HTC-2036CB Camera

HTC-2036CB Camera

Development Manual

Ver 2.1

2013. 06. 25

http://www.hontek.co.kr

http://cafe.naver.com/hontek

Jun 25, 2013

목 차

1. 개 요 4

2. HTC-2036CB Camera Specification 5

2.1 General feature 5

2.2 PC Requirements 5

2.3 Contents 6

2.4 Dimension and Description 6

2.5 Camera Interface 7

2.6 Trigger / Strobe Connector Port 7

2.7 Electrical Operating Condition 8

3. 개발요구사항 9

3.1 개발 요구사항 9

3.2 Development Architecture 9

4. Software Archtecture 10

4.1 Multi Camera connection and Access 10

4.2 Sequence of Initialization 11

4.3 Main Control Dialog 12

4.4 ISP Control Dialog 22

4.5 OpenCV Control Dialog 26

5. System File Description 35

5.1 MVC_Vision.ini..... 35

6. API Lists 36

6.1 API Access Flow 36

6.2 Camera Initialization, control, and Image acquisition 37

6.3 ISP(Image Signal Processing) property and access 38

목 차

7. API 설명 39
7.1 HVR_Init 39
7.2 HVR_Release 40
7.3 HVR_EmpiaispAutoGainControlEnable 41
7.4 HVR_EmpiaispGetAWBGainControlRange..... 42
7.5 HVR_EmpiaispSetAWBGainControl 43
7.6 HVR_EmpiaispGetExposureRange 44
7.7 HVR_EmpiaispSetExposureTime 45
7.8 IIC_Read16i 46
7.9 IIC_Write16i 47
7.10 SoftTrigerOnOff 48
7.11 TriggerOnOff 49
7.12 SoftTriger 50
7.13 Slider 51
7.14 HVR_OpenCV_Create 52
7.15 HVR_OpenCV_Release 53
7.16 HVR_OpenCV_Display 54
7.17 HVR_OpenCV_StartAVI 55
7.18 HVR_OpenCV_StopAVI 56
7.19 HVR_OpenCV_TM_targetFile 57
7.20 HVR_OpenCV_LoadObjectOpenCV 58
7.21 HVR_OpenCV_MatchingObjectOpenCV 59
7.22 HVR_OpenCV_LaserTracking 60
7.23 HVR_OpenCV_SelEdge 61
7.24 HVR_OpenCV_Labeling 62
7.25 MVC_OpenCV_Sharpening 63

1. 개요

본 문서는 HTC-2036CB 비젼카메라를 이용하기 위한 사용자의 이해를 돕기 위해 만든 문서로 하드웨어 구성, 소프트웨어 구성, 프로그램 개발에 대한 설명을 나열 하였습니다. 기타 추가적인 질문사항이 있으시면 아래와 같은 방법으로 문의 하시기 바랍니다.

혼 텍

주소 : 경기도 성남시 중원구 상대원동 190-1 SKⓝ테크노파크 테크동 810호

전화 : 031-776-3897,9

팩스 : 031-776-3898

홈페이지 : www.hontek.co.kr

2. HTC-2036CB Camera Specification

2.1 General feature

☞ 30만 화소(VGA) 비젼카메라

☞ Effective pixel	640(H) x 480(V)
☞ Frame rate	Max. 60fps@VGA
☞ PC Interface	USB 2.0 (480Mbps)
☞ Sensor video data format	8Bit Bayer
☞ Sensor command interface	표준 I2C protocol
☞ Compression	압축 안 함
☞ Image Processing	Host 프로그램 구현 또는 센서 ISP 컨트롤 가능
☞ Multi-Camera connect	동일 PC 최대 4대 연결 가능

2.2 PC Requirements

☞ CPU	Pentium4 2.0GHz이상 (2.8GHz 이상 추천)
☞ RAM	512MB 이상 (1GB이상 권장)
☞ USB 2.0 Host Controller	Intel Controller chip 권장
☞ Operating System	Windows 2000 SP4 이상 / WindowsXP SP2이상
	Windows Vista / Windows 7

