

Universidad de las Ciencias Informáticas

Facultad 6



**Trabajo de Diploma para optar por el Título de
Ingeniero en Ciencias Informáticas**

**Componente para evaluar la calidad de videos digitales en e
Sistema de Transmisión de Canales Virtuales utilizando la
técnica Referencia Completa**

Autores:

Diana Rosa Varona Ocaña
Rolando Alberto Escobar

Tutores:

Msc. Rafael Cardero Álvarez
Ing. Eduardo Cepero Utra

**Junio, 2015
Año 57 de la Revolución**

Declaración de autoría

Nosotros, Diana Rosa Varona Ocaña y Rolando Alberto Escobar, con carné de identidad 92032547410 y 91042938761 respectivamente, declaramos ser autores de la presente tesis que tiene por título “*Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa*” y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmamos la presente a los ____ días del mes de _____ del año 2015.

Autor: Diana Rosa Varona Ocaña

Autor: Rolando Alberto Escobar

Tutor: Ing. Eduardo Cepero Utra

Co-Tutor: Msc. Rafael Cardero Álvarez

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Datos de contacto

Tutor: Msc. Rafael Cardero Álvarez.

Ingeniero en Ciencias Informáticas (2008). Máster en Informática Aplicada (2011). Trabaja en lo referido al procesamiento de imágenes digitales y el reconocimiento de patrones en general. Ha participado en varios eventos afines a su campo de interés.

Tutor: Ing. Eduardo Cepero Utra.

Ingeniero en Ciencias Informáticas (2011). Trabaja en lo referido al procesamiento, transmisión y monitorización de señales digitales. Ha participado en varios eventos afines al área en que se desarrolla.

Agradecimientos

A mi mamá, por ser madre, padre, amiga y consejera. Por haberme enseñado a valorar cada regalo que nos da la vida. Porque a pesar de todos los obstáculos que se le han presentado, siempre has estado a mi lado con una sonrisa y con su amor incondicional. Quiero agradecerle por todo lo que es y por todos los valores que me ha enseñado. Te Amo Madre.

A mi abuelita Celia, “la reina de mi corazón”, por ser mi principal motivación en la vida y ser la persona más maravillosa y especial que he conocido. Por ser la fuerza que me impulsa a levantarme en los momentos más difíciles.

A mi papá por haberme regalado momentos tan maravillosos en mi niñez.

A mi abuelito por su infinito amor y por todo su apoyo. Por ser el mejor abuelo del mundo.

A mi hermanita por todo su apoyo y cariño. Por siempre estar disponible para mí. Gracias Tita.

A mis tías y primos por su cariño y amor, por su apoyo durante toda la vida.

A mi tío “Uro” por su cariño y por ser para mi mamá el hermano que nunca tuvo.

A mi esposo Ale, por regalarme 5 años llenos de amor incondicional. Por apoyarme en todos los momentos tristes y poner el pecho ante las balas dirigidas a mí.

A quienes han sido como unos padres durante estos 5 años. Amneris y Remigio, gracias por su apoyo, por sus consejos y por acogerme en su familia con los brazos abiertos.

A mis amigos Dayana y Chino, por todos los momentos que hemos compartido y por ser como unos hermanos para mí.

A mis tutores, Especialmente a Rafael por las largas horas de revisión del documento, su preocupación y responsabilidad. Por ser el responsable de que hoy nos podamos graduar.

A los profesores Vilmavis y Reynier Pupo por ayudarme siempre que los necesite.

Gracias.

Diana

Agradecimientos

A mi mamá, por darme todo el amor del mundo, por estar presente en todos los momentos de mi vida, por ser mi hermana, mi amiga y por tratar de darme siempre lo mejor.

A mi papá, por ser mi mejor amigo, por darme los mejores consejos, por no perder ni un segundo la confianza en mí, por ser mi ejemplo a seguir en la vida, por el esfuerzo que hace para que pueda seguir adelante y cumplir todas mis metas.

A mis abuelos, por aconsejarme y por enseñarme a ser mejor cada día.

A mi abuelita mima, por ser la persona más maravillosa que he conocido y brindarme su amor de madre incondicional.

A mis tíos, Danilo, Danio, Roly, Albarito, mi tía Aya, mi primo José, Arsenio y Alie por apoyarme durante estos cinco años, por brindarme su cariño incondicional y ser sobre todo mis amigos en todo momento.

A mis compañeros que a lo largo de la carrera he llegado a considerar hermanos, Daldis, Eduardo, Reidel, Adrián, Serguey, Ronald, Yusniel y Jorge, todos siempre presentes en las mejores y peores situaciones.

A nuestro co-tutor Rafael, que con persistencia y rigor hizo posible que este trabajo saliera adelante.

A todas las amistades que fui encontrando a lo largo de la carrera que aunque no se mencionen siempre están presente.

A las personas del proyecto en especial la profe Mónica.

Rolando

Dedicatoria

A mi mamá por su apoyo.

A mi abuela por ser motivo de inspiración.

A mi hermana por su amor.

A papá por su exigencia.

A mi abuelito por estar siempre conmigo.

A mi esposo por su amor incondicional.

A todas esas personas que ocupan un lugar en mi corazón

Diana

Dedicatoria

A mis padres, por su confianza, por sus consejos, por ayudarme a ser cada día mejor y por estar junto a mí en todo momento...

A mi familia, por contar siempre con su apoyo...

Y muy especial a mi abuela Vivo, que tanto deseó este momento.

Rolando

Resumen

La transmisión de canales virtuales es utilizada en los medios digitales para difundir información. Su relación con la tecnología y el surgimiento de nuevas formas para obtener videos digitales provoca que la calidad en ellos varíe. En el Sistema de Transmisión de Canales Virtuales la medición de la calidad en la transmisión de videos se realizaba mediante la comparación entre la cantidad de paquetes del video original y el video transmitido. Este método obviaba los errores comunes que percibe la visión humana. En busca de darle solución a la problemática planteada se decidió crear un componente que evaluara la calidad utilizando la técnica de tipo “Referencia Completa”. Para ello se basa en la detección de las distorsiones “Efecto de bloque” y “Borrosidad”. El desarrollo del componente se realizó utilizando la metodología RUP. Las herramientas utilizadas fueron Visual Paradigm para el modelado de los diagramas propuestos por la metodología y QtCreator para el desarrollo del software. Entre las tecnologías se encuentran como lenguaje de programación C++, el marco de trabajo QT, la biblioteca OpenCV, y la librería XMLRPC. Además se utilizó el lenguaje de modelado UML para apoyar la etapa de diseño de la solución. Se realizaron pruebas de integración, eficiencia y pruebas unitarias al producto final para verificar los resultados obtenidos.

Palabras claves: calidad, componente, transmisión de canales virtuales, video digital.

Abstract

The broadcast of virtual channels is a common way, used in digital media to spread information. Its relationship with technology and the emergence apparition of new ways for getting digital videos makes that the quality of them vary. At the University of Informatics Sciences measuring the quality of streaming video was done by comparing the number of packages of video and the number of packages transmitted. This method avoided the common mistakes that human vision perceives. In the searching of a solution to the existing problematic situation, it was decided to create a component specifically using the complete reference technique that is within the objective methods. The technique is supported by evaluating the Effect of Block and Fuzziness distortion. The component development was carried out using the RUP methodology. The tools used were Visual Paradigm for modelling the diagrams proposed by the methodology and QtCreator for the software development. Among the technologies there are C++ as programming language, the Qt framework and the OpenCV, XML-RPC and VLC library. Besides, the UML modelling language was used for supporting the period of solution design. Functional tests, tests of loading and stress as well as unitary tests were also carried out for checking the results obtained.

Keywords: component, digital video, quality, virtual transmission channels.

Índice

INTRODUCCIÓN	1
1.1 CONCEPTOS ASOCIADOS	5
1.2 SISTEMAS HOMÓLOGOS	6
1.3 CLASIFICACIÓN DE LOS MÉTODOS PARA EVALUAR LA CALIDAD EN VIDEOS DIGITALES	8
1.3.2 <i>Posibles distorsiones en videos digitales</i>	9
1.4 ALGORITMOS PARA LA DETECCIÓN DE LA DISTORSIÓN “EFECTO DE BLOQUE”	10
1.5 ALGORITMOS PARA LA DETECCIÓN DE LA DISTORSIÓN “BORROSIDAD”	12
1.6 METODOLOGÍA DE DESARROLLO	13
1.7 HERRAMIENTAS Y TECNOLOGÍAS	14
1.7.1 <i>Lenguaje Unificado de Modelado</i>	14
1.7.2 <i>Visual Paradigm</i>	14
1.7.3 <i>Lenguaje de Programación C++</i>	15
1.7.4 <i>Marco de trabajo Qt</i>	15
1.7.5 <i>Entorno de Desarrollo Integrado (IDE) QtCreator</i>	16
1.7.6 <i>Biblioteca OpenCV</i>	16
1.7.8 <i>Biblioteca XML-RPC</i>	17
1.8 CONCLUSIONES PARCIALES	17
CAPÍTULO 2: DISEÑO DE LA SOLUCIÓN PROPUESTA	18
2.1 ALGORITMOS PROPUESTOS	18
2.1.1 <i>Algoritmo propuesto para detectar “Efecto de bloque”</i>	19
2.1.2 <i>Algoritmo propuesto para detectar “Borrosidad”</i>	20
2.3 SOLUCIÓN PROPUESTA	21
2.4 ESPECIFICACIÓN DE REQUISITOS	21
2.4.1 <i>Modelo de dominio</i>	22
2.4.2 <i>Diagrama de clases del Modelo de Dominio</i>	22
2.4.3 <i>Requisitos del componente</i>	23
2.5 ANÁLISIS DE LA SOLUCIÓN	24
2.6 DISEÑO DE LA SOLUCIÓN	24
2.6.1 <i>Descripción de los actores</i>	25

2.6.2	<i>Modelo de casos de uso del sistema</i>	25
2.6.3	<i>Arquitectura del Sistema</i>	28
2.6.4	<i>Patrones de Diseño</i>	28
2.6.5	<i>Estándar de Codificación</i>	29
2.6.6	<i>Diagrama de clases de diseño</i>	30
2.6.7	<i>Diagrama de secuencia</i>	30
2.7	FUNCIONES DE OPENCV UTILIZADAS	31
2.8	CONCLUSIONES PARCIALES	31
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS		32
3.1	DIAGRAMA DE COMPONENTES	32
3.2	DIAGRAMA DE DESPLIEGUE	32
3.3	PRUEBAS	33
3.3.1	<i>Pruebas unitarias</i>	33
3.3.2	<i>Pruebas de integración</i>	36
3.3.3	<i>Pruebas Precision y Recall</i>	36
3.4	CONCLUSIONES PARCIALES	40
CONCLUSIONES GENERALES		41
RECOMENDACIONES		42
REFERENCIAS BIBLIOGRÁFICAS		43
BIBLIOGRAFÍA CONSULTADA		46
ANEXOS		49

Índice de Figuras

<i>Figura 1. Demostración del “Efecto de bloque”</i>	10
<i>Figura 2. Demostración de “Borrosidad”</i>	10
<i>Figura 3. Sincronización entre el video original y el video recibido</i>	18
<i>Figura 4. Vecindad del fotograma clave en el video original</i>	19
<i>Figura 5. Evolución del algoritmo propuesto para detectar “Borrosidad”</i>	21
<i>Figura 6. Modelo de dominio</i>	22
<i>Figura 7. Diagrama de caso de uso de la solución propuesta</i>	25
<i>Figura 8. Ejemplo del estilo de código utilizado en el desarrollo del componente propuesto</i>	30
<i>Figura 9. Diagrama de clases de diseño del componente</i>	30
<i>Figura 10. Diagrama de componentes de la solución</i>	32
<i>Figura 11. Diagrama de despliegue de la solución</i>	33
<i>Figura 12. Grafo de flujo del método “blockMetric”</i>	35

Índice de Tablas

<i>Tabla 1. Caso de uso “Evaluar calidad de video digital”</i>	25
<i>Tabla 2. Valores para analizar la Complejidad ciclomática (Martínez Salazar, 2012)</i>	34
<i>Tabla 3. Total de imágenes clasificadas por tipo de distorsión</i>	37
<i>Tabla 4. Total de imágenes de acuerdo a las características del ambiente</i>	37
<i>Tabla 5. Resultados obtenidos al evaluar la “Borrosidad” utilizando distintos umbrales</i>	38
<i>Tabla 6. Resultados obtenidos al evaluar la “Efecto de bloque” utilizando distintos umbrales</i>	39
<i>Tabla 7. Resultados globales del componente propuesto en la investigación</i>	39

Introducción

La calidad es un indicador a valorar en distintas áreas de la sociedad, ya sean empresas, organizaciones u otras entidades. El término es variable pues está sujeto a criterios personales, por lo que definirlo ha sido tarea constante de varios autores entre los que se destacan Joseph M. Juran, William E. Deming, Philip B. Crosby y Kaoru Ishikawa. Según la Organización Internacional de Estandarización (ISO), el término “calidad” se refiere al grado en el que un conjunto de características inherentes cumple con los requisitos, entendiéndose por “requisito” necesidad o expectativa establecida, generalmente implícita u obligatoria (ISO, 2015).

En todas las épocas este indicador ha estado presente ayudando a elegir las opciones más convenientes. El avance progresivo y apresurado que ha tenido la tecnología en las últimas décadas, ha incluido de manera ineludible a la calidad como métrica. Su utilización va desde la valoración técnica de un dispositivo móvil, de un elemento de *hardware* o de un producto informático, hasta análisis de imágenes y videos digitales.

A grandes rasgos un video consiste en “la captura de una serie de fotografías (en este contexto llamadas fotogramas) que luego se muestran en secuencia y a gran velocidad para reconstruir la escena original” (Puerto Acevedo, 2014). El concepto especifica características particulares que dependen de la tecnología con que se realice o el medio donde se utiliza.

Un espacio donde se encuentra un cúmulo sorprendente de videos digitales es en la red de redes, conocida como Internet. Estos son tomados desde dispositivos diferentes, lo que implica que se generen variedades de formatos. La difusión de videos por esta vía ha provocado que la obtención de ellos sea necesariamente realizando descargas desde páginas web. Según el Dr José Joskowicz, destacado investigador latinoamericano, el video digital distribuido a través de redes de comunicaciones, sufre varios tipos de distorsiones durante el proceso de adquisición, compresión, procesamiento, transmisión y reproducción. Por ejemplo, las técnicas utilizadas habitualmente en la codificación digital del video, introducen pérdidas de información, lo que genera distorsiones adicionales (Joskowicz, 2008).

