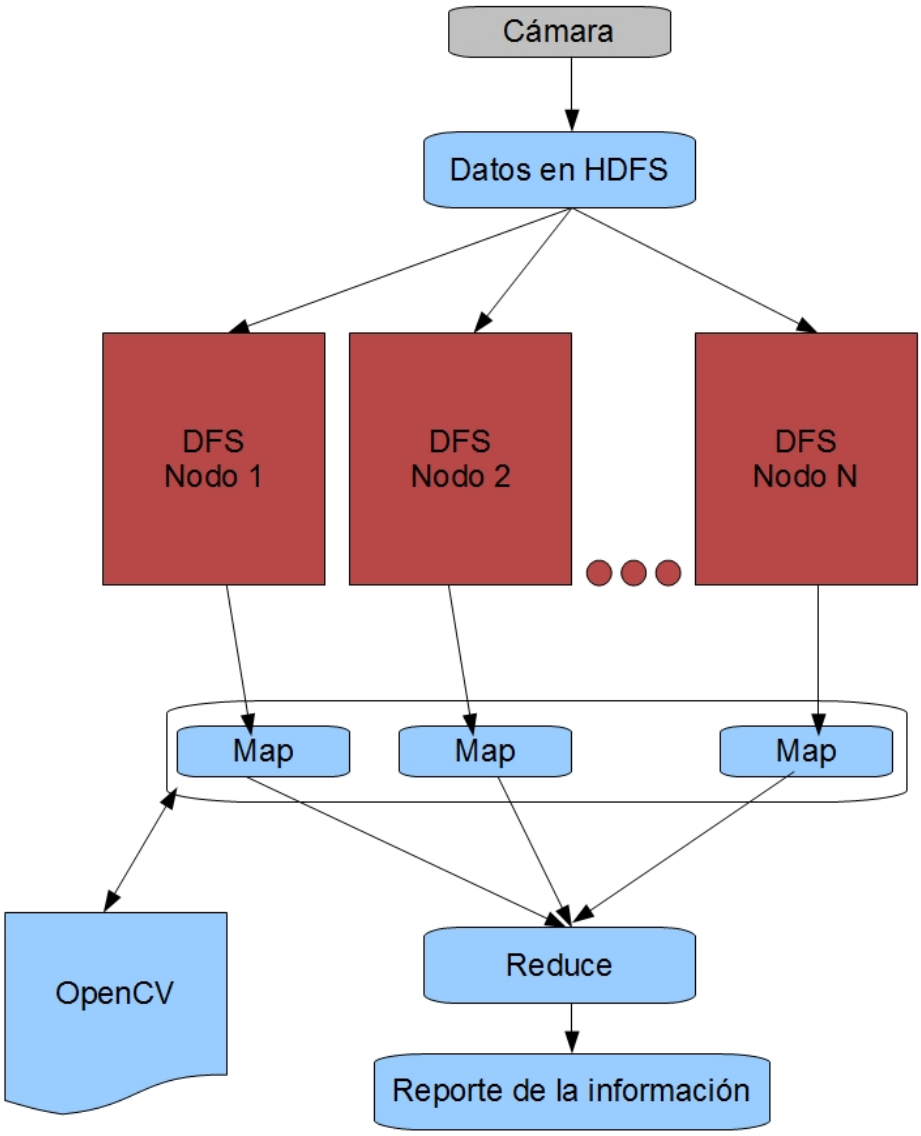
Generar reportes con la información generada.

Número de veces que una persona visita una sala.

Modelo de Diseño

Se describe la arquitectura de la solución propuesta y presente:

Arquitectura conceptual



A continuación se describe el flujo de operación de la aplicación:

1. Insertar datos en el sistema distribuido.
2. Los datos se distribuyen entre los nodos.
3. Se realiza un mapeo de la información con los parámetros de la información que se requiere obtener de las imágenes ya sea rostros sin reconocimiento facial o con reconocimiento facil.
4. Esto genera Maps por cada nodo, para los cuales se hace uso la libreria OpenCV para obtener cierta información de las imágenes procesadas(imagen posee o no una persona, fecha, dia hora de la imagen), los cuales luego son analizados por el Reduce para generar una sola información.
5. Se realiza una reducción de la información con los parámetros de la información que se desea obtener, como por ejemplo cantidad de accesos en un dia a una sala determinada, flujos de tráfico de personas, y sus horas(dependiendo de que se desea obtener).
6. Una vez reducida la información esta se muestra de manera de reportes(Cantidad de personas que entraron a una sala por dia, mes, horas pico de flujo de personas, depende de los parámetros de entrada para realizar el mapeo y la reducción), los reportes son presentados en archivos de texto.