2. HTC-2036CB Camera Specification

2.3 Contents

☞ HTC-2036CB Camera

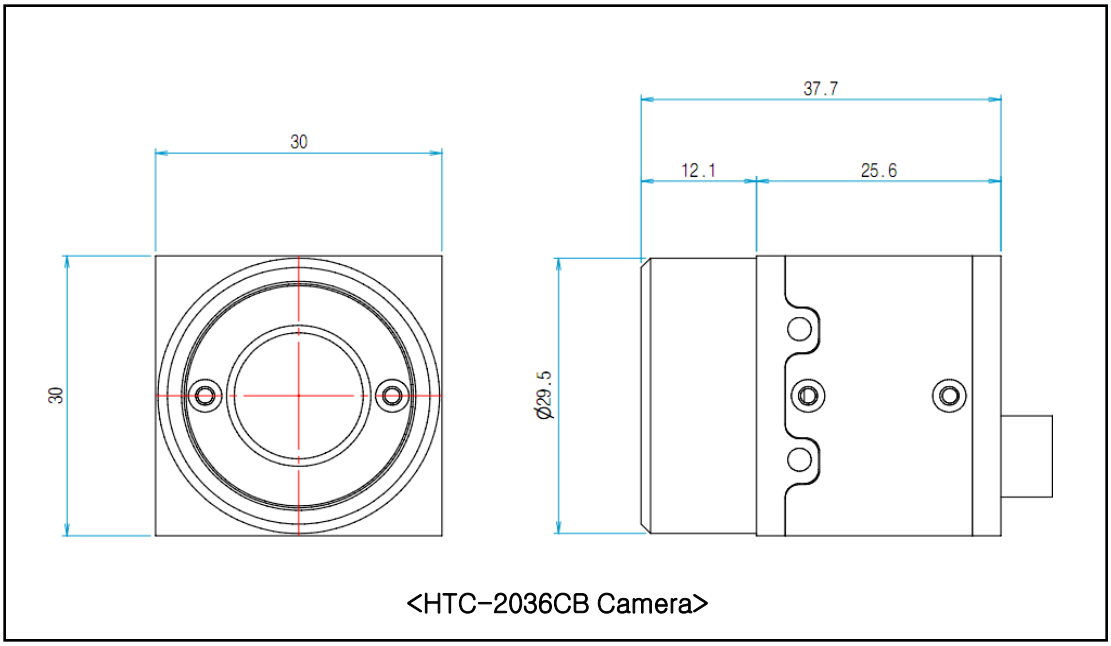
☞ Mini USB Cable(5Pin)

☞ User's manual / Program CD

※ Trig/Strobe 원형 Cable(4Pin) → (Option)

2.4 Dimension and Description

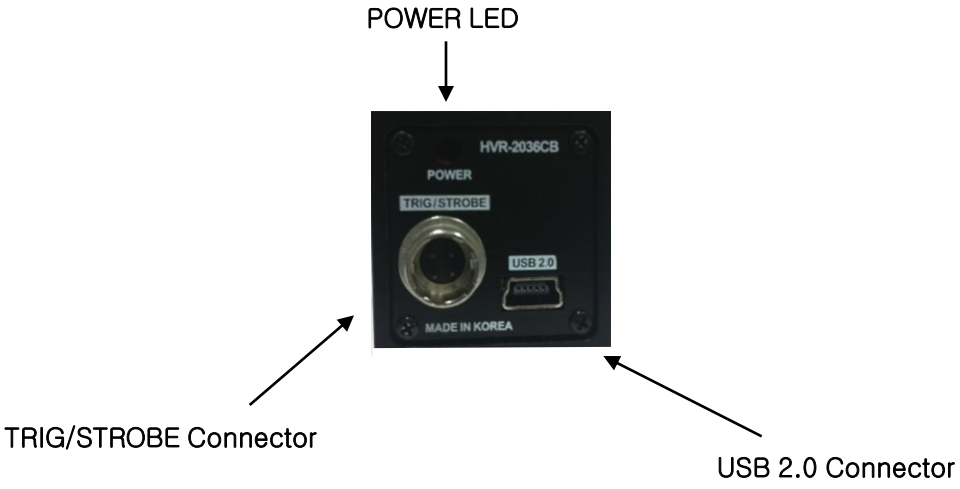
☞ Camera Body Size : 30(w) x 30(h) x 38(D) mm