En Cuba la utilización de videos digitales en la televisión, espectáculos en vivo y otras vías de difusión masiva para la población, también ha aumentado. En instituciones donde la tecnología está presente en el desarrollo de la mayoría de las actividades, es imprescindible poseer un mecanismo para monitorizar y controlar la calidad de los videos digitales que se utilizan. Entre estas entidades se encuentra la Universidad de las Ciencias Informáticas (UCI).

La UCI posee diferentes centros de desarrollo especializados en distintas temáticas de interés para la universidad. En el caso de la Facultad 6 se puede mencionar el centro Geoinformática y Señales Digitales (GEYSED). Dicho

centro se encuentra estructurado por dos departamentos: Desarrollo de Componentes y Departamento de Integración de Soluciones. Específicamente en el departamento de Soluciones se desarrolló el Subsistema de Monitorización de Canales Virtuales (SMCV). Este subsistema tiene como objetivo detectar los problemas en las transmisiones planificadas para el día, según transcurra la cartelera con las medias planificadas.

El análisis de la calidad en el SMCV, de los videos que transmite el Sistema de Transmisión de Canales Virtuales (STCV) se realiza a través del conteo de la cantidad de paquetes recibidos y perdidos; sin embargo no se detectan los errores específicos que percibe la visión humana. Este carece de elementos que respalden el resultado y dificulta la toma de acciones correctivas, ya que no es capaz de detectar ningún tipo de distorsión a las que se expone el flujo de transmisión en un entorno, ni de informarlo. El proceso de monitorización actual precisa de un trabajador que se dedique a observar constantemente la transmisión, tornándose engorroso y poco fiable, dependiendo del grado de compromiso del supervisor. Teniendo en cuenta todo lo anteriormente expuesto se plantea como **problema a resolver**: ¿Cómo evaluar la calidad de video en las transmisiones del Sistema de Transmisión de Canales Virtuales (STCV)?

El **Objeto de Estudio** de la investigación es “la evaluación de la calidad de videos digitales” y el **Campo de Acción** es “la evaluación de la calidad de videos digitales utilizando la técnica Referencia Completa”. Se plantea como **Objetivo General** “Desarrollar un componente de software que permita evaluar la calidad de video en las transmisiones del Sistema de Transmisión de Canales Virtuales utilizando la técnica de Referencia Completa”.

Para resolver la problemática planteada se definen como **Preguntas Científicas**:

1. ¿Cuáles son los fundamentos teórico-metodológicos asociados a los métodos para evaluar la calidad de videos digitales y en específico a los basados en la técnica de Referencia Completa?
2. ¿Cuáles son las herramientas, tecnologías y la metodología de desarrollo de software necesaria para desarrollar un componente de software que permita evaluar la calidad de video en las transmisiones del STCV utilizando la técnica Referencia Completa?
3. ¿Cómo desarrollar un componente que permita evaluar la calidad de video en las transmisiones del STCV utilizando la técnica Referencia Completa?
4. ¿Cómo validar el componente para evaluar la calidad de video en las transmisiones del STCV utilizando la técnica Referencia Completa?

Se definen, como guía, las siguientes **Tareas de la Investigación**:

1. Elaboración del marco teórico-metodológico asociado a los métodos para evaluar la calidad de videos digitales y en específico a los basados en la técnica Referencia Completa.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

2. Determinación de las herramientas, tecnologías y la metodología de desarrollo necesarias para desarrollar un componente de software que permita evaluar la calidad de video en las transmisiones del STCV utilizando la técnica Referencia Completa.
3. Elaboración de los artefactos indicados por la metodología de desarrollo de software seleccionada.
4. Desarrollo de un componente de software que permita evaluar la calidad de video en las transmisiones del STCV utilizando la técnica Referencia Completa.
5. Validación del componente para evaluar la calidad de video en las transmisiones del STCV utilizando la técnica Referencia Completa.

Para la investigación se utilizaron los siguientes **Métodos Científicos**:

Métodos Teóricos

- ❖ **Histórico - lógico:** se utilizó durante el estudio de la evolución de los métodos y aplicaciones que permiten evaluar la calidad de videos digitales.
- ❖ **Analítico - sintético:** se utilizó al realizar el estudio de las tecnologías y herramientas a emplear en el desarrollo del componente propuesto, así como la metodología de desarrollo a utilizar. Además permitió sintetizar sus características y analizar la viabilidad de cada una.
- ❖ **Modelación:** se empleó fundamentalmente para modelar los artefactos que se generan a lo largo del proceso de desarrollo del componente para evaluar la calidad de video.

Técnica de recopilación de información

- ❖ **Entrevista:** se utilizó para conocer a través de reuniones con miembros del proyecto STCV, la manera en que se analiza la calidad de los videos digitales en la actualidad y las deficiencias que posee la estrategia existente. Además se empleó para conocer los requerimientos necesarios del componente a desarrollar (Ver Anexo 1).

El documento que sustenta la solución propuesta está dividido en tres capítulos de la siguiente manera:

Capítulo 1: Se describe el marco teórico de la investigación a través de los principales conceptos asociados a los sistemas de transmisión y la calidad de videos digitales. Además muestra características de los sistemas que realizan la evaluación de la calidad en videos digitales, enmarcado a los sistemas de transmisión de canales virtuales. Se analizan algoritmos para detectar las distorsiones “Borrosidad” y “Efecto de Bloque” y se describen las herramientas y tecnologías utilizadas, así como la metodología de desarrollo seleccionada.

Capítulo 2: Se describen los algoritmos propuestos, la solución propuesta y se especifican los requisitos funcionales. Se construyen el modelo de dominio, diagramas de caso de uso y de clases del diseño, elaborados

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

para sustentar la implementación del componente. Además se plantea la información referente a los actores del sistema, la arquitectura y los patrones del diseño presentes en el componente.

Capítulo 3: Se muestran los resultados generados por los algoritmos propuestos. Se describe la fase de implementación a través de los diagramas correspondientes así como el proceso de prueba realizado para validar la solución desarrollada.

Capítulo 1: Fundamentación Teórica

En el presente capítulo se describen los principales conceptos asociados a la investigación, con el fin de brindar un entendimiento teórico de la solución que se propone. Se realiza el análisis de sistemas homólogos. También se estudian algoritmos para la detección de las distorsiones “Borrosidad” y “Efecto de Bloque”. Además se plantean las características asociadas a las herramientas y tecnologías que intervienen en la elaboración del componente, así como de la metodología de desarrollo seleccionada.

1.1 Conceptos asociados

Video Digital

La palabra “video” etimológicamente hablando significa “yo veo”. Se conoce como video a la tecnología que permite recoger o registrar fragmentos de la realidad, para posteriormente reproducirlos mediante aparatos creados para tal efecto (Casado Alvarado, 2000).

De manera general, Manuel Agustí y sus colaboradores, consideran que un video digital “representa una secuencia de imágenes (fotogramas), donde además de los parámetros de resolución espacial (número de píxeles) y profundidad (número de bits), hay que considerar un número adecuado de imágenes por segundo que permitan crear la ilusión de movimiento” (Agustí, et al., 2010). Los autores del presente trabajo coinciden con este concepto por su relación con la investigación planteada.

Calidad de video

En esta investigación ya fue expuesto el significado asociado al término “calidad”, según la ISO. Sin embargo los autores de la misma consideran el siguiente concepto como uno más ajustado al dominio de la investigación:

“El término calidad hace alusión a una propiedad o conjunto de propiedades inherentes a un objeto. Extrapolando este concepto a un video, se entendería como la capacidad que tiene una secuencia de video de representar el objeto original, es decir, la exactitud o parecido entre ambos” (Hormigos Baz, 2009).

Trasmisión de canales virtuales

Los canales virtuales son el resultado de la integración de la Informática y las Telecomunicaciones. Estos permiten almacenar información y emitirla usando servidores de video, bases de datos, sistemas de almacenamiento en cinta y almacenamiento masivo de archivos.

Los archivos de video emitidos en los canales virtuales “incluyen información referente a su contenido, lo que se conoce como metadato. El acceso a la información de los archivos se realiza desde equipos de cómputo normales,

vía Internet o Intranet, que ejecutan a su vez un software de edición o reproducción de video, lo que asegura que la información de los productos televisivos esté disponible en línea para el usuario” (Carvajal Álvarez , 2009).

La trasmisión de canales virtuales ha tomado gran auge e importancia por las ventajas de tiempo, espacio, flexibilidad, independencia y disponibilidad que proporciona tanto a quien emite como al usuario final.

1.2 Sistemas homólogos

Universidad Libre Canal Virtual o ULCV

Es un producto para implementar y poner en operación un Canal Virtual. Su objetivo es brindarle a la comunidad de la Universidad Libre Seccional de Cali, en Colombia, un medio de comunicación útil para la divulgación y la gestión del conocimiento.

EL producto está dividido en 4 módulos específicos a través de los cuales se maneja la transmisión virtual (Carvajal Álvarez , 2009):

1. **Módulo de Registro (MR):** a través de él se obtienen las imágenes y el sonido, necesarios para la realización de un programa.
2. **Módulo de Edición y Copiado (MEC):** permite editar los registros almacenados ya sea con Edición Lineal (EL) o Edición No Lineal (ENL).
3. **Módulo de Procesamiento (MP):** realiza las operaciones de sincronización, amplificación, limitación, filtrado, ajuste y corrección a las que se someten las señales de banda base.
4. **Módulo de Emisión (ME):** es el conjunto de equipos y subsistemas que permiten difundir el programa producido por medio de una transmisión para que sea recibido por el público (televidente) que es el usuario final.

El producto descrito no gestiona la calidad del video a gran escala, solamente hace alusión a ella en el Módulo de Procesamiento. Este módulo evalúa la calidad utilizando técnicas de análisis visual y no utiliza métricas de evaluación, funcionando similar a como se analiza la calidad actualmente en STCV. La forma de implementación del sistema no añade funcionalidades concretas sino que reutiliza elementos de investigaciones relacionadas con el tema. Todo esto provoca que se desestime la utilización ULCV como solución a la problemática identificada.

Virtual Power Video Server

Es un software de televisión que funciona como un sistema de administración de materiales audiovisuales. Permite la emisión de contenidos gráficos, de video, animaciones, texto y con interactividad. Utiliza un planificador de emisiones para programar días y horas exactas para la reproducción del contenido audiovisual. Entre sus principales características se encuentran (Virtual, 2010):

- ❖ Tres modos de operación (dos automáticos y uno manual).
- ❖ Generador de caracteres y gestor de gráfico incorporado.
- ❖ Asistente de programación.
- ❖ Bajos costos de instalación y de operación.
- ❖ Registro detallado de los videos emitidos (autor, descripción del video, duración, fecha y hora de emisión).

El sistema en general posee ventajas tentadoras para su selección sin embargo no es un sistema modular. Esto trae como inconveniente que es preciso obtener el sistema completo lo que difiere del objetivo propuesto en la investigación. Además, aunque su precio es bajo, no es un sistema libre de pago y el código no es público, lo cual provoca que no pueda ajustarse al subsistema de trasmisión existente.

Además, un dato importante a tener en cuenta es que el país se encuentra inmerso en un proceso de independencia tecnológica que incluye la producción de productos informáticos propios, por lo que no cumple objetivo realizar una inversión para adquirir un sistema extranjero.

MSU (Video Quality Measurement Tool (VQMT))

Es un programa que realiza la medición de la calidad de video de forma objetiva. Esta aplicación permite crear una comparación objetiva de los *codecs*¹ de video y realizar el proceso de análisis de videos-filtros. El software evalúa la calidad de los videos aplicando una serie de métricas, representadas en forma de *plugins*², para cuantificar el índice de calidad.

Algunas de las características que posee son (Menéndez Verde, et al., 2011):

- ❖ Es flexible y proporciona funcionalidades para el cálculo varias métricas.
- ❖ Brinda soporte para 15 métricas objetivas y más de 5 *plugins*.
- ❖ Brinda soporte para más de 30 formatos de video, incluyendo aquellos con 10, 14 y 16 bits de profundidad de color.

La herramienta se desarrolló en dos versiones: privativa y libre. La opción de pago posee funcionalidades completas y útiles pero sería necesario comprar la licencia. Aunque esto fuera posible, en la actualidad Cuba se encuentra en un proceso de migración a software libre, por lo que sería más conveniente desarrollar un nuevo sistema que utilizar uno privativo.

¹ Abreviatura de *codificador-decodificador*. Describe una especificación desarrollada en software, hardware o una combinación de ambos, capaz de transformar un archivo con un flujo de datos (*stream*) o una señal.

² Aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

En el caso de la opción libre, aunque la descarga es posible, la mayoría de los *plugins* están inhabilitados por completo. Además la arquitectura y composición de este software no permite agregar funcionalidades para adaptarlo a las necesidades del STCV.

Se realizó, además, el análisis de otros sistemas como **VQM (Video Quality Measurement PC)** pero de igual forma es privativo, lo que interfiere en la obtención de la licencia. Aunque fuese posible comprar el software, el código fuente no estaría disponible para realizar los cambios necesarios que posibiliten integrarlo al sistema STCV.

1.3 Clasificación de los métodos para evaluar la calidad en videos digitales

De manera general los métodos para evaluar la calidad de video se clasifican en (Joskowicz, et al., 2012):

- ❖ **Métodos subjetivos:** la calidad de una imagen o un video es verificada por un conjunto de personas que opinan acerca de lo que observan. La principal desventaja que tienen es que dependen de la percepción individual de cada persona sobre la calidad del video.
- ❖ **Métodos objetivos:** analizan con fiabilidad la calidad percibida, en base a medidas objetivas tomadas en lugares aleatorios del video por el sistema que realiza el análisis. Se clasifican de acuerdo a la disponibilidad de la señal original y tienen como objetivo analizar particularidades que la percepción humana pueda pasar por alto al analizar la calidad de video.

Como se plantea anteriormente, los métodos objetivos se clasifican según la disponibilidad de la señal original. Las categorías de dicha clasificación son (Baz Hormigos, 2009):

- ❖ **Referencia Completa (FR - Full Reference):** realizan la evaluación de la calidad del video tomando como base el video original.
- ❖ **Referencia Reducida (RR – Reduced Reference):** realizan la evaluación de la calidad del video a partir de características del video original.
- ❖ **Sin Referencia (NR - Null Reference):** realizan la evaluación de la calidad del video a ciegas, es decir, solo emplean al propio video para emitir un criterio sobre la calidad del mismo.

El desarrollo de la solución propuesta está basado en los métodos objetivos, específicamente en la técnica “Referencia Completa”. Esto se debe principalmente a las necesidades identificadas por el proyecto STCV.