Los modelos de subsistemas

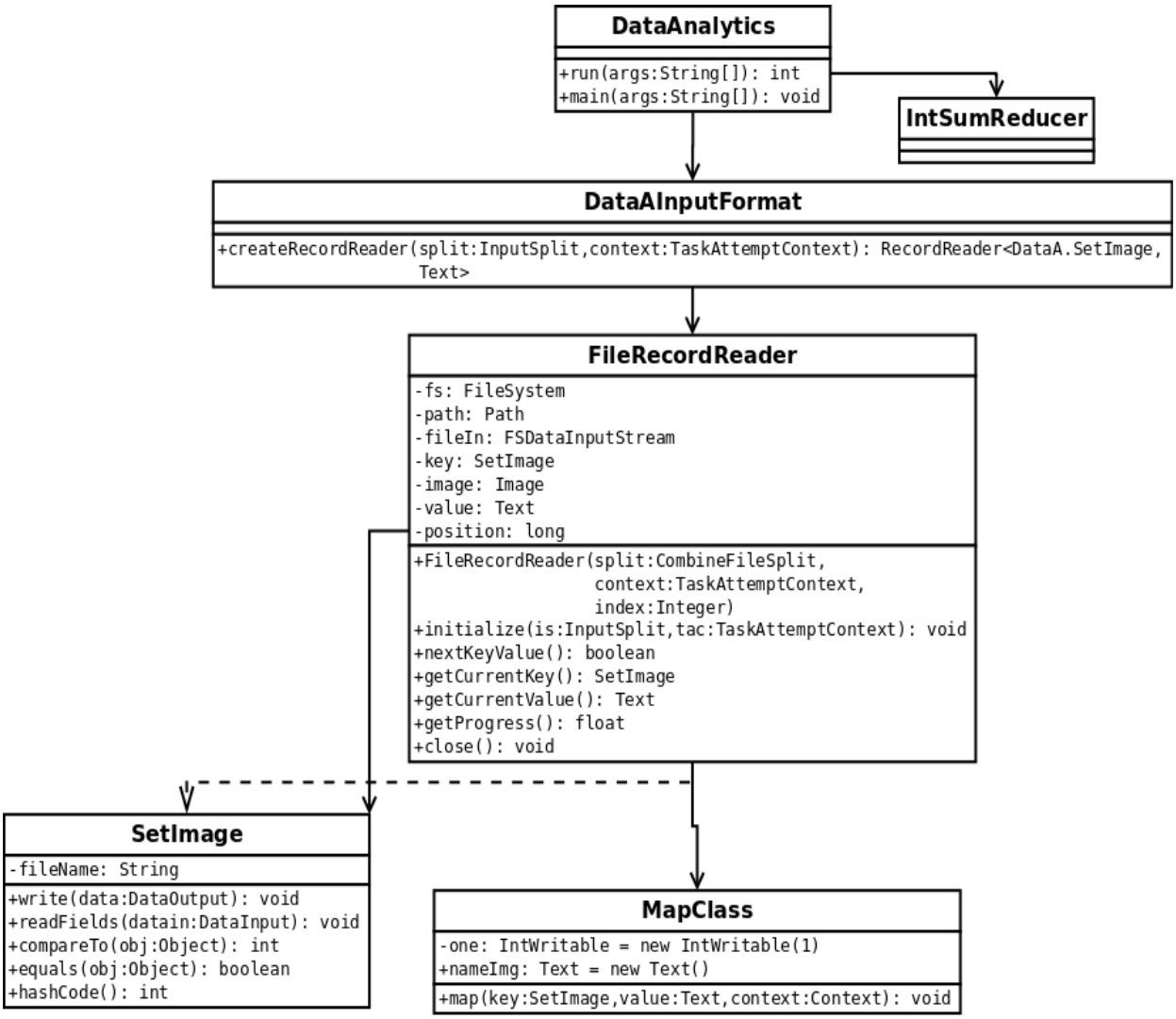
SaveImageHDFS

Se desarrolló una aplicación sencilla en Java que toma una imagen cada 3 segundos por un tiempo definido y las almacena en la carpeta del sistema distribuido de hadoop, esto para poder almacenar grandes volúmenes de información distribuidos entre los nodos del cluster.