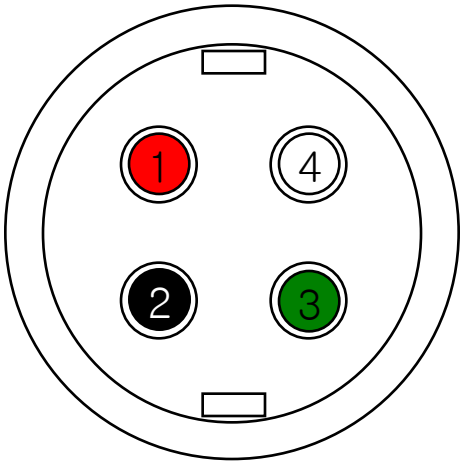
<HTC-2036CB Camera>

2. HTC-2036CB Camera Specification

2.5 Camera Interface



2.6 Trigger / Strobe Connector Port



<Camera Side>

Pin	Signal Name
1	Trigger +
2	Trigger -
3	Strobe +
4	Strobe -

* Wire Color

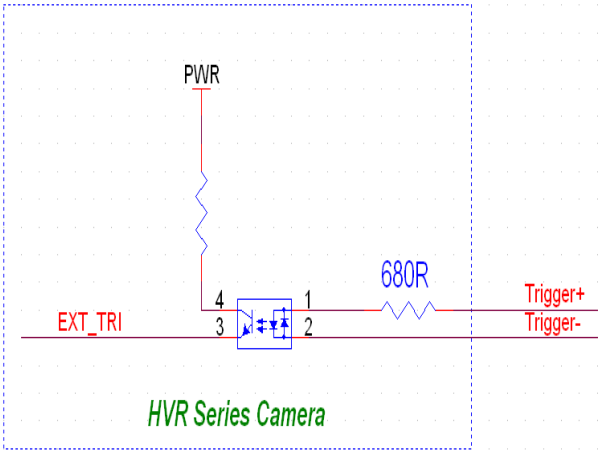
1 : Red 2 : Black

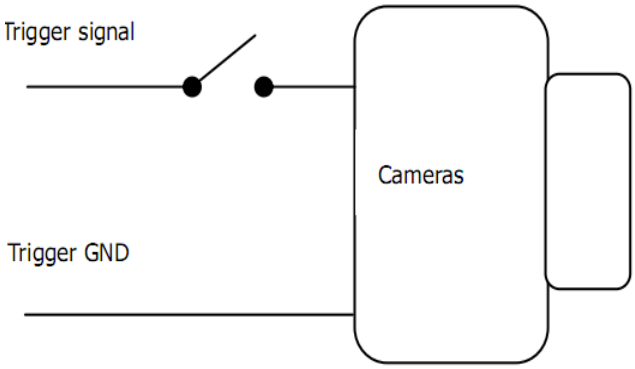
3 : Green 4 : White

2. HTC-2036CB Camera Specification

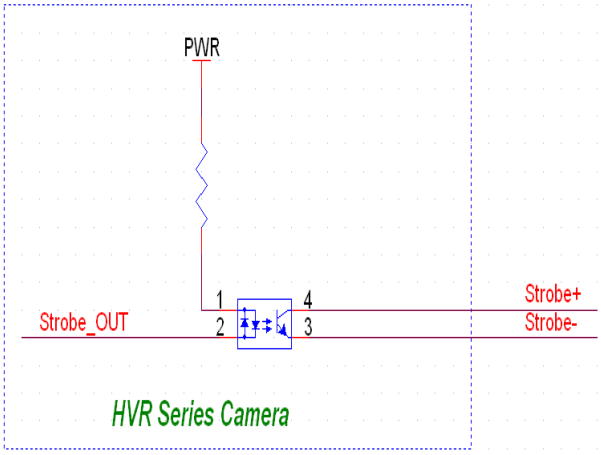
2.7 Electrical Operating Condition

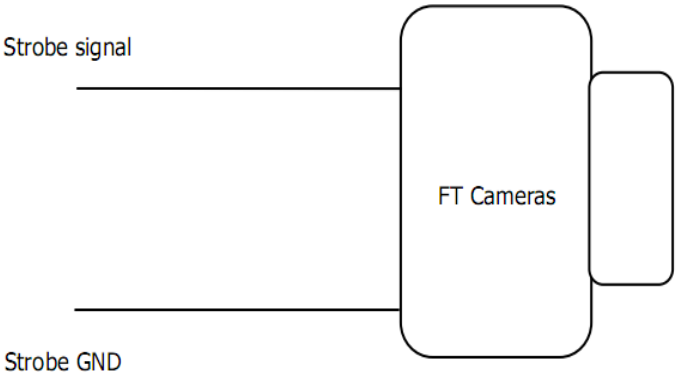
<Trigger>





<Strobe>





3. 개발 환경

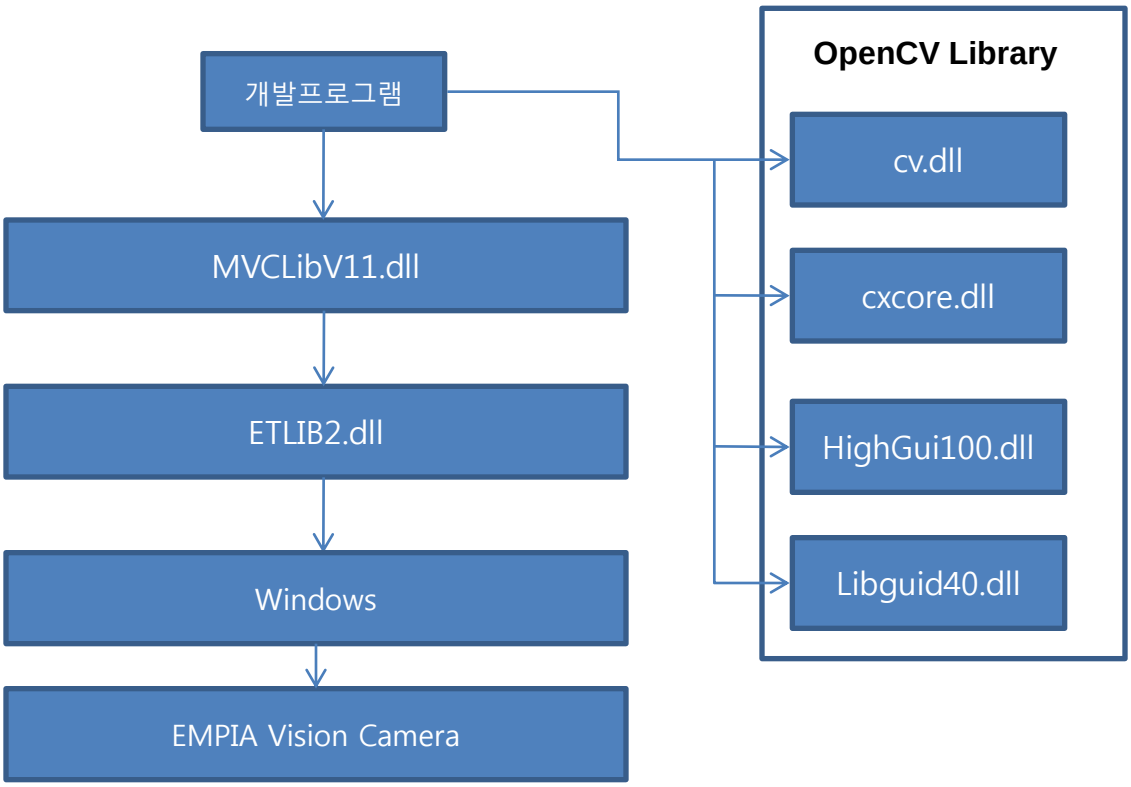
3.1 개발 요구사항

☞ 운영체제 : Microsoft Windows 2000 SP4 이상, WindowsXP SP2 이상,
Windows Vista, Windows 7

☞ 컴파일러 : Microsoft Visual Basic / Visual C++ 6.0
Visual Basic .NET / Visual C++ .NET

☞ 기타 : openCV 라이브러리

3.2 Development Architecture



4. Software Architecture

4.1 Multi Camera connection and Access

HTC-2036CB 카메라는 동일 PC상에 최대 4대의 카메라 연결이 가능합니다. 카메라에 시리얼 번호를 각각 부여하여(1001~1004) 각 카메라를 식별합니다.

Single Connection

Serial_Number[5]=L"1001";

M fps

Multi Connection

Serial_Number[5]=L"1001";

M/N fps

Serial_Number[5]=L"1002";

M/N fps

Serial_Number[5]=L"1003";

M/N fps

Serial_Number[5]=L"1004";

M/N fps

Fig 1. Multi Camera connection and Access

4. Software Architecture

4.2 Sequence of Initialization

실행파일 실행시, Device Check에 관한 초기화 과정을 설명합니다. 개발자 분은 아래 과정을 참조하시기 바랍니다.