1.3.2 Posibles distorsiones en videos digitales

El sistema visual humano es extremadamente complejo, y puede detectar fácilmente ciertos tipos de distorsiones, mientras que puede pasar por alto otras, dependiendo de diversos factores. Algunos tipos de distorsiones son (Joskowicz, et al., 2012):

- ❖ Efecto de imagen de base.
- ❖ Efecto escalera.
- ❖ Patrones de mosaico.
- ❖ Bordes falso.
- ❖ Efecto de bloque.
- ❖ Borrosidad.

El “Efecto de bloque” es una distorsión que ocurre comúnmente en las transmisiones de videos digitales. Además cuando ocurre, generalmente provoca “Borrosidad”. Este comportamiento fue identificado durante la investigación y verificado con miembros del proyecto STCV.

En esta investigación se tiene como alcance evaluar la calidad de videos digitales, tomando como base la presencia de las distorsiones “Efecto de bloque” y “Borrosidad”, y utilizando una técnica de “Referencia Completa”. A continuación se describen detalladamente dichas distorsiones:

Efecto de bloque

Esta posiblemente sea la degradación que más se nota en un video digital (Fig 1). Tiene su origen en la codificación basada en bloques que segmenta la imagen en áreas pequeñas realizando una transformación de cada una de ellas de forma independiente. El “Efecto de bloque” se presenta como discontinuidades en los bordes de bloques adyacentes al reconstruir la imagen (Menéndez Verde, et al., 2011).

Dentro de un mismo cuadro, cuanto más “gruesa” sea la cuantificación realizada, más visible es el “Efecto de bloque”. El umbral de cuantificación a partir del cual es percibido el “Efecto de bloque” depende del tipo de imagen y del movimiento, por lo que no es posible definir un valor estándar e independiente de otros factores (Joskowicz, 2008).



Figura 1. Demostración del "Efecto de bloque".

Borrosidad o falta de definición

La falta de definición, se manifiesta como una pérdida de los detalles de la imagen (Fig 2). Si bien esto puede estar dado por imágenes tomadas fuera de foco, también puede ser un efecto introducido por el proceso de digitalización. Esta degradación también puede aportar al "Efecto de bloque" y al de "Mosaico" (Joskowicz, 2008).



Figura 2. Demostración de "Borrosidad".

1.4 Algoritmos para la detección de la distorsión "Efecto de bloque"

En (Gandam, et al., 2010) se propone un algoritmo para eliminar el "Efecto de bloque" que se genera al comprimir imágenes al formato JPEG. Dicho algoritmo se compone de dos etapas: 1) detectar los valores de esquina atípicas en la imagen; 2) reducir los valores de esquina atípicas. En la primera etapa se identifican las esquinas atípicas

de los cuadrantes generados al comprimir la imagen, teniendo en cuenta la vecindad de los píxeles. Posteriormente en la segunda etapa el valor de los píxeles que son detectados como esquinas atípicas es remplazado por una suma ponderada de los píxeles de su vecindad.

En (Santos, et al., 2011) se plantea un algoritmo para reducir el “Efecto de bloque” tomando como principio el funcionamiento del sistema visual humano. Como parte de dicho algoritmo se define una métrica de percepción visual denominada “Índice de efecto visual en bloque”. Esta métrica está basada en la pérdida de criterios de correlación en regiones entre bloques y en la distorsión de la luminancia en la zona media del video. Además tiene en cuenta la sensibilidad al contraste del sistema visual humano.

En dicho método se aplica un filtro adaptativo que actúa de acuerdo a la visibilidad de las regiones de bloque en la imagen. La intensidad del “Efecto de bloque” es calculada teniendo en cuenta la variación del brillo en los píxeles ubicados en la frontera de los bloques.

En (Yanhui, y otros, 2013) se plantea un algoritmo para detectar el “Efecto de bloque” mediante el análisis de las regiones planas de la imagen y de los bordes. Primeramente se detectan los bordes de la imagen utilizando el operador de Sobel (C. Gonzalez, y otros, 2007). Luego se realiza un análisis para eliminar los falsos puntos de borde y a continuación se calculan dos indicadores llamados en dicho artículo “Tasa de bloque” e “Intensidad de bloque”. Finalmente se retorna como resultado un promedio ponderado de dichas tasas.

En (Roddick, y otros, 2014) se propone un algoritmo del tipo “Sin referencia” para evaluar el “Efecto de bloque” en una imagen. El mismo está basado en el principio de que la existencia del “Efecto de bloque” tiene un impacto directo en la distribución de los momentos de Tchebichef. Primeramente la imagen es dividida en bloques de forma tal que se cubran las regiones ruidosas que puedan existir potencialmente. Por cada región se extrae el momento de Tchebichef y se utiliza el mismo como un descriptor de la región. Posteriormente, a partir de los momentos, se calcula un indicador local para la región, y luego se combinan los indicadores de todas las regiones para emitir un criterio que represente a toda la imagen.

En (Fu, y otros, 2004), (Yanhui, y otros, 2013), (Muijs, y otros, 2005) y (Xingang, y otros, 2008) se plantean algoritmos para evaluar el “Efecto de bloque”. Todos tienen como rasgo común que dividen la imagen en bloques de 8x8 y posteriormente analizan la visibilidad y la suavidad de los mismos.

Durante el estudio de los algoritmos planteados anteriormente se pudo apreciar que la mayoría basan su funcionamiento, de una u otra forma, en el surgimiento de bloques en la imagen como resultado de la existencia

del “Efecto de bloque”. Además se apreció que en múltiples ocasiones los métodos se basan en la aparición de los bordes que delimitan a los bloques. Esto sugiere la utilización de los detectores de bordes para desarrollar un algoritmo que permita evaluar el “Efecto de bloque” en una imagen. La principal ventaja de esta variante es que la teoría asociada a los detectores de bordes está fundamentada sólidamente. Además los detectores de bordes son fáciles de implementar y poco costosos computacionalmente.

1.5 Algoritmos para la detección de la distorsión “Borrosidad”

En (Kerouh, et al., 2011) se plantea un algoritmo para medir cuán borrosa es una imagen. El mismo está basado en el análisis de los bordes y de la transformada Wavelet-Harr. En (Marziliano, et al., 2010) se plantea otro algoritmo para detectar la borrosidad y también está basado en el análisis de los bordes. Primeramente se aplica un detector de borde vertical y luego se recorre cada fila para encontrar los píxeles de borde inicial y final. Luego se utiliza esa información para calcular la borrosidad local y global.

En (Ladret, et al., 2008) se plantea un método muy novedoso, del tipo “Sin Referencia”, para evaluar la borrosidad en una imagen. El mismo está basado en el funcionamiento del sistema visual humano, que es poco sensible a los cambios de borrosidad en una misma imagen. Tiene como ventaja que evade la alta sensibilidad ante el ruido, típica de métodos similares que están basados en la detección de bordes. El método evoluciona del modo que sigue. Primeramente se le aplica un filtro paso-bajo a la imagen original para generar borrosidad en la misma. Luego la imagen borrosa y la original son procesadas para calcular la variación de la intensidad de los píxeles, teniendo en cuenta su vecindad. Finalmente se utiliza dicha variación para emitir un criterio sobre la borrosidad presente en la imagen.

En (Marichal, y otros, 1998) se plantea un algoritmo basado en el análisis de la Transformada Discreta del Coseno (DCT). El uso de la DCT para detectar si una imagen es borrosa, parte del hecho de que una imagen sin bordes carece de altas frecuencias (Blasco, 2008). Primeramente la imagen en escala de grises se divide en bloques de 8x8 píxeles y se calcula la DCT de dichos bloques. Luego se calcula un histograma a partir de dichos bloques, teniendo en cuenta solamente los píxeles que superen cierto umbral. A partir de dicho histograma y de una matriz de pesos, se calcula si la imagen es borrosa o no.

Durante el estudio de los algoritmos existentes se pudo apreciar que la mayoría utilizan de un modo u otro, la información de los bordes presentes en la imagen. Como un método novedoso se identificó el propuesto en (Ladret, et al., 2008), el cual está basado en el funcionamiento del sistema visual humano y tiene como mérito que evade la alta sensibilidad ante el ruido típica de los métodos basados en bordes.

1.6 Metodología de desarrollo

Una metodología para el desarrollo de software es un modo sistemático de realizar, gestionar y administrar un proyecto. Comprende los procesos a seguir sistemáticamente para idear, implementar y mantener un producto software desde que surge la necesidad del producto hasta que se cumple el objetivo por el cual fue creado. Se basan en una combinación de los modelos de proceso genéricos (cascada, incremental, entre otros) y definen artefactos, roles y actividades (Laboratorio Nacional de Calidad del Software, 2009).

Existen distintos tipos de metodologías de desarrollo, entre las que se encuentran las tradicionales. Estas metodologías son exquisitas en la documentación que generan y recorren cada fase del desarrollo de software, sugiriendo un conjunto de artefactos que ayudan al desarrollo de la fase siguiente. Con su utilización se obtiene como resultado un producto desarrollado bajo buenas prácticas y respaldado en su totalidad por una detallada documentación. Entre las metodologías tradicionales se puede mencionar el Proceso Unificado de Desarrollo de Software (RUP, por sus siglas en inglés) como una de las más conocidas.

El proyecto donde se desarrolló el subsistema de monitorización del STCV utiliza la metodología RUP. Debido a que el componente propuesto debe integrarse con este subsistema, se consideró utilizar la misma metodología de desarrollo. Por este motivo no se realiza la comparación con otras metodologías, sino que se describen las características específicas de RUP.

RUP

Esta metodología proporciona un enfoque disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su meta es asegurar la producción de software de alta calidad que resuelva las necesidades de los usuarios dentro de los presupuestos y tiempos establecidos (Salgado Magaña, et al., 2011).

Las principales características de esta metodología son (Jacobson, et al., 2000):

- ❖ Dirigida por casos de uso.
- ❖ Iterativa e incremental.
- ❖ Centrada en la arquitectura.

Además define los flujos de trabajo Requisitos, Análisis, Diseño, Implementación y Pruebas. Cada flujo recorre las fases Inicio, Elaboración, Construcción y Transición (Jacobson, et al., 2000).

1.7 Herramientas y tecnologías

Se utilizaron las herramientas y tecnologías definidas en el proyecto STCV para facilitar la integración del componente con los demás artefactos desarrollados por el proyecto. Además para aprovechar la experiencia del equipo de proyecto en dicho entorno tecnológico.

1.7.1 Lenguaje Unificado de Modelado

El Lenguaje Unificado de Modelado (UML) es un lenguaje estándar de modelado para software que permite visualizar, especificar, construir y documentar los artefactos que se crean durante el proceso de desarrollo del software (Jacobson, et al., 2000). Especifica diagramas que permiten la modelación de los diferentes componentes que integran el sistema. UML es utilizado por distintas metodologías de desarrollo. Además presenta una estrecha relación con el Proceso Unificado de Desarrollo de Software (RUP), que es la metodología a utilizar en la presente investigación (Salgado Magaña, et al., 2011).

Otras de las características de UML son:

- ❖ Indica lo que supuestamente hará el sistema pero no cómo lo hará.
- ❖ Incluye estereotipos como mecanismo de extensibilidad.
- ❖ Puede describir cualquier tipo de sistemas mediante diagramas.

En la presente investigación se utiliza UML para diseñar los diagramas indicados por la metodología de desarrollo RUP. Se utiliza este lenguaje por su fácil integración con la mayoría de las herramientas CASE³ existentes y las facilidades que brinda para representar las entidades, componentes o partes en general del producto que se está diseñando.

1.7.1 Visual Paradigm

Visual Paradigm es una herramienta CASE que soporta el modelado mediante UML y proporciona asistencia a los analistas, ingenieros de software y desarrolladores, durante todo el ciclo de vida de un software. Entre las ventajas que presenta se encuentran (Software., 2013):

- ❖ **Dibujo:** facilita el modelado de UML, ya que tiene herramientas específicas para ello.
- ❖ **Corrección sintáctica:** controla que el modelado con UML sea correcto.
- ❖ **Coherencia entre diagramas:** al disponer de un repositorio común, es posible visualizar el mismo elemento en varios diagramas, evitando duplicidades.

³ Ingeniería de software asistida por computadora.

- ❖ **Reutilización:** facilita la reutilización, ya que dispone de una herramienta centralizada donde se encuentran los modelos utilizados por otros proyectos.
- ❖ **Generación de código:** permite generar código de forma automática, reduciendo el tiempo de desarrollo y evitando errores en la codificación del software.

Al utilizar la metodología de desarrollo RUP, la documentación del componente será sustentada principalmente por el diseño de diagramas UML. La utilización de esta herramienta facilitará el modelado de dichos diagramas.

1.7.3 Lenguaje de Programación C++

Los lenguajes de programación son idiomas artificiales compuestos por reglas semánticas y sintácticas escritas mediante símbolos que facilitan la conversión del lenguaje natural al lenguaje de máquina (Wilson, 1993).

El lenguaje de programación C++ surgió como una extensión del lenguaje C y se comenzó a desarrollar en 1980. En la actualidad es un lenguaje versátil que mantiene las ventajas del C en cuanto a riqueza de operadores, expresiones, flexibilidad y eficiencia. Este lenguaje es orientado a objetos y multiplataforma (González, 2008).

Ventajas de C++ (Yerovi , et al., 2013):

- ❖ Brinda soporte para la programación orientada a objetos.
- ❖ Es un lenguaje muy flexible que permite programar con múltiples estilos.
- ❖ Posee un conjunto reducido de palabras claves.
- ❖ Posee tipos de datos agregados que permiten que se combinen con los tipos nativos y se manipulen como un todo.
- ❖ Permite el acceso a memoria mediante el uso de punteros.

Además existe una amplia documentación sobre el lenguaje y una comunidad de desarrolladores muy madura.

1.7.4 Marco de trabajo Qt

Los *frameworks* o marcos de trabajo por su significado en español, se refieren a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un *framework* se puede considerar como una aplicación genérica incompleta y configurable a la que se le pueden añadir las últimas piezas para construir una aplicación concreta (Pachacama, et al., 2010).

Qt es conocido como el *framework* por excelencia cuando se elige desarrollar con C++. Posee un amplio conjunto de funcionalidades predefinidas que permiten crear productos con mayor facilidad. Las aplicaciones construidas sobre Qt se ejecutan de forma nativa, compiladas desde el código fuente, en la mayoría de los sistemas operativos existentes, incluyendo las variantes que cada uno de ellos posea. El desarrollo con Qt “aumenta el rendimiento,

brinda independencia de plataforma, aumenta la robustez del software y permite una fácil integración con bibliotecas y productos de terceros” (Pérez Míñquez, et al., 2005).