Para esto Java genera una imagen temporal que se va a guardar, luego se utiliza la ejecución de comandos de consola de linux para guardar la imagen en el HDFS, y luego eliminarla.

Diagrama de clases

Aplicación en Java - Hadoop



Flujo de datos

Class DataAnalytics

- Esta clase incluye todas las otras clases

- El método Main consiste sólo en inicializar el método run de la aplicación

- En el método run se procede primeramente a inicializar un nuevo job(trabajo), que es el que va a estar encargado de ejecutar todas las clases y métodos de la aplicación, por lo cual se le setean valores como los formatos de entrada de datos, de salida, configuración, las clases de mapeo, combinación y reducción, paths de entrada de datos y salida.
- Empieza la ejecución imagen por imagen que se encuentre en el hdfs.
- Se llama al método **createRecordReader.**

Class DataAInputFormat

- Posee un unico metodo el **createRecordReader** que recibe por parámetros un **split** de entrada con todos los path de las imágenes y en **context** el ID de la tarea asignada y retorna un registro combinado con el split, context y el resultado de la llamada a clase **FileRecReader.**

Class FileRecReader

- Se crean variables privadas para almacenar resultados de los métodos que se van a ejecutar.
- El método FileRecReader establece un nuevo archivo de registro de lectura en el cual se lee y se carga la dirección archivo.
- Llama al método initialize el cual esta vacío, debido a que la clase FileRecReader extiende de una clase predefinida del framework pero para cuestiones de esta aplicación no es requerido.
- Se ejecuta el siguiente método nextKeyValue en el cual se van establecer las direcciones de la librería de openCV, es acá donde se va a realizar la validación de si la imagen registra o no con OpenCV un rostro.
- Para verificar si existe o no un rostro, por medio de ejecución de comandos de consola por medio de java se baja la imagen a un directorio temporal, se ejecuta las líneas de comando para el facedetection , el cual retorna un true o false en consola, el cual se obtiene el resultado y se verifica en java.
- Si el resultado es un true la variable de estado position se pone en 0 y se procede a crear una nueva llave la cual es un objeto de la clase SetImage en el cual seteamos el nombre del archivo que se esta leyendo y se crea un nuevo value también con el nombre de la imagen..se cambia el valor de la variable estado position a 1(esto para confirmar que se leyó ya 1 archivo).

- Verifica si ya se leyó un archivo o no se encontró rostro en la imagen(position = 1), esto para avanzar a leer la siguiente imagen y setear el key y value en null.
- Si este método retorna true a la llamada en DataAInputFormat es por se leyó un rostro entonces llama a la función de map en la clase MapClass, de lo contrario si retorna false ignora y avanza a la siguiente imagen.

class SetImage

Es una clase que crea un objeto imagen

class MapClass

- Posee el método de mapeo, el cual recibe el key, value que se creó en fileRecReader.
- inicializa una variable IntWritable en 1, la cual va a ser el número que va a sumar cuando se esta procesando el proceso de combinación y reducción, crea una variable tipo Text que va a contener el nombre que queremos que muestre los resultados, para esto en el método map se realizan varios splits para separar el nombre de la imagen “diaSemana-mes-dia-hora#minutos#segundos-año”, al seleccionar solo mes día y año del nombre se obtiene una segmentación diferente, para lo cual se va a estar contando todas las imágenes que registren un rostro el mes-día-año indicado,(obteniendo resultados de accesos por dias entre las imágenes de entrada).
- Escribe en el context o el ID de la tarea el par con el nombre(mes-día-año) y el valor 1 del intWritable.

Vuelve a llamar a método nextKeyValue de la clase FileRecReader en donde las variables privadas key y value no son nulas porque ya se leyó un archivo, este método las setea nuevamente en null y retorna false para avanzar a la siguiente imagen.

Una vez completado todo el proceso de mapeo, se procede con el proceso de combinación y reducción para lo cual se utiliza(por ser contar registros leídos) una clase predeterminada de este framework IntSumReducer, el combiner

agarrá todos los pares <key,value>.y los une todos por key, luego el reducer tiene todos esos y los reduce sumando los values de cada par.
Ejemplo:

Resultado de las imágenes mapeadas
key: Wed-22-2012, value: 1
key: Wed-22-2012, value: 1
key: Wed-22-2012, value: 1

Resultado del combiner y reducer
key: Wed-22-2012, value: (1, 1, 1).