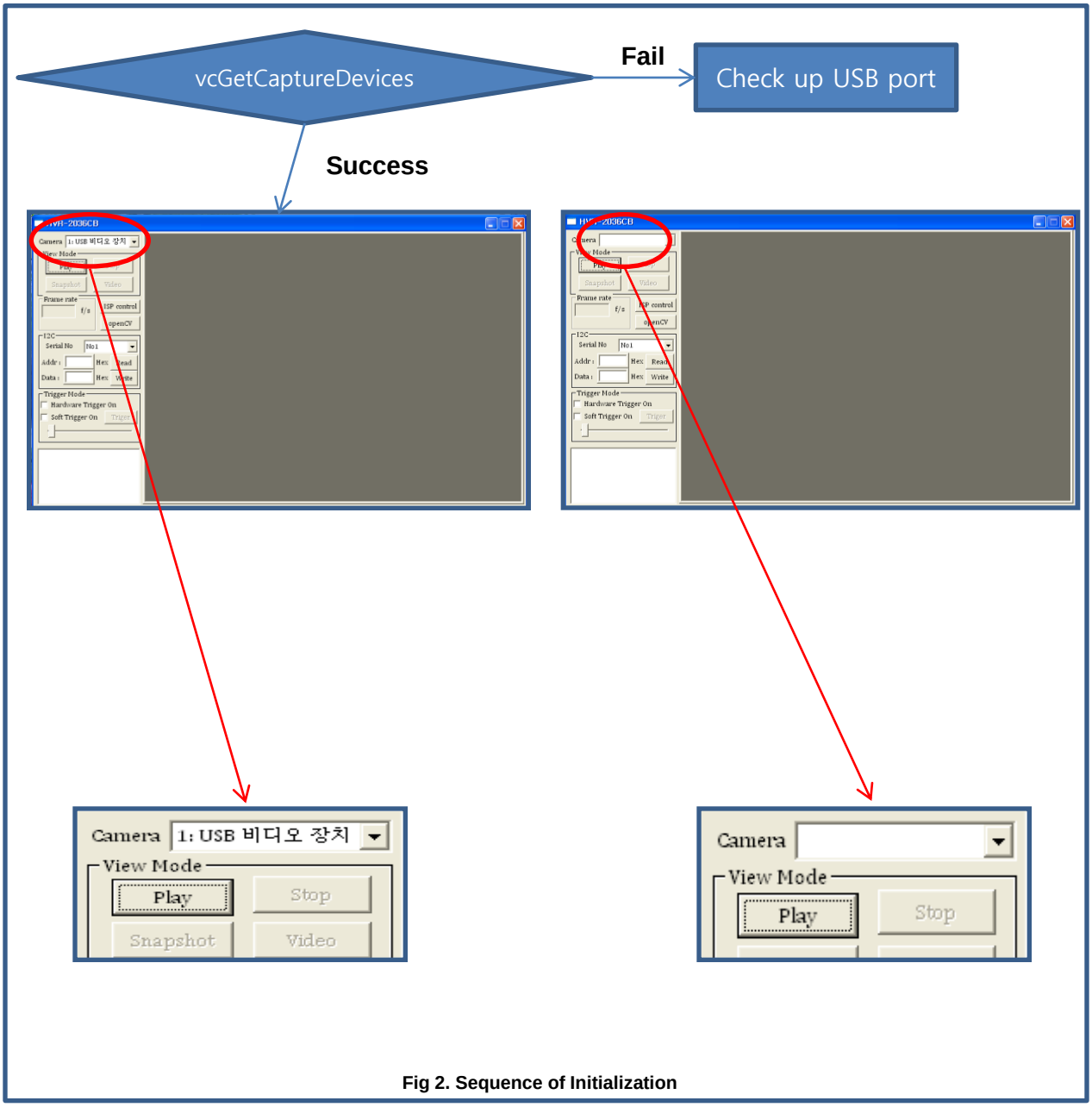


Fig 2. Sequence of Initialization

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.

4.3.1 카메라 검색

장비와 연결된 카메라를 검색하고 연결된 카메라를 Combo Box에 나열한다.

Fig 3. Main Control Dialog - 해상도

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.

4.3.2 View Mode

선택된 카메라를 실행시키거나 중지, 또는 정지영상, 동영상을 저장한다.

Play : 카메라를 실행

Stop : 카메라 실생을 중지

Snapshot : 카메라 실행 시에만 동작하며, 하나의 정지영상을 BMP파일로 저장한다.

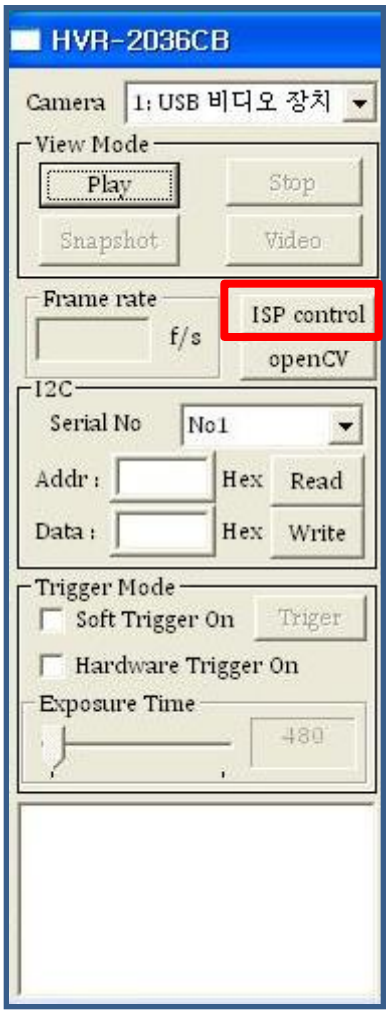
Video : 카메라 실행 시에만 동작하며, 동영상을 저장한다.

Fig 4. Main Control Dialog - 해상도

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.



4.3.3 ISP(Image Signal Processing) Control

카메라로부터 입력되는 영상을 가공하는 기능을 담당

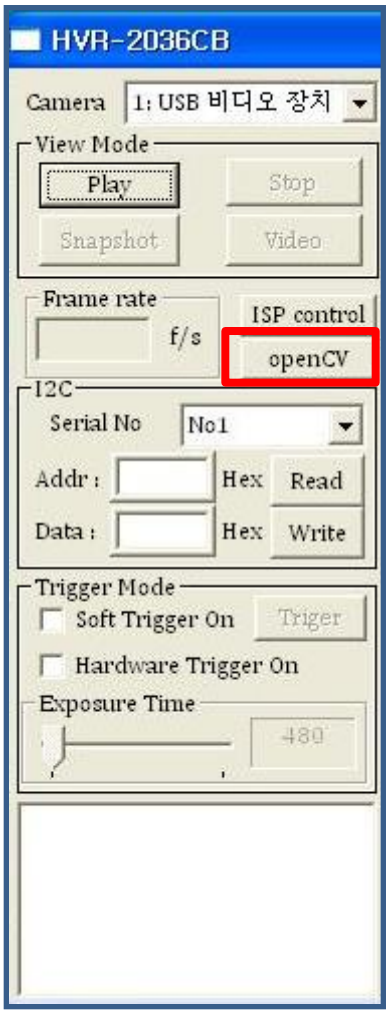
ISP control : ISP 설정 다이얼로그 실행

Fig 5. Main Control Dialog – ISP Control

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.