La utilización de este *framework* posibilita el desarrollo estandarizado del componente abordado en esta tesis. Además posibilita desarrollar el mismo con mayor rapidez y menor esfuerzo.

1.7.5 Entorno de Desarrollo Integrado (IDE) QtCreator

Una vez que se tienen definidos el lenguaje de programación y el *framework* a utilizar, se debe seleccionar el entorno de desarrollo integrado (IDE) que mejor se ajuste a los mismos.

Qt Creator es un IDE para desarrollar interfaces gráficas de usuario y también para el desarrollo de programas sin interfaz gráfica como herramientas de la consola y servidores. Cuenta aproximadamente con 500 clases, más de 9000 funciones y 500.000 líneas de código. Es distribuido bajo los términos de GNU General Public License, es software libre y es de código abierto (Corporation, 2011).

Las características principales de QtCreator son (Vega González, et al., 2013):

- ❖ Permite desarrollar aplicaciones Qt de manera rápida y fácil.
- ❖ Cuenta con dos editores visuales, *QtDesigner* para diseñar interfaces de usuario a partir de *QtWidgets*, y *Qt Quick Designer* para el desarrollo de interfaces de usuario animadas.
- ❖ Contiene un sofisticado editor de código que proporciona completamiento de código y ayuda de contexto para el lenguaje C++.
- ❖ Permite generar, ejecutar e implementar proyectos de Qt dirigidos a plataformas móviles y de escritorio.

1.7.6 Biblioteca OpenCV

OpenCV es una biblioteca de código abierto para C/C++ desarrollada inicialmente por Intel para el procesamiento de imágenes y la visión por computadora. Su primera versión estable fue liberada en 2006. Está disponible para los sistemas operativos Linux, Mac, y Windows. Permite la visualización de datos de manera sencilla y la obtención de información de imágenes y videos. Posee funciones de captura y presentación de imágenes.

La biblioteca OpenCV está compuesta por los siguientes módulos (Furfaro, 2010):

- ❖ **cv**: contiene las funciones principales de la biblioteca.
- ❖ **cvaux**: contiene las funciones auxiliares (experimental).
- ❖ **cxcore**: contiene las estructuras de datos y funciones de soporte para Álgebra lineal.
- ❖ **highgui**: funciones para manejar la interfaz gráfica de usuario.
- ❖ **imgproc**: contiene las principales funciones para la manipulación de imágenes.

- ❖ **core:** contiene las funcionalidades principales de la biblioteca, como son las estructuras de datos básicas y las funciones aritméticas.

1.7.8 Biblioteca XML-RPC

XML-RPC es un protocolo que utiliza XML⁴ para gestionar las llamadas de los métodos y sus resultados. Es una colección de implementaciones que permiten que el software se ejecute en distintos sistemas operativos y en distintos entornos de desarrollo. RPC significa llamada a procedimiento remoto. Sus principales características son (Slonneger, 2006):

- ❖ Está diseñado para ser lo más simple posible.
- ❖ Codifica mensajes para llamar a métodos y describir sus resultados mediante un vocabulario estándar, utilizando XML.
- ❖ Tiene un vocabulario limitado de etiquetas XML para describir los tipos de parámetros y el tipo de valor de retorno en una función.

1.8 Conclusiones parciales

En este capítulo se concluye que:

- ❖ No existe un componente que pueda ser integrado al STCV para monitorizar la calidad de video durante las transmisiones.
- ❖ Los algoritmos para la detección del “Efecto de bloque” aprovechan el hecho de que dicho efecto genera muchos bordes en la imagen.
- ❖ Los algoritmos para la detección de “Borrosidad” tienen en cuenta que dicho efecto disminuye la intensidad de los bordes presentes en la imagen. Además tienen en cuenta el funcionamiento del sistema visual humano.
- ❖ Las herramientas y tecnologías seleccionadas están en correspondencias con las utilizadas en el proyecto STCV, lo cual facilita la integración del componente a desarrollar con el entorno existente.

⁴ *eXtensible Markup Language* ('lenguaje de marcas extensible'), es un lenguaje de marcas utilizado para almacenar datos en forma legible.

Capítulo 2: Diseño de la solución propuesta

En el presente capítulo se describen los algoritmos propuestos para detectar “Efecto de bloque” y “Borrosidad”. Además se describen los requisitos y se muestran diagramas asociados a la fase de diseño. También se describen los patrones utilizados durante el desarrollo del componente y se muestra en estándar de codificación seleccionado.

2.1 Algoritmos propuestos

Debido que en esta tesis se abordan métodos del tipo “Referencia Completa” en transmisiones de video, es necesario realizar una sincronización entre los fotogramas del video original y el video recibido. Este proceso es descrito a continuación.

Primeramente se selecciona el primer fotograma de la transmisión recibida y se comienza a buscar en el video original a partir del inicio. Una vez encontrado, su posición representa la cantidad de fotogramas perdidos al recibir la transmisión, la cual se utiliza para hacer corresponder los fotogramas claves del video original con los fotogramas claves del video recibido. Debe decirse que en el STCV, por cada video, se almacena el índice de los fotogramas claves, los cuales están disponibles para cualquier componente de la misma.

A continuación se muestra de manera gráfica lo descrito anteriormente.

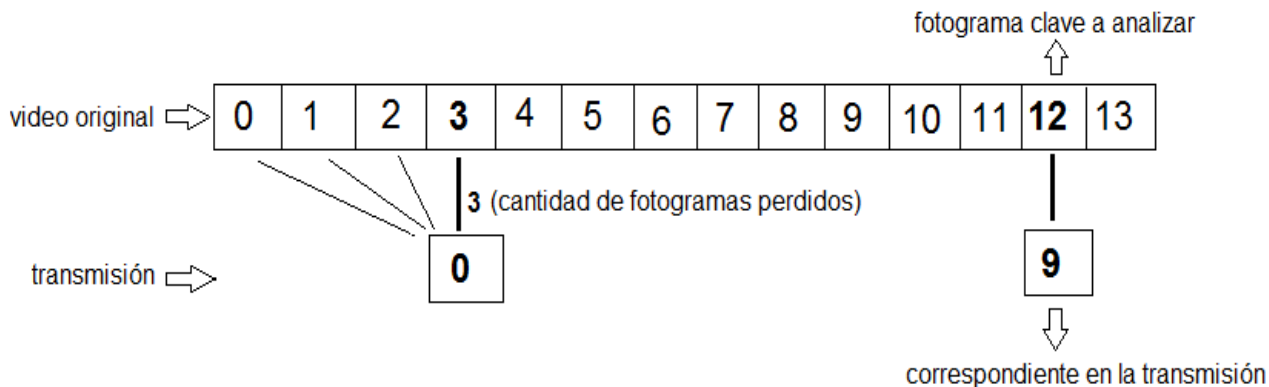


Figura 3. Sincronización entre el video original y el video recibido.

Una vez que se estima el índice del fotograma clave en el video original, este no es tomado como definitivo. En su lugar, se analizan los fotogramas en la vecindad del mismo y se elige el que tiene más similitud con el fotograma correspondiente en el video recibido.

Por ejemplo, en la figura 3 se evidencia que hay 3 fotogramas perdidos, por tanto el fotograma 12 en el video original se corresponde con el fotograma 9 en el video recibido. De acuerdo a lo planteado en el párrafo anterior el fotograma 12 no sería tomado como definitivo, sino que se seleccionaría el más similar a él en su vecindad. En la figura 4 se muestra esto último utilizando una vecindad de tamaño igual a tres fotogramas.

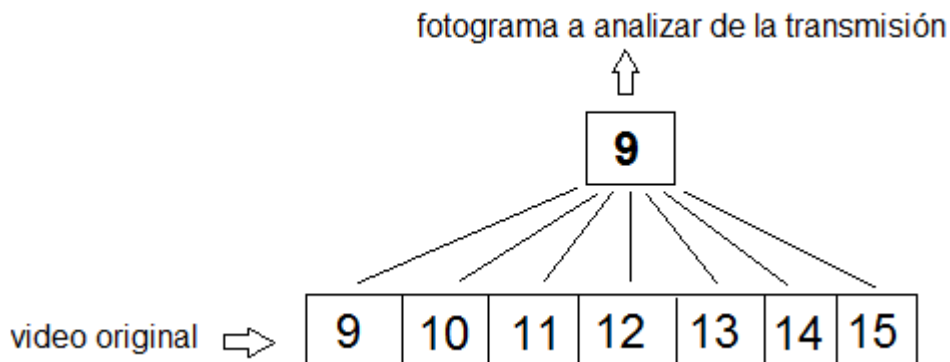


Figura 4. Vecindad del fotograma clave en el video original.

Aclarar que el proceso de búsqueda está basado en el detector de características SURF, en un proceso de alineación de los rasgos, y en la medida de similitud Ochiai-I (Tappert, y otros, 2010).

2.1.1 Algoritmo propuesto para detectar “Efecto de bloque”

Primeramente se convierten los dos fotogramas bajo análisis a escala de grises y posteriormente se detectan los bordes utilizando el operador de *Canny* (C. Gonzalez, et al., 2007). Luego se aplica la ecuación 1 (Tappert, et al., 2010) para medir el “Efecto de bloque” presente en la imagen. Dicha ecuación arroja un valor en el intervalo [0,1] donde 0 significa la ausencia total de “Efecto de bloque” y 1 la presencia total de “Efecto de bloque”.

$$D_{lance\&willieams} = \frac{b+c}{2a+b+c} \quad (\text{ecuación 1})$$

Donde:

- ❖ a es la cantidad de puntos de bordes que están presentes en ambas imágenes.
- ❖ b es la cantidad de puntos de bordes presentes en la imagen distorsionada que no están presentes en la imagen original.
- ❖ c es la cantidad de puntos de bordes presentes en la imagen original que no están presentes en la imagen distorsionada.

2.1.2 Algoritmo propuesto para detectar “Borrosidad”

El algoritmo propuesto para detectar “Borrosidad” es una variación al algoritmo de tipo “Sin referencia” propuesto en (Ladret, et al., 2008). La variación es que en vez de generar una imagen borrosa, se utiliza la imagen recibida.

En lo que sigue considérese que:

- ❖ F es el fotograma en el video original.
- ❖ B es el fotograma correspondiente en el video recibido.
- ❖ m es la cantidad de filas del fotograma.
- ❖ n es la cantidad de columnas del fotograma.

Primeramente se calculan las diferencias vertical y horizontal en el fotograma del video original y en el fotograma del video recibido (ecuación 2).

$$\begin{aligned}
 D_{FVer}(i,j) &= Abs(F(i,j) - F(i-1,j)) \quad \text{for } i=1 \text{ to } m-1, j=0 \text{ to } n-1 \\
 D_{FHor}(i,j) &= Abs(F(i,j) - F(i,j-1)) \quad \text{for } j=1 \text{ to } n-1, i=0 \text{ to } m-1 \\
 D_{BVer}(i,j) &= Abs(BVer(i,j) - BVer(i-1,j)) \quad \text{for } i=1 \text{ to } m-1, j=0 \text{ to } n-1 \\
 D_{BHor}(i,j) &= Abs(BHor(i,j) - BHor(i,j-1)) \quad \text{for } j=1 \text{ to } n-1, i=0 \text{ to } m-1
 \end{aligned}
 \tag{ecuación 2}$$

Luego se calcula la varianza entre píxeles (ecuación 3).

$$\begin{aligned}
 WVer &= Max(0, D_{FVer}(i,j) - D_{BVer}(i,j)) \quad \text{for } i=1 \text{ to } m-1, j=1 \text{ to } n-1 \\
 VHor &= Max(0, D_{FHor}(i,j) - D_{BHor}(i,j)) \quad \text{for } i=1 \text{ to } m-1, j=1 \text{ to } n-1
 \end{aligned}
 \tag{ecuación 3}$$

Posteriormente con el fin de comparar las variaciones de la imagen original, se calcula la suma de los coeficientes de D_{FVer} , D_{FHor} , D_{VVer} , D_{VHor} de la forma descrita a continuación (ecuación 4):

$$\begin{aligned}
 s_{FVer} &= \sum_{i,j=1}^{m-1,n-1} D_{FVer}(i,j) & s_{FHor} &= \sum_{i,j=1}^{m-1,n-1} D_{FHor}(i,j) \\
 s_{VVer} &= \sum_{i,j=1}^{m-1,n-1} D_{VVer}(i,j) & s_{VHor} &= \sum_{i,j=1}^{m-1,n-1} D_{VHor}(i,j)
 \end{aligned}
 \tag{ecuación 4}$$

Finalmente se normaliza el resultado en el rango [0,1] (ecuación 5). Luego se selecciona como valor estimado de la “Borrosidad”, al mayor valor entre b_{FVer} y b_{FHor} (ecuación 6).

$$b_{_FVer} = \frac{s_{_FVer} - s_{_VVer}}{s_{_FVer}} \qquad b_{_FHor} = \frac{s_{_FHor} - s_{_VHor}}{s_{_FHor}} \qquad \text{(ecuación 5)}$$

$$blur_F = \text{Max}(b_{_FVer}, b_{_FHor}) \qquad \text{(ecuación 6)}$$

En la Figura 5 se muestra de manera gráfica lo descrito anteriormente

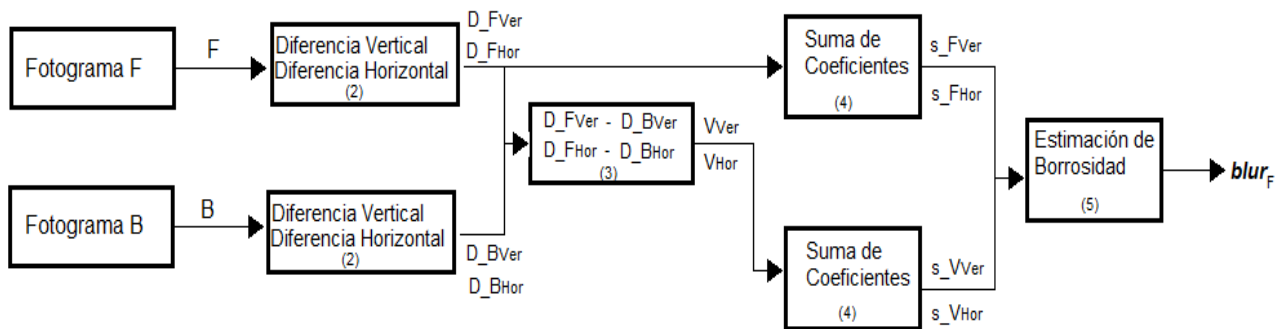


Figura 5. Evolución del algoritmo propuesto para detectar “Borrosidad”.