Resultado final
Wed-22-2012 3

Face:

Este método se encarga de recibir la imagen obtenida por del HDFS, crear una imagen en formato iplimage, la cual luego se crea otra en escala de grises, esto para hacer mas rapido el face detection, se guarda en memoria y se llama a una función de la librería que recibe por parámetros la imagen, el cascade(xml con parámetros para el reconocimiento de rostros en las imágenes), la imagen de memoria para encontrar las caras que detecte en la imagen, y retorna un valor boolean de true si encontro minimo un rostro o false si no detecto ninguno.

CASCADE_FILE es un archivo XML que se encuentra en la librería OpenCV, consiste en patrones de búsqueda para un rostro humano.

Trabajo a Futuro

El tema de Big Data y Hadoop existen una gran variedad de áreas en las cuales se puede invertir e investigar. Hadoop, su procesamiento en paralelo y distribuido de la información, brinda tiempos de respuesta rápida a procesamientos que en un solo servidor se toman más tiempo realizar, el HDFS brinda almacenamiento distribuido y replicado de la información en los diferentes nodos conectados al cluster.

En la empresa existen muchos programas que generan mucha información que puede ser útil para generar inteligencia de negocios, pero para realizar este análisis se deben de analizar cuales son los datos que realmente importan y que van a generar conocimiento para la empresa, como por ejemplo la idea inicial del proyecto la cual consistía en que las fuentes de datos de las cuales se iba a alimentar la aplicación de hadoop eran la gama de aplicaciones que utiliza la empresa día a día, pero para esto se requiere un buen análisis y conocimiento a fondo de las herramientas para poder tener información útil.

Este proyecto queda como base para generar nuevos utilidades como lo puede ser no sólo la detección de rostros sino el reconocimiento de esos rostros, para lo que se necesitaría una base de datos de patrones con imágenes de las personas de la empresa, de este modo generar conocimiento más seccionado de la información.

Con el uso de la librería OpenCV quedan varias aplicaciones en las que se puede desarrollar, ejemplo el análisis de video de las cámaras, se podría generar una aplicación a modo de práctica, porque depende de su implementación y grado de error, el acceso a la empresa por medio de reconocimiento facial de la persona cuando se acerca a la cámara de entrada.

Conclusiones

Producto	Estado	Observaciones
Documento de herramientas y procedimientos de instalación, configuración de hadoop	Completo	
Una aplicación software que permita procesar los datos de las cámaras de manera distribuida	Completo	
Documento de herramientas, configuración y utilización de los algoritmos de detección de rostros	Completo	

Pese a los cambios realizados a lo largo del desarrollo del proyecto, debido a que no se tenía conocimiento sobre estas tendencias tecnológicas se estimó lo que al final quedó para trabajo futuro por cuestiones de tiempo y cantidad de trabajo que se podía realizar en ese periodo de tiempo. El proyecto se concluyó completo se generó para la empresa los documento necesarios para aportar conocimiento sobre estas nuevas tecnologías, documentos con los procesos de instalación y configuración de las herramientas utilizadas(Hadoop y OpenCV con ayuda de JavaCV), se generó una aplicación en Java para capturar imágenes de la cámara de vigilancia y almacenarlas en el HDFS(sistema distribuido), y otra aplicación utilizando el framework the Hadoop y JavaCV, el cual permite procesar de manera paralela, las imágenes almacenadas en el sistema distribuido, contando y seccionando las imágenes en donde reconozca un rostro humano, generando resultados en archivos de texto con la cantidad de accesos por día.

Así como un documento de herramientas, configuración y utilización de los algoritmos de detección de rostros.

Experiencias adquiridas

Primero quiero mencionar un factor muy importante para mi el ambiente laboral en la empresa es muy agradable, la mayoría de las personas que trabajan aquí son egresadas del TEC, compañeros del TEC, o personas que conocía de la ciudad, o del colegio, por lo que es más fácil adaptarse.

El ajustarse a un horario es un aspecto importante ya que las tareas asignadas tienen que ser realizadas dentro del tiempo establecido y no se pueden dejar para después, como podía hacerse mientras se estudia.

Las bases importantes que le brinda el tec, el aspecto principal es la investigación la cual es muy necesaria y requerida en muchos de los trabajos en el TEC, esto ayuda a fortalecer este aspecto, el proyecto emplea tecnologías nuevas, tuve que realizar mucha investigación sobre las tecnologías y su implementación y funcionamiento. También el saber programar en lenguajes como Java y C me fue bastante útil.