4.3.4 OpenCV

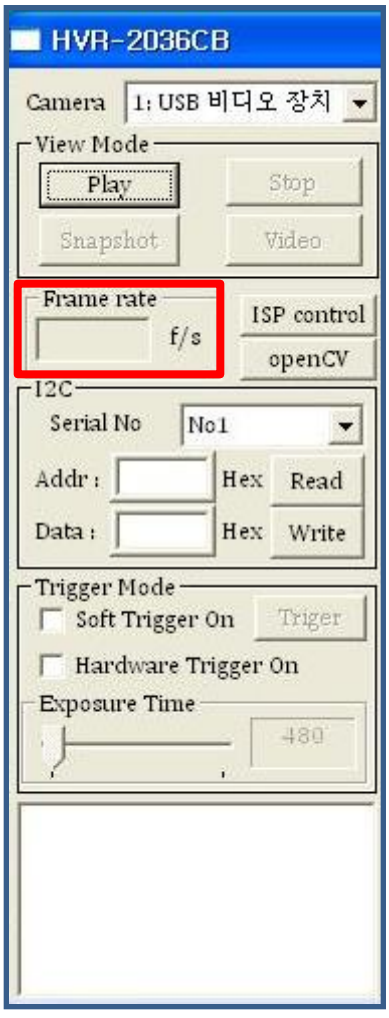
OpenCV 라이브러리의 이용하여 영상 데이터에 기본적인 효과를 적용

Fig 6. Main Control Dialog - OpenCV

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.



4.3.5 Frame Rate

영상데이터를 카메라로부터 입력 받아 화면에 출력하는 속도. (Frame/Sec)

Fig 7. Main Control Dialog – Frame Rate

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.

4.3.6 Serial No

사용하고자 하는 카메라의 Serial No 를 선택한다.

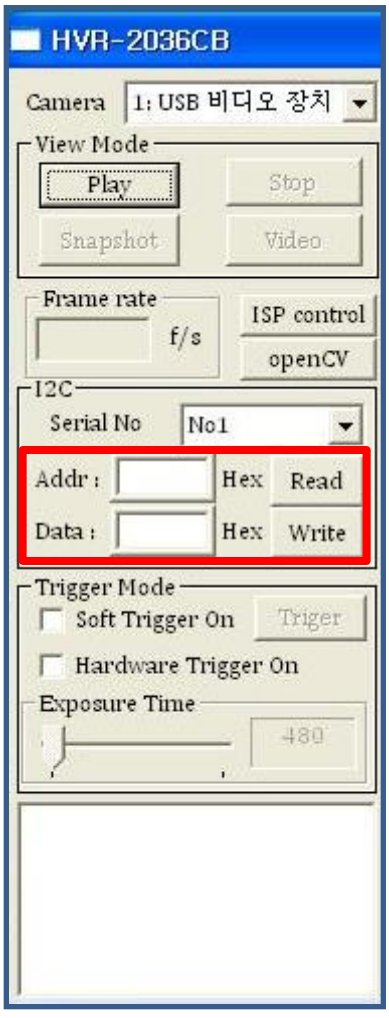
(1001, 1002, 1003, 1104)

Fig 8. Main Control Dialog – Frame Rate

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.



4.3.7 I2C

I2C 인터페이스를 이용하여 카메라에 데이터를 쓰거나 읽는다.

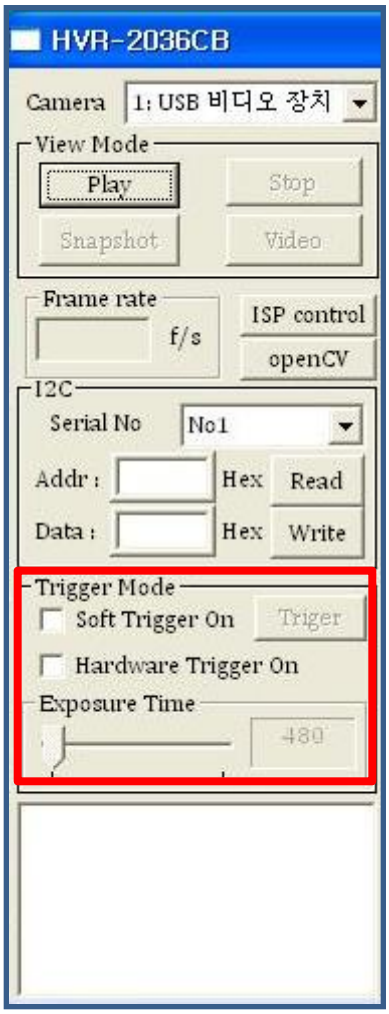
```
IIC_Read16i(
Serial_Number,  // 카메라 시리얼번호
I2cSubAddr);    // 레지스트리 주소

IIC_Write16i (
Serial_Number, // 카메라 시리얼 번호
I2cSubAddr,    // 레지스트리 주소
I2cValue);     // 데이터 값
```

Fig 9. Main Control Dialog – I2C

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.



4.3.8 Trigger Mode

트리거 관련 설정을 할 수 있다.

- Hardware Trigger On : 하드웨어 트리거를 사용
- Software Trigger On: 소프트웨어 트리거를 사용
  (*두개를 동시에 사용할 수 없다.)
- Triger 버튼 : 소프트웨어 트리거를 위한 버튼으로 버튼을 누르게 되면 한 프레임을 가져와 출력한다.
- 슬라이드바 : 트리거를 위한 것으로 Exposure time(노출시간)을 설정한다.