2.3 Solución propuesta

Se propone un componente que implemente los algoritmos de detección de “Borrosidad” y “Efecto de bloque” descritos anteriormente, con el objetivo de monitorizar la calidad de video en las transmisiones del STCV. Debe decirse que aunque en los algoritmos anteriormente descritos se hace uso de los fotogramas claves, estos no son extraídos por el componente propuesto en esta investigación, sino que dicha responsabilidad la tiene otro componente del STCV. Esto permite que cualquier componente de análisis del STCV, que haga uso de fotogramas claves, pueda conocer dicha información fácilmente.

2.4 Especificación de requisitos

El propósito fundamental de la especificación de requisitos es guiar el desarrollo hacia el sistema correcto. Esto se consigue mediante la descripción de las capacidades o condiciones que el sistema debe cumplir, indicándole al desarrollador qué debe hacer y qué no debe hacer el sistema (Jacobson, et al., 2000). Los requisitos se dividen en dos tipos: Funcionales y No Funcionales. Para determinarlos es necesario utilizar algún modelo específico que apoye esta acción.

Cuando se utiliza la metodología RUP se aconseja realizar el modelo de negocio y/o el modelo de dominio, según corresponda, para conocer los requisitos del sistema a desarrollar.

2.4.1 Modelo de dominio

El modelo de dominio “captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las “cosas” que existen o los eventos que suceden en el entorno en el que trabaja el sistema. La forma de desarrollar este modelo es a través de diagramas UML y su objetivo principal es mostrar a los desarrolladores, usuarios, clientes u otras personas que interactúan con el sistema, las clases del dominio y las relaciones que existen entre ellas” (Jacobson, et al., 2000).

Para el desarrollo de la presente investigación se realizó un modelo de dominio para mostrar de manera visual los principales conceptos asociados a la situación problemática planteada.

2.4.2 Diagrama de clases del Modelo de Dominio

El Especialista es la persona encargada de visualizar el comportamiento de la transmisión a través del Subsistema de Monitorización. Básicamente el subsistema posee dos funcionalidades:

1. **Monitorizar el flujo de video:** el Subsistema de Monitorización monitoriza el flujo de video transmitido por el Subsistema de Transmisión.
2. **Notificar errores:** el Subsistema de Monitorización detecta y notifica los errores relacionados con la pérdida de paquetes en el flujo transmitido.

El Subsistema de Transmisión accede al NFS, que es quien contiene los videos que serán transmitidos.

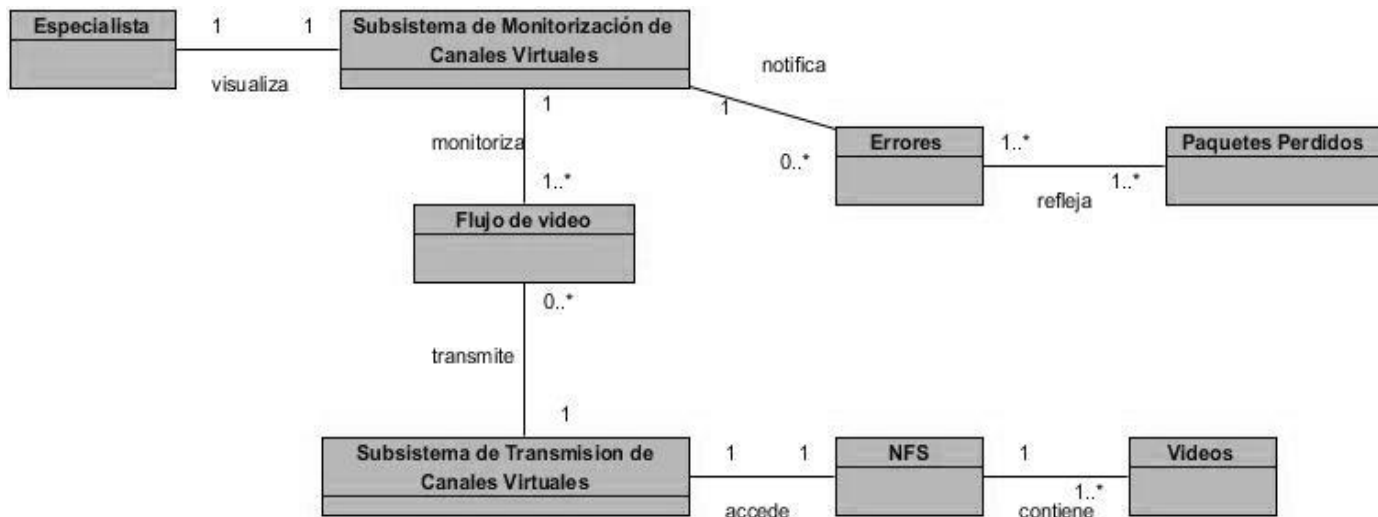


Figura 6. Modelo de dominio.

Glosario de Términos del Modelo de Dominio.

- ❖ **NFS:** es un protocolo de nivel de aplicación, según el Modelo OSI. Es utilizado para sistemas de archivos distribuido en un entorno de red de computadoras de área local. Posibilita que distintos sistemas conectados a una misma red accedan a ficheros remotos como si se tratara de ficheros locales. En el caso de la investigación se refiere a un servidor que implementa dicho protocolo. Este servidor almacena los videos a transmitir.
- ❖ **Flujo de video:** es la distribución digital de multimedia a través de una red de computadoras de manera que el usuario consume el producto, generalmente un archivo de video o audio, mientras se descarga.

2.4.3 Requisitos del componente

Requisitos funcionales

“Los requisitos no funcionales son las restricciones de los servicios o funciones ofrecidas por el sistema, restringen el espacio de posibles soluciones. Debe pensarse en estos, como las características que hacen al producto atractivo, usable, rápido o confiable” (Pressman, 2010).

Los requisitos funcionales del componente son:

- ❖ **RF1 Evaluar calidad de video digital:** el sistema debe permitir evaluar la calidad de un video digital basándose en la detección de las distorsiones “Borrosidad” y “Efecto de Bloque”, tomando como referencia el video original.
 - Entrada: los datos para la comunicación con el Subsistema de Transmisión.
 - Salida: existencia o no de las distorsiones evaluadas.
- ❖ **RF2 Detener evaluación de calidad de video digital:** el sistema debe permitir detener la evaluación de calidad.
 - Entrada: no procede.
 - Salida: no procede.

Requisitos no funcionales

“Los requisitos funcionales son declaraciones de servicios que el sistema debe proporcionar, definen la manera en que éste debe reaccionar a determinadas entradas y cómo se debe comportar en situaciones particulares. Estos requisitos son las características fundamentales del sistema y expresan la capacidad de acción del mismo” (Pressman, 2010).

Los requisitos no funcionales del componente son:

- ❖ **RNF de hardware:** se requiere para el funcionamiento del componente una computadora con un microprocesador Core i3 a 2.40 GHz como mínimo. Debe, además, contar con 2GB como mínimo de memoria RAM.
- ❖ **RNF de portabilidad:** el componente se debe poder ejecutar en Ubuntu.
- ❖ **RNF de restricciones del diseño e implementación:** el componente debe ser implementado con el lenguaje de programación C++ utilizando el *framework* Qt, el IDE *QtCreator* y las bibliotecas *OpenCV*. Debe permitir integrarse al Subsistema de Monitorización de Canales Virtuales mediante el uso de una interfaz de comunicación que sirve para recibir y enviar información en ambas direcciones.

2.5 Análisis de la solución

La fase de análisis en la metodología RUP se describe como la fase que “analiza los requisitos capturados, refinándolos y estructurándolos. El objetivo de hacerlo es conseguir una comprensión más precisa de los requisitos y una descripción de los mismos que sea fácil de mantener y que ayude a estructurar el sistema entero; incluyendo su arquitectura” (Jacobson, et al., 2000). Esta fase se apoya fundamentalmente en el Modelo de análisis.

Según lo planteado en (Jacobson, et al., 2000) se puede prescindir del análisis cuando los requisitos son analizados y comprendidos por el equipo de desarrollo durante su captura, además deben estar formalmente identificados y reflejados en el modelo de Casos de uso del sistema, con el objetivo de que el cliente sea capaz de comprender los resultados y de que no exista la necesidad de refinarlos posteriormente.

Debido a que existe gran coincidencia entre lo planteado en el párrafo anterior y las características de equipo de desarrollo, en esta investigación se prescindirá del análisis, haciéndose directamente el diseño

2.6 Diseño de la solución

“El diseño es el centro de atención al final de la fase de Elaboración y comienzo de las iteraciones de Construcción. Esto contribuye a una arquitectura estable y sólida y a crear un plano del modelo de implementación” (Jacobson, et al., 2000).

A continuación se describen los elementos del diseño que intervienen en el desarrollo del componente propuesto en esta tesis.

2.6.1 Descripción de los actores

A continuación se describe el único actor que interactúa con el sistema:

- ❖ Subsistema de monitorización: interactúa con el componente propuesto en la investigación. Inicia el caso de uso “Evaluar calidad de video digital”, a través del cual se desencadena el proceso de evaluación de la calidad.

2.6.2 Modelo de casos de uso del sistema

“Un caso de uso especifica el comportamiento de un sistema o de una parte del mismo y es una descripción de un conjunto de secuencias de acciones, incluyendo variantes, que ejecuta un sistema para producir un resultado observable de valor para un actor. Proporciona un medio para que los desarrolladores, usuarios finales del sistema y los expertos del dominio, lleguen a una comprensión común del sistema” (Jacobson, et al., 1999).

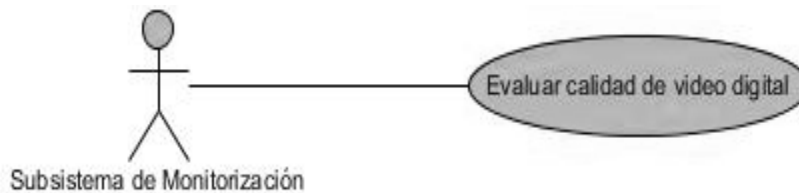


Figura 7. Diagrama de caso de uso de la solución propuesta.

Descripción de casos de uso del sistema

A continuación se describe el único caso de uso del componente.

Tabla 1. Caso de uso “Evaluar calidad de video digital”.

Objetivo	Evaluar la calidad de un video digital basándose en la detección de las distorsiones “Efecto de bloque” y “Borrosidad”, aplicando un enfoque del tipo “Referencia completa”
Actores	Subsistema de monitorización (Inicia)
Resumen	El caso de uso inicia cuando el Subsistema de monitorización hace la petición de “Evaluar Calidad”. Se procede a evaluar la calidad del video recibido basándose en la detección de las distorsiones “Efecto de bloque” y “Borrosidad”, y aplicando un enfoque del tipo “Referencia completa”.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	Debe haberse almacenado previamente en la base de datos el índice de los fotogramas claves del video original.	
Poscondiciones	Ninguna	
Flujo de eventos		
Flujo básico: Evaluar la calidad de un video digital.		
Paso	Actor	Sistema

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

1.		Selecciona "Evaluar calidad".	
1.1		Se captura el flujo del video en transmisión.	
1.2		Se guardan los fotogramas capturados.	
1.3		Se carga el video original.	
1.4		Se sincroniza el video original con el flujo recibido.	
1.5		Se detecta la presencia de las distorsiones "Borrosidad" y "Efecto de bloque" en los fotogramas recibidos, tomando como referencia los fotogramas del video original.	
1.6		Se notifican los resultados obtenidos al Subsistema de Monitorización.	
2.		Selecciona la opción "Detener"	
2.1		Se detiene la captura del flujo del video en transmisión y la evaluación de la calidad.	
2.2		Finaliza el caso de uso.	
Relaciones		CU Incluidos	No procede.
		CU Extendidos	No procede.
Requisitos funcionales	no	Ninguno	
Asuntos pendientes		No procede.	

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

2.6.3 *Arquitectura del Sistema*

“La arquitectura de software es imprescindible a la hora de guiar el desarrollo de un software complejo, pues su diseño, especificación y entendimiento resulta muy difícil para un solo individuo. Además durante el progreso del sistema, facilita el desarrollo simultáneo de componentes y permite detectar errores de diseño en fases tempranas” (Camacho, et al., 2004).

El componente utiliza una arquitectura de tipo N-capas. Esta arquitectura se caracteriza por (Craig, 1999):

- ❖ Organiza la estructura lógica de gran escala de un sistema en capas separadas de responsabilidades distintas y relacionadas, con una separación clara y cohesiva de intereses. Las capas "más bajas" son servicios generales de bajo nivel y las capas más altas son más específicas de la aplicación.
- ❖ La colaboración y el acoplamiento se realiza desde las capas más altas hacia las más bajas.

Específicamente el componente propuesto utiliza tres capas.

- ❖ Capa Comunicación: Contiene la clase *cController* que es la encargada de la comunicación entre el componente y el Subsistema de monitorización.
- ❖ Capa Lógica del negocio: contiene las clases asociadas a las funcionalidades del componente. Algunas de estas clases son *cTask*, *cThread* y *cVideoProcessor*.
- ❖ Capa Acceso a datos: Contiene la clase *cConexion* que se encarga de la conexión con la base de datos.

2.6.4 *Patrones de Diseño*

Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí para resolver un problema de diseño general en un contexto particular (Craig, 1999).

Patrones GRASP⁵

A continuación se describen los patrones GRASP utilizados en el desarrollo del componente:

- ❖ **Experto**: su objetivo es asignar responsabilidades específicas a las clases que poseen la información necesaria para cumplirla. Cada clase contiene los métodos relacionados con la información que posee (Craig, 1999).
- ❖ **Creador**: plantea la necesidad de asignarle a una clase la responsabilidad de crear una instancia de otra clase siempre y cuando agregue los objetos de la clase, los contenga, registre las instancias de estos objetos y los utilice específicamente (Craig, 1999).

⁵ Patrones para la asignación de responsabilidades.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

- ❖ **Alta cohesión:** este patrón es una medida de cuán relacionadas y orientadas están las responsabilidades de una clase. Planteando la contribución entre clases para realizar tareas de elevada complejidad (Craig, 1999).
- ❖ **Bajo acoplamiento:** propone tener las clases del diseño lo menos ligadas posible, con el fin de reducir el impacto de los cambios y permitir una mayor reutilización del código (Craig, 1999).
- ❖ **Controlador:** define quién deberá encargarse de atender un evento del sistema. Es un objeto en particular, en una aplicación, que lleva el manejo de la misma (Craig, 1999).