※ 소프트웨어 트리거 사용시 하드웨어 트리거 포트는 오픈 되어 있어야 한다.

Fig 10. Main Control Dialog – I2C

4. Software Architecture

4.3 Main Control Dialog

프로그램 전체를 운영하는 Dialog base의 controller로서, 그 역할을 설명한다.

4.3.9 상태정보

카메라의 상태나 실행 정보 등을 출력

Fig 11. Main Control Dialog – 상태 정보 창

4. Software Architecture

4.4 ISP(Image Signal Processing) Control Dialog

ISP를 설정할 수 있는 다이얼로그

2.4.1 White Balance

카메라로부터 입력된 영상의 GAIN에 대한 가중치를 설정

"AGC" Check Box : Auto Gain control 사용 유무로 사용시 슬라이드바의 GAIN값은 무시된다.

Min : 16
Max : 64

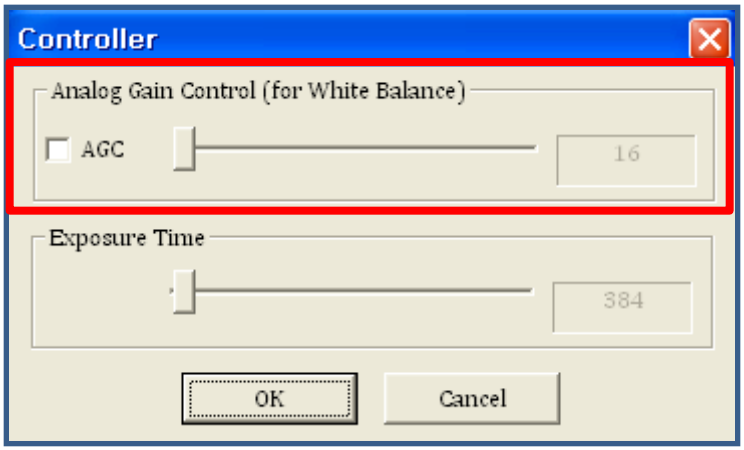


Fig 12. Soft ISP Control Dialog – 화이트 발란스

4. Software Architecture

4.4 ISP(Image Signal Processing) Control Dialog

ISP를 설정할 수 있는 다이얼로그

2.4.2 Exposure Time

카메라로부터 입력된 영상의 ExposureTime(노출 시간)을 설정한다.

Min : 1
Max : 32767

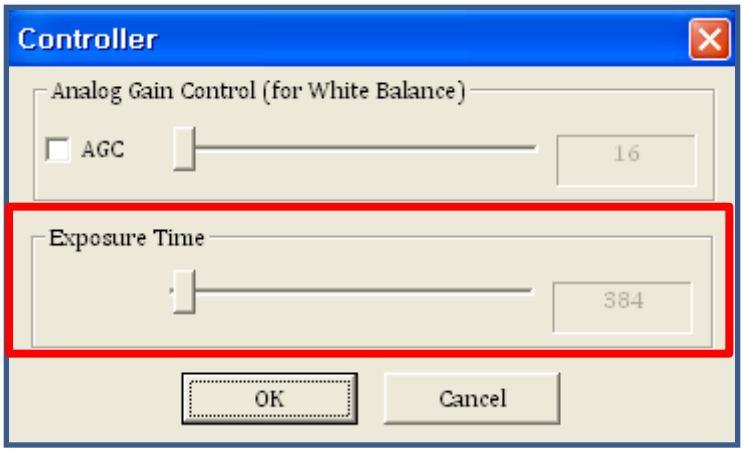
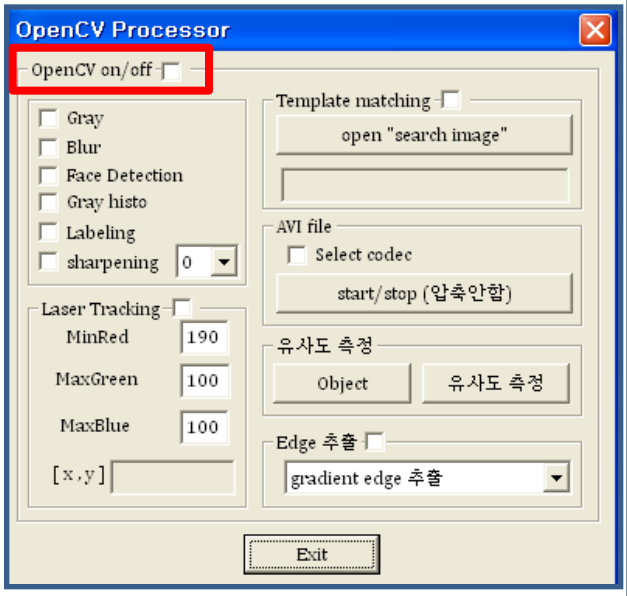