Patrones GoF

GOF (*Gang of Four*) representa un grupo de personas que se dedicaron a analizar los problemas recurrentes en el desarrollo de software y realizaron una clasificación y agrupación a partir de dos criterios, su propósito y alcance. Según este grupo, los patrones “describen soluciones simples y elegantes a problemas específicos en el diseño de software orientado a objetos” (Guerrero, et al., 2013).

A continuación se describe el patrón GoF presente en la solución propuesta:

- ❖ **Observador:** permite que si algún objeto observado cambia de estado, los objetos que lo observan sean notificados automáticamente. En el componente propuesto el patrón se manifiesta a través del mecanismo de *señales y slots*⁶ del *framework* QT, que se utiliza durante la implementación del componente.

2.6.5 Estándar de Codificación

Las convenciones o estándares de codificación son reglas para escribir el código fuente de una manera clara y legible, facilitando su comprensión y apariencia. En la codificación del componente se utilizó la notación Camello (Fig 8), específicamente el estilo “minúscula”, para nombrar los atributos, los métodos y las clases (Miranda, 2010). En este tipo específico de notación Camello, la primera palabra se inicia con minúscula.

⁶ Es el encargado de generar todo el código necesario para que en tiempo de ejecución se enlacen y sincronicen las señales emitidas con sus respectivos slots receptores.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

```

void cAlgorithm::getVariation(cv::Mat orig, cv::Mat dist,cv::Mat vVer)
{
    for (int i = 1; i < orig.rows-1; ++i)
    {
        for (int j = 1; j < orig.cols-1; ++j)
        {
            if((orig.at<int>(i,j)-dist.at<int>(i,j))>0)
                vVer.at<int>(i,j)=orig.at<int>(i,j)-dist.at<int>(i,j);
            else
                vVer.at<int>(i,j)=0;
        }
    }
}

```

Figura 8. Ejemplo del estilo de código utilizado en el desarrollo del componente propuesto.

2.6.6 Diagrama de clases de diseño

Los diagramas de clases de diseño constituyen un elemento esencial en la concepción de la aplicación que se propone, ya que sirven de guía a los desarrolladores para la construcción de una aproximación del sistema que se desea implementar, contribuyendo así a una mayor calidad del producto final. Muestran el sistema en términos de clases y métodos. Además reflejan el funcionamiento de la aplicación en términos lógicos (Vázquez Sosa, 2012).

A continuación se muestra una representación del diagrama de clases de diseño del componente (Fig 9). El diagrama con todos los detalles puede encontrarse en el Anexo 2.

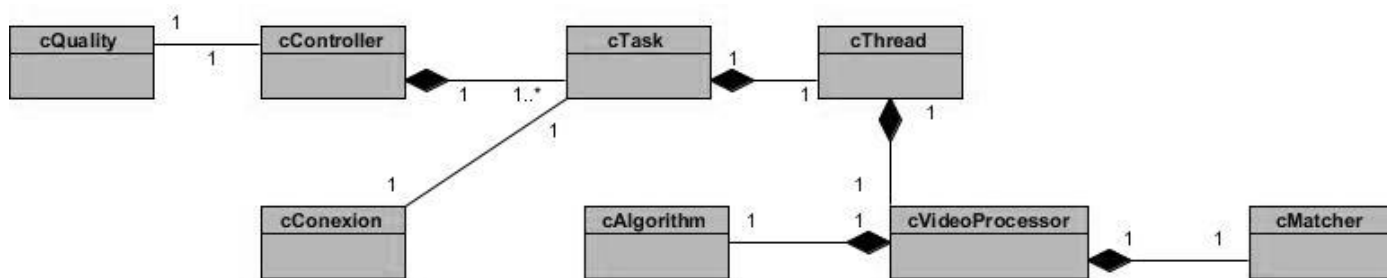


Figura 9. Diagrama de clases de diseño del componente.

2.6.7 Diagrama de secuencia

Los Diagramas de secuencia muestran la forma en que un grupo de objetos se comunican (interactúan) entre sí a lo largo del tiempo. Constan de objetos, mensajes entre estos y su línea de vida (Gutierrez, 2011).

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Véase el Diagrama de secuencia del caso de uso “Evaluar calidad de video” en el Anexo 3.

2.7 Funciones de OpenCV utilizadas

Se utilizó la biblioteca OpenCV para aplicar algoritmos relacionados con el procesamiento de imágenes. Esta biblioteca constituye un estándar de-facto de la industria. Su gran ventaja es que encapsula fácilmente algoritmos muy complejos, lo cual disminuye el esfuerzo requerido durante el desarrollo.

A continuación se describen brevemente algunas funciones de OpenCV utilizadas durante el desarrollo del componente:

- ❖ **cvtColor**: convierte una imagen del formato de color a otro.
- ❖ **Canny**: detecta bordes en una imagen.
- ❖ **SURF**: extrae características invariantes de una imagen.

2.8 Conclusiones parciales

En este capítulo se concluye que:

- ❖ La descripción de los algoritmos propuestos para detectar “Borrosidad” y “Efecto de bloque” constituye el basamento teórico para implementar el componente abordado en esta investigación.
- ❖ La descripción del caso de uso del sistema y el Diagrama de clases de diseño sirven de guía para la implementación del componente.
- ❖ La elección de la arquitectura N-capas, la aplicación de patrones de diseño y la utilización de un estándar de codificación, evidencian la aplicación de buenas prácticas durante el desarrollo del componente.

Capítulo 3: Implementación y pruebas

En este capítulo se abordan la implementación y las pruebas al componente propuesto en la investigación. Como parte de ello se muestra el Diagrama de componentes y el Diagrama de despliegue. Además se muestran los resultados obtenidos al probar los algoritmos propuestos, se describen las pruebas unitarias y las pruebas de integración.

3.1 Diagrama de componentes

El Modelo de implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Un componente es el empaquetamiento físico de los elementos de un modelo como lo son las clases del diseño (Jacobson, et al., 1999). Es por ello que se utiliza este diagrama UML para mostrar de forma sencilla la relación entre cada una de las capas y sus componentes.

A continuación se muestra el Diagrama de componentes de la solución propuesta:

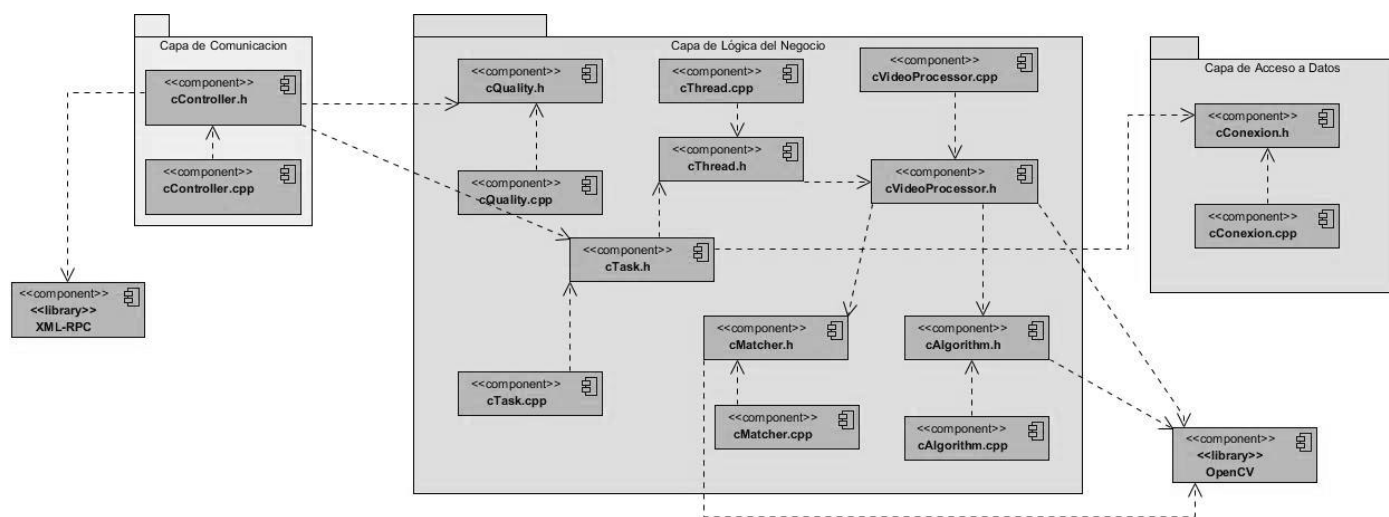


Figura 10. Diagrama de componentes de la solución.

3.2 Diagrama de despliegue

El Diagrama de despliegue permite mostrar la arquitectura en tiempo de ejecución del sistema respecto al hardware y software (Platero Dueñas, 2009). El Diagrama de despliegue del componente propuesto posee tres nodos:

- ❖ **Subsistema de monitorización:** representa el módulo al cual se integrará el componente desarrollado.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

- ❖ **Subsistema de transmisión:** representa el módulo que se encarga de la transmisión de los videos digitales.
- ❖ **Servidor de base de datos:** almacena la base de datos del STCV, en la cual se guarda el índice de los fotogramas claves.



Figura 11. Diagrama de despliegue de la solución.

3.3 Pruebas

Pressman categoriza las pruebas como: “un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones, el diseño y de la codificación” (Pressman, 2010). Teniendo en cuenta esta clasificación se denota la relevante importancia que posee esta etapa dentro de la construcción de productos de software. Durante esta etapa se reconocen los errores que se cometen durante el desarrollo y permite que se corrijan, evitando que el cliente final reciba un sistema defectuoso. Realizar las pruebas adecuadas puede aportar al sistema e incluso a la entidad desarrolladora un nivel de confiabilidad mayor ante los usuarios para los que trabaja.

El componente constituye una parte del SMCV ya implementado y en explotación. Debido a que el STCV está configurado y probado en cuanto a rendimiento y seguridad se decidió realizar pruebas unitarias basadas en la técnica del camino básico, pruebas de eficacia para validar el componente y pruebas de integración.

3.3.1 Pruebas unitarias

Las pruebas unitarias se describen como “un procedimiento usado para probar que un módulo o método funciona apropiadamente y en forma independiente. Permiten, además, detectar efectivamente la inyección de defectos durante fases sucesivas de desarrollo o mantenimiento (Pressman, 2010).

Estas pruebas se pueden realizar aplicando distintas técnicas dentro del método de Caja Blanca, que pueden ser ejecutadas de forma manual o mediante un programa automatizado. Al componente en cuestión se le aplica la técnica del camino básico y se hace de forma manual. Como parte de ello, se genera un grafo y se le calcula la complejidad ciclomática.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Los pasos para la realización de esta técnica son los siguientes:

- ❖ **Paso 1.** Generar el grafo de flujo.
- ❖ **Paso 2.** Calcular la complejidad ciclomática (CC), la cual es una métrica que indica la complejidad del problema. $CC = A$ (Número de aristas) – N (Número de vértices) + 2
- ❖ **Paso 3.** Definir los caminos básicos a partir de la complejidad ciclomática.
- ❖ **Paso 4.** Generar un Caso de prueba para cada camino básico.

Tabla 2. Valores para analizar la Complejidad ciclomática (Martínez Salazar, 2012).

Complejidad ciclomática	Evaluación del riesgo
1 – 10	Programa simple, sin mucho riesgo
11 – 20	Más complejo, riesgo moderado
21 – 50	Complejo, programa de alto riesgo
50 en adelante	Programa de muy alto riesgo

En el caso de la presente investigación se seleccionó un método según su complejidad e importancia dentro del componente, para aplicarle la prueba de “Caja blanca”. Este método es “*blockMetric*”, ubicado en la clase *cAlgorithm*. Dicho método es el encargado de calcular el “Efecto de bloque”.

A continuación se describe el proceso de pruebas realizado al Método “*blockMetric*”:

Paso 1: Generar grafo de flujo.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

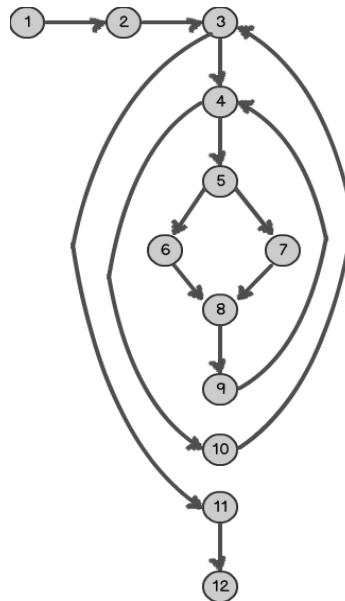


Figura 12. Grafo de flujo del método "blockMetric".

Paso 2: Calcular la complejidad ciclomática.

$$\begin{aligned}
 CC &= A - V + 2 \\
 &= 14 - 12 + 2 \\
 &= 4
 \end{aligned}$$

Siendo de 1 a 10 el primer rango de evaluación de la complejidad ciclomática, se obtiene que el programa es sin mucho riesgo.

Paso 3: Determinar los caminos básicos.

CB1: 1-2-3-11-12

CB2: 1-2-3-4-10-3-11-12

CB3: 1-2-3-4-5-6-8-9-4-10-3-11-12

CB4: 1-2-3-4-5-7-9-4-10-3-11-12

Paso 4: Generar casos de pruebas para los caminos básicos identificados.

Caso de prueba: CB1

Entrada: `for (int i = 0; i < o1.rows; ++i)`

Resultado de la prueba: Satisfactorio.

Caso de prueba: CB2

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Entrada: `for (int j = 0; j < o1.cols; ++j)`

Resultado de la prueba: Satisfactorio.

Caso de prueba: CB3

Entrada: `if (abs(o1.at<int>(i,j))==0&&abs(d1.at<int>(i,j))>0)`

Resultado de la prueba: Satisfactorio.

Caso de prueba: CB4

Entrada: `else if (abs(o1.at<int>(i,j))>0&&abs(d1.at<int>(i,j))==0)`

Resultado de la prueba: Satisfactorio.

3.3.2 Pruebas de integración

Las pruebas de integración se utilizan para verificar que los componentes integrados a una solución funcionan correctamente y como un todo (Pressman, 2010).

Integración descendente

La prueba de integración descendente es un planteamiento incremental a la construcción de la estructura de programas. Se integran los módulos moviéndose hacia abajo por la jerarquía de control (Pressman, 2010).

En el caso de esta investigación, se aplica este tipo de prueba comenzando por el “Subsistema de monitorización” y terminando en el componente propuesto.

En la fase de prueba se realizaron dos iteraciones al componente. En la primera iteración se encontró una no conformidad, la cual consistía en que no se detenía correctamente la evaluación de la calidad. Luego se realizó una segunda iteración, la cual arrojó que dicho problema había sido erradicado.

3.3.3 Pruebas Precision y Recall

Los indicadores “*Precision*” y “*Recall*” se utilizan para medir la efectividad de un algoritmo de clasificación.

Estos son definidos por las siguientes ecuaciones (Makhoul, et al., 2007):

$$Precision\ rate = \frac{DC}{DC+Fp} \quad Recall\ rate = \frac{DC}{DC+Fn} \quad (\text{ecuación 7})$$

Donde:

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

- ❖ **DC**: cantidad de detecciones correctas,
- ❖ **Fp**: cantidad de falsos positivos.
- ❖ **Fn**: cantidad de falsos negativos.

Los valores de Precision y Recall se combinan en una métrica F (ecuación 8), la cual indica el rendimiento del componente. Mientras mayor sea su valor, mejor se considera el rendimiento del componente.

$$F = 2 * \frac{Precision\ rate * Recall\ rate}{Precision\ rate + Recall\ rate} \quad (\text{ecuación 8})$$

De la ecuación anterior se deriva un indicador de error, el cual se define en la ecuación 9:

$$E = 1 - F. \quad (\text{ecuación 9})$$

Se seleccionaron de forma empírica 700 imágenes de acuerdo a una base de datos de prueba del proyecto STCV, cuyas características se muestran en las tablas 3 y 4.

Tabla 3. Total de imágenes clasificadas por tipo de distorsión.

Clasificación	Total
Sin Distorsiones	100
Poco "Efecto de bloque"	130
Mucho "Efecto de bloque"	170
Poca "Borrosidad"	100
Mucha "Borrosidad"	200
TOTAL	700

Tabla 4. Total de imágenes de acuerdo a las características del ambiente.

Ambiente	Total
De día	323

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

De noche	377
TOTAL	700
En exteriores	120
En interiores	580
TOTAL	700

Los algoritmos propuestos para detectar “Borrosidad” y “Efecto de Bloque” retornan un resultado en el rango [0,1], donde 0 significa ausencia total de la distorsión y 1 significa presencia total de la distorsión. Sin embargo en la implementación del componente, con el objetivo de facilitar el uso del mismo en el STCV, se adoptó una estrategia basada en umbral. Si el valor de la “Borrosidad” o el “Efecto de bloque” retornado por los algoritmos propuestos, es menor que el umbral correspondiente, se plantea que la distorsión no está presente; en caso contrario se plantea que sí está presente.

A continuación se describe el proceso que se utilizó para seleccionar los umbrales anteriormente descritos, en la implementación del componente. Debe decirse que este proceso estuvo basado en un subconjunto de 400 imágenes, tomadas a partir de las 700 imágenes que conforman la base de datos de prueba. Dichas imágenes fueron tomadas según el tipo de distorsión, de forma aleatoria.

Se utilizaron 3 umbrales para decidir si hay “Borrosidad” o no. Estos umbrales son 0.15, 0.30 y 0.70. Por cada umbral se calcularon las métricas. Como se puede apreciar en la tabla 5 el mejor resultado se obtuvo con el umbral 0.15. Por tanto este es el que se emplea en el componente.

Tabla 5. Resultados obtenidos al evaluar la “Borrosidad” utilizando distintos umbrales.

Umbrales	DC	Fp	Fn	Precision	Recall	F	E
0.70	116	13	171	0.89	0,40	0,54	0,46
0.30	311	7	82	0.97	0.79	0.86	0.14
0.15	393	7	0	1	0.98	0.98	0.02

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Se utilizaron 2 umbrales para decidir si hay “Efecto de bloque” o no. Estos umbrales son 0.30 y 0.70. Por cada umbral se calcularon las métricas. Como se puede apreciar en la tabla 6 el mejor resultado se obtuvo con el umbral 0.30. Por tanto este es el que se emplea en el componente.

Tabla 6. Resultados obtenidos al evaluar la “Efecto de bloque” utilizando distintos umbrales.

Umbrales	DC	Fp	Fn	Precision	Recall	F	E
0.70	283	5	112	0.98	0.71	0.80	0.20
0.30	395	1	4	0.99	0.98	0,98	0.02

Una vez identificados los dos umbrales mencionados anteriormente, se probó la efectividad del componente probándolo en todas las imágenes de la base de datos de pruebas, cuyo tamaño es de 700 imágenes. En la tabla 7 se muestran los resultados obtenidos.

Tabla 7. Resultados globales del componente propuesto en la investigación.

Parámetros	Valores
DC	686
Fp	10
Fn	4
Precision	0.98
Recall	0.99
F	0.98
E	0.02

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

3.4 Conclusiones parciales

En este capítulo se concluye que:

- ❖ Los resultados obtenidos al probar los algoritmos para evaluar la “Borrosidad” y el “Efecto de bloque” son satisfactorios, por lo cual el componente puede ser utilizado en un ambiente real para monitorizar la calidad de video durante la transmisión.
- ❖ Las pruebas aplicadas al componente demostraron que el mismo se integra satisfactoriamente con el Subsistema de monitorización y cumple con los requisitos trazados.

Conclusiones generales

Con la confección del trabajo de diploma “Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa” se dio cumplimiento al objetivo general planteado al comienzo de la investigación. Para la solución del problema científico se definieron los aspectos teóricos, prácticos y se desarrollaron en su totalidad las tareas de la investigación que contribuyeron a la solución final. Se puede concluir que:

- La utilización de RUP como metodología de desarrollo generó artefactos que representan una garantía para la continuidad del trabajo en el proyecto STCV.
- El lenguaje utilizado, las herramientas y bibliotecas incluidas ofrecieron el soporte necesario para lograr un componente con los requerimientos y exigencias planteadas por el cliente.
- El componente resultante fue estructurado siguiendo una arquitectura en capas, en su variante 3 capas.
- Se definió un flujo de procesamiento para la evaluación de la calidad de video digital en transmisiones utilizando la técnica Referencia Completa, el cual garantiza la monitorización de la calidad de la transmisión, disminuyendo el esfuerzo humano.
- Se identificaron umbrales sobre características de una imagen que fueron validados con la realización de pruebas sobre una base de datos de imágenes. Los mismos demostraron la efectividad de la investigación, con la cual se sentaron las bases para futuros trabajos.
- La solución presentada automatiza y refuerza el proceso de evaluación de la calidad en el Subsistema Monitorización de Canales Virtuales.
- Se logró obtener un producto final de calidad, avalado por las pruebas de software realizadas al componente.

De esta manera se puede concluir que se logró definir un flujo de procesamiento para la evaluación de la calidad de video, implementado como un componente que se integró al subsistema Monitorización luego de ser validado y probado para garantizar la efectividad del mismo.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Recomendaciones

- ❖ Implementar otros algoritmos para la detección de distorsiones como “Efecto de imagen de base”, “Efecto escalera” y “Patrones de mosaico”, para aumentar la eficiencia de los resultados en la evaluación de calidad.
- ❖ Disminuir el tiempo de respuesta del proceso de sincronización.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Referencias bibliográficas

- Fu, Pan Ming , Lin, Xiaola y Ong, Ee Ping. 2004.** *Measuring Blocking Artifacts Using Edge Direction Information*, *IEEE International Conference on Multimedia and Expo (ICME)*. Singapore : s.n., 2004.
- Ladret, Patricia, y otros. 2008.** *The Blur Effect: Perception and Estimation with a New No-Reference Perceptual Blur Metric*. Francia : s.n., 2008. pp.El 6492-16.
- Agustí , Manuel , Benlloch, Jose V. y Atienza, Vicente . 2010.** *Adquisición y representación de medios digitales*. 2010.
- Baz Hormigos, Carlos Esteban. 2009.** *Medidas de calidad subjetiva*. 2009.
- Blasco, Irene Hernández. 2008.** *Algoritmos de detección de imágenes ruidosas y duplicadas*. 2008.
- C. Gonzalez, Rafael y E. Woods, Richard. 2007.** *Digital Image Processing (3rd Edition)*. 2007.
- Camacho, Erika, Nuñez, Gabriel y Cardeso, Fabio. 2004.** *Arquitecturas de software*. 2004.
- Carvajal Álvarez , Leonardo Andrés . 2009.** *Sistemas. Concepto sobre infraestructura tecnológica de un Canal de Televisión y su aplicación al diseño*. Cali. : Unilibre., 2009. Vol. Vol. 5, No. 1. 9: 76 -106.
- Casado Alvarado, Marco . 2000.** *La creación de video experimental como material didáctico en la licenciatura de Comunicación*. México : Universidad Autónoma del Estado de México Facultad de Ciencias Políticas y Administración Pública., 2000.
- Corporation, Nokia. 2011.** *QT*. 2011.
- Craig, Lerman. 1999.** *UML y Patrones*. Montevideo. Uruguay : s.n., 1999.
- Furfaro, Alejandro. 2010.** Manejo de Bibliotecas Opencv. [En línea] 2010. <http://www.electron.frba.utn.edu.ar/~afurfaro/Info1/Opencv/opencv.pdf>.
- Gandam, Anudeep y Singh Sidhu, Jagroop . 2010.** *A Post-Processing Algorithm for Detection & Removal of Corner Outlier*. s.l. : International Journal of Computer Applications, 2010. 0975 – 8887.
- González, Carlos Daniel. 2008.** Introducción a C++ y a la Resolución de Problemas. [En línea] 2008. <http://www.usabilidadweb.com.ar/cpp.php>.
- Guerrero, Carlos A., Suárez, Johanna M. y Gutiérrez, Luz E. 2013.** No. 3, Colombia : La Serena, 2013, Vol. Vol. 24. ISSN 0718-0764..
- Gutierrez, Demián. 2011.** *UML.Diagrama de Secuencia*. Venezuela. : Universidad de los Andes., 2011.
- Hormigos Baz, Carlos Esteban. 2009.** *Archivo Abierto Institucional de la Universidad Carlos III de Madrid*. [En línea] 2009. http://e-archivo.uc3m.es/bitstream/10016/7202/2/PFC_Carlos_Esteban_Baz_Hormigos.pdf.
- ISO. 2015.** iso.org. [En línea] 2015.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Jacobson, Ivar , Booch, Grady y Rumbaugh, James. 2000. *El Proceso Unificado de Desarrollo de Software. Madrid.* Madrid : Adisson Wesley., 2000.

Jacobson, Ivar, Rumbaugh, James y Booch, Grady. 1999. Caps. 16 y 17. *El lenguaje unificado de modelado.* Madrid : s.n., 1999.

Joskowicz, Ing. José . 2008. *Estudio de la Medida de la Calidad Perceptual de Video.* 2008.

Joskowicz, José y Sotelo, Rafael. 2012. No. 10, Montevideo, Uruguay. : Universidad de la República., 2012, Vol. Memoria de Trabajos de Difusión Científica y Técnica. ISSN 1510-7450.

Kerouh, F. y Serir, A. . 2011. *A No Reference Quality Metric for Measuring Image Blur In Wavelet Domain.* s.l. : The Society of Digital Information and Wireless Communications, 2011. ISSN 2225-658X.

Laboratorio Nacional de Calidad del Software. 2009. *Ingeniería de Software. Metodologías y ciclos de vida.* España : INTECO, 2009.

Makhoul, John, y otros. 2007. *Performance Measures for Information Extraction.* Cambridge : BBN Technologies, GTE Corp., 2007. MA 02138.

Marichal, Xavier, Ma, Wei-Ying y Zhang, HongJiang. 1998. *Blur determination in the compressed domain using DCT.* 1998.

Martínez Salazar, Ing. Eduardo. 2012. *Propuesta de procedimiento para realizar pruebas de Caja Blanca a la aplicaciones que se desarrollan en el lenguaje Python.* Cuba : Facultad regional de Granma, 2012.

Marziliano, Pina , y otros. 2010. *A no-refrence perceptual blur metric.* 2010.

Menéndez Verde, Magalys Yisel y Aleman Martínez, Ana Lizandra . 2011. *Evaluación de la calidad de los videos en la plataforma PTARTV basado en métricas de detección de errores.* La Habana, Cuba : Universidad de las Ciencias Informáticas., 2011.

Miranda, Marco. 2010. *Documento de Estándares de Programación.* 2010.

Muijs, R y Kirenko, I. 2005. *A no-reference blocking artefacts .* 2005.

Pachacama, Adrián y Carvajal Catagña, Edwin Aníbal. 2010. *Implementación de un Sistema.* 2010.

Pérez Míñquez, Eduardo y Enríquez Garcinuño, David . 2005. *Departamento de Informática y Automática.* Salamanca : QT, 2005.

Platero Dueñas, Carlos. 2009. *UML Dinámico.* s.l. : Dpto. Electrónica, Automática e Informática Industrial , 2009.

Pressman, Roger. 2010. *Ingeniería de Software. Un enfoque práctico.* 2010.

Puerto Acevedo, Jesús Andrés. 2014. *Informática Educativa. El video.* 2014. 0701067.

Roddick, John F, y otros. 2014. *No-reference Quality Metric of Blocking Artifacts.* 2014.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Salgado Magaña, Pedro Diosmel y Díaz Saez, Yelen . 2011. *Evaluación de la calidad de los videos en la plataforma PTARTV basado en métricas de distorsión estructural.* La Habana, Cuba : Universidad de las Ciencias Informáticas, 2011.

Santos, Jorge de Jesus, Ian, Yuzo y Sampaio Gradvoh, Andre León. 2011. *A Novel Blocking Effect Reduction Method Based on Human Vision System Features.* s.l. : Revista Ciencia y Tecnología, 2011. ISSN:1677-9649.

Slonneger, Ken. 2006. *XML-RPC.* 2006.

Software., Ingeniería de. 2013. Ingeniería de Software. [En línea] 2013. <http://www.ie.inf.uc3m.es/grupo/docencia/reglada/1s1y2/PracticaVP.pdf>).

Tappert, Charles c., Choi, Seung-Seok y Cha, Sung-Hyunk. 2010. *A survey of Similarity and Distance Measures.* Estados Unidos : Departament of Computer Science, Pace University, 2010. ISSN: 1690-4524.

Vázquez Sosa, Sisley. 2012. *Módulo web de catalogación de materiales audiovisuales del Sistema de captura y catalogación de medias.* La Habana : Unversidad de las Ciencias infomáticas., 2012.

Vega González, Yosami y Valdés Ledesma, Yasmani . 2013. *Sistema para la gestión de la información en el estudio neuroinmunológico de proteínas del líquido cefalorraquídeo.* La Habana : Universidad de las Ciencias Informáticas., 2013.

Virtual, Impacto. 2010. Impacto Virtual. [En línea] 2010. <http://www.impactovirtual.com/VirtualPowerVideo/>). 572.

Wilson, Leslie B. 1993. No. 75, s.l. : Addison-Wesley, 1993, Vol. 2da Edición. ISBN 0-201-56885-3.

Xingang, Liu y Yeol-Kook, Yoo. 2008. *Real-Time Reference-Free Video Quality Measurement for Multimedia Communication.* 2008.

Yanhui, Xia, y otros. 2013. *Blind Measurement of Blocking Artifacts of.* 2013. ISSN:1677-9649.

Yerovi , Ximena y Poz, Ing. Alejandra. 2013. *Ventajas de C++.* . Ecuador : s.n., 2013.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Bibliografía consultada

- Fu, Pan Ming , Lin, Xiaola y Ong, Ee Ping. 2004.** *Measuring Blocking Artifacts Using Edge Direction Information, IEEE International Conference on Multimedia and Expo (ICME).* Singapore : s.n., 2004.
- Ladret, Patricia, y otros. 2008.** *The Blur Effect: Perception and Estimation with a New No-Reference Perceptual Blur Metric.* Francia : s.n., 2008. pp.El 6492-16.
- Agustí , Manuel , Benlloch, Jose V. y Atienza, Vicente . 2010.** *Adquisición y representación de medios digitales.* 2010.
- Baz Hormigos, Carlos Esteban. 2009.** *Medidas de calidad subjetiva.* 2009.
- Blasco, Irene Hernández. 2008.** *Algoritmos de detección de imágenes ruidosas y duplicadas.* 2008.
- C. Gonzalez, Rafael y E. Woods, Richard. 2007.** *Digital Image Processing (3rd Edition).* 2007.
- Camacho, Erika, Nuñez, Gabriel y Cardeso, Fabio. 2004.** *Arquitecturas de software.* 2004.
- Carvajal Álvarez , Leonardo Andrés . 2009.** *Sistemas. Concepto sobre infraestructura tecnológica de un Canal de Televisión y su aplicación al diseño.* Cali. : Unilibre., 2009. Vol. Vol. 5, No. 1. 9: 76 -106.
- Casado Alvarado, Marco . 2000.** *La creación de video experimental como material didáctico en la licenciatura de Comunicación.* México : Universidad Autónoma del Estado de México Facultad de Ciencias Políticas y Administración Pública., 2000.
- Corporation, Nokia. 2011.** *QT.* 2011.
- Craig, Lerman. 1999.** *UML y Patrones.* Montevideo. Uruguay : s.n., 1999.
- Furfaro, Alejandro. 2010.** Manejo de Bibliotecas Opencv. [En línea] 2010. <http://www.electron.frba.utn.edu.ar/~afurfaro/Info1/Opencv/opencv.pdf>.
- Gandam, Anudeep y Singh Sidhu, Jagroop . 2010.** *A Post-Processing Algorithm for Detection & Removal of Corner Outlier.* s.l. : International Journal of Computer Applications, 2010. 0975 – 8887.
- González, Carlos Daniel. 2008.** Introducción a C++ y a la Resolución de Problemas. [En línea] 2008. <http://www.usabilidadweb.com.ar/cpp.php>.
- Guerrero, Carlos A., Suárez, Johanna M. y Gutiérrez, Luz E. 2013.** No. 3, Colombia : La Serena, 2013, Vol. Vol. 24. ISSN 0718-0764..
- Gutierrez, Demián. 2011.** *UML.Diagrama de Secuencia.* Venezuela. : Universidad de los Andes., 2011.
- Hormigos Baz, Carlos Esteban. 2009.** *Archivo Abierto Institucional de la Universidad Carlos III de Madrid.* [En línea] 2009. http://e-archivo.uc3m.es/bitstream/10016/7202/2/PFC_Carlos_Esteban_Baz_Hormigos.pdf.
- ISO. 2015.** iso.org. [En línea] 2015.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

- Jacobson, Ivar , Booch, Grady y Rumbaugh, James. 2000.** *El Proceso Unificado de Desarrollo de Software. Madrid.* Madrid : Adisson Wesley., 2000.
- Jacobson, Ivar, Rumbaugh, James y Booch, Grady. 1999.** Caps. 16 y 17. *El lenguaje unificado de modelado.* Madrid : s.n., 1999.
- Joskowicz, Ing. José . 2008.** *Estudio de la Medida de la Calidad Perceptual de Video.* 2008.
- Joskowicz, José y Sotelo, Rafael. 2012.** No. 10, Montevideo, Uruguay. : Universidad de la República., 2012, Vol. Memoria de Trabajos de Difusión Científica y Técnica. ISSN 1510-7450.
- Kerouh, F. y Serir, A. . 2011.** *A No Reference Quality Metric for Measuring Image Blur In Wavelet Domain.* s.l. : The Society of Digital Information and Wireless Communications, 2011. ISSN 2225-658X.
- Laboratorio Nacional de Calidad del Software. 2009.** *Ingeniería de Software. Metodologías y ciclos de vida.* España : INTECO, 2009.
- Makhoul, John, y otros. 2007.** *Performance Measures for Information Extraction.* Cambridge : BBN Technologies, GTE Corp., 2007. MA 02138.
- Marichal, Xavier, Ma, Wei-Ying y Zhang, HongJiang. 1998.** *Blur determination in the compressed domain using DCT.* 1998.
- Martínez Salazar, Ing. Eduardo. 2012.** *Propuesta de procedimiento para realizar pruebas de Caja Blanca a la aplicaciones que se desarrollan en el lenguaje Python.* Cuba : Facultad regional de Granma, 2012.
- Marziliano, Pina , y otros. 2010.** *A no-refrence perceptual blur metric.* 2010.
- Menéndez Verde, Magalys Yisel y Aleman Martínez, Ana Lizandra . 2011.** *Evaluación de la calidad de los videos en la plataforma PTARTV basado en métricas de detección de errores.* La Habana, Cuba : Universidad de las Ciencias Informáticas., 2011.
- Miranda, Marco. 2010.** *Documento de Estándares de Programación.* 2010.
- Muijs, R y Kirenko, I. 2005.** *A no-reference blocking artefacts .* 2005.
- Pachacama, Adrián y Carvajal Catagña, Edwin Aníbal. 2010.** *Implementación de un Sistema.* 2010.
- Pérez Míñquez, Eduardo y Enríquez Garcinuño, David . 2005.** *Departamento de Informática y Automática.* Salamanca : QT, 2005.
- Platero Dueñas, Carlos. 2009.** *UML Dinámico.* s.l. : Dpto. Electrónica, Automática e Informática Industrial , 2009.
- Pressman, Roger. 2010.** *Ingeniería de Software. Un enfoque práctico.* 2010.
- Puerto Acevedo, Jesús Andrés. 2014.** *Informática Educativa. El video.* 2014. 0701067.
- Roddick, John F, y otros. 2014.** *No-reference Quality Metric of Blocking Artifacts.* 2014.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Salgado Magaña, Pedro Diosmel y Díaz Saez, Yelen . 2011. *Evaluación de la calidad de los videos en la plataforma PTARTV basado en métricas de distorsión estructural.* La Habana, Cuba : Universidad de las Ciencias Informáticas, 2011.

Santos, Jorge de Jesus, Ian, Yuzo y Sampaio Gradvoh, Andre León. 2011. *A Novel Blocking Effect Reduction Method Based on Human Vision System Features.* s.l. : Revista Ciencia y Tecnología, 2011. ISSN:1677-9649.

Slonneger, Ken. 2006. *XML-RPC.* 2006.

Software., Ingeniería de. 2013. Ingeniería de Software. [En línea] 2013. <http://www.ie.inf.uc3m.es/grupo/docencia/reglada/ls1y2/PracticaVP.pdf>).

Tappert, Charles c., Choi, Seung-Seok y Cha, Sung-Hyunk. 2010. *A survey of Similarity and Distance Measures.* Estados Unidos : Departament of Computer Science, Pace University, 2010. ISSN: 1690-4524.

Vázquez Sosa, Sisley. 2012. *Módulo web de catalogación de materiales audiovisuales del Sistema de captura y catalogación de medias.* La Habana : Unversidad de las Ciencias infomáticas., 2012.

Vega González, Yosami y Valdés Ledesma, Yasmani . 2013. *Sistema para la gestión de la información en el estudio neuroinmunológico de proteínas del líquido cefalorraquídeo.* La Habana : Universidad de las Ciencias Informáticas., 2013.

Virtual, Impacto. 2010. Impacto Virtual. [En línea] 2010. <http://www.impactovirtual.com/VirtualPowerVideo/>). 572.

Wilson, Leslie B. 1993. No. 75, s.l. : Addison-Wesley, 1993, Vol. 2da Edición. ISBN 0-201-56885-3.

Xingang, Liu y Yeol-Kook, Yoo. 2008. *Real-Time Reference-Free Video Quality Measurement for Multimedia Communication.* 2008.

Yanhui, Xia, y otros. 2013. *Blind Measurement of Blocking Artifacts of.* 2013. ISSN:1677-9649.

Yerovi , Ximena y Poz, Ing. Alejandra. 2013. *Ventajas de C++.* . Ecuador : s.n., 2013.

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Anexos

Anexo #1 Entrevista

(Realizada a especialistas del proyecto Sistema de Transmisión de Canales Virtuales)

Objetivo: Conocer el sistema que se utiliza en el Subsistema de Monitorización para evaluar la calidad de videos digitales.

Compañero o compañera.

Se necesita su valiosa colaboración para la realización de esta investigación, de sus respuestas dependen los resultados de la misma. **MUCHAS GRACIAS.**

Datos generales.

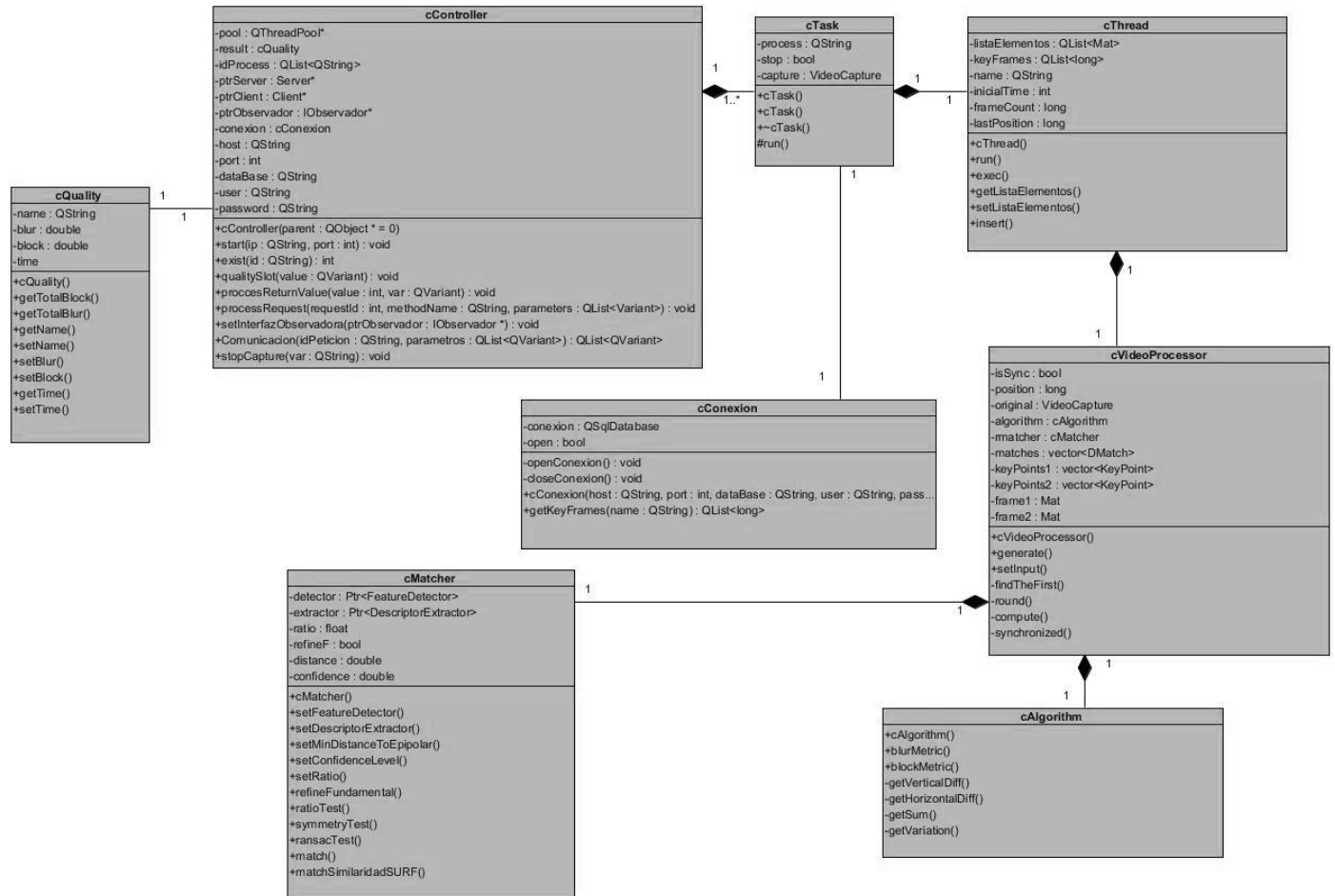
- Nombre y Apellidos _____
- Años de experiencia _____

Aspectos a encuestar

1. ¿Los videos digitales que se transmiten en el sistema provienen de una misma fuente de datos? De ser negativa la respuesta, explique los formatos de video que utilizan.
2. ¿Qué importancia le concede a que los videos digitales sean transmitidos con calidad?
3. ¿Qué método de evaluación se encuentra implementado en el Subsistema de Monitorización?
4. ¿Son efectivos los resultados que se obtienen con el método actual? ¿Por qué?
5. Explique brevemente qué ventajas considera que le brindará al Subsistema la creación de un nuevo método para evaluar la calidad de videos digitales.
6. ¿Conoce usted algunos métodos o técnicas que realicen la evaluación de la calidad?
7. ¿Cuáles de las alternativas propuestas sería más efectiva, teniendo en cuenta las características del subsistema?

Componente para evaluar la calidad de videos digitales en el Sistema de Transmisión de Canales Virtuales utilizando la técnica Referencia Completa

Anexo #2: Diagrama de clases del diseño



Anexo #3: Diagrama de secuencia