Fig 13. Soft ISP Control Dialog – 화이트 발란스

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.1 OpenCV On/Off

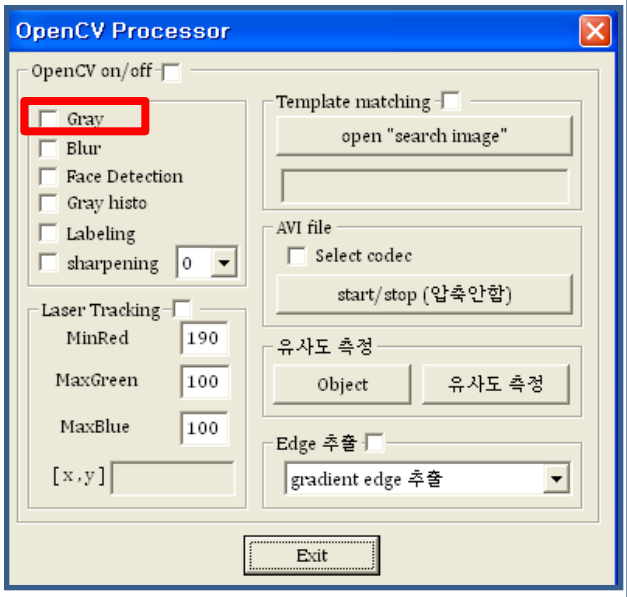
OpenCV 기능 적용 유무선택

Fig 14. OpenCV Control Dialog – OpenCV on/off

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.2 Gray

영상을 회색영상으로 보여준다.

CVAPI(void) cvCvtColor(

const CvArr* src,

CvArr* dst,

int code);

Src : 원본영상

Dst : 출력영상

Code : 변형타입(BGR2GRAY)

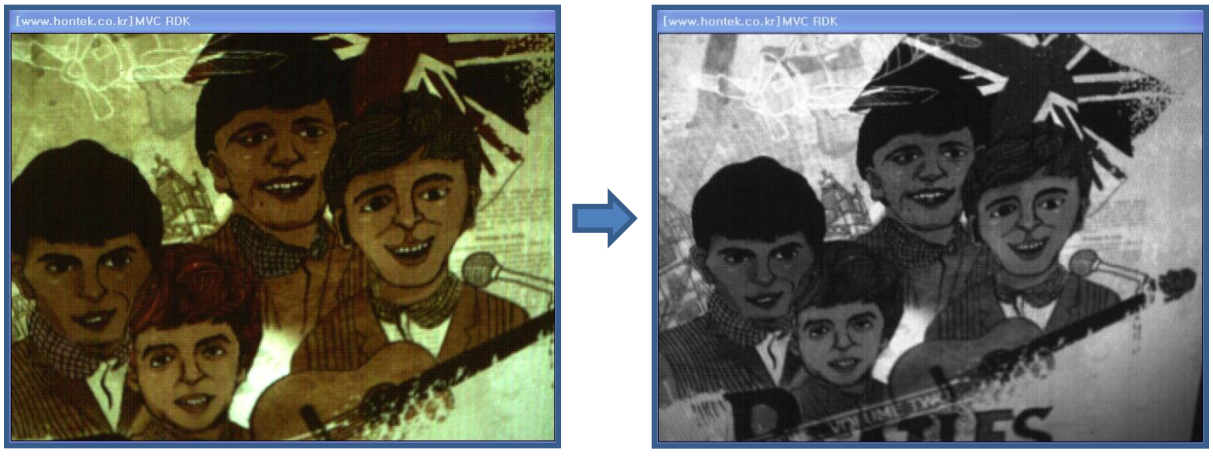
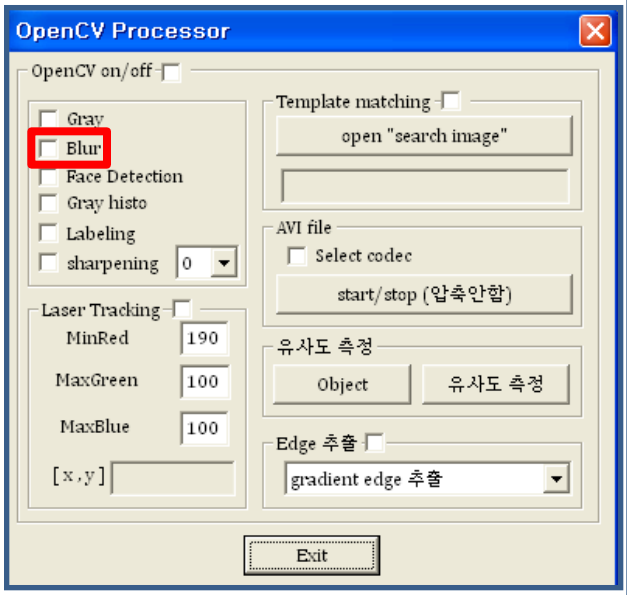


Fig 15. OpenCV Control Dialog - Gray

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.3 Blur

영상을 흐리게 나타낸다.

```
CVAPI(void) cvSmooth(
const CvArr* src, // 입력 영상
CvArr* dst, // 출력 영상
int smoothtype, // 블러링 타입
int param1,    // 영상 너비
int param2,    // 영상 높이
);
```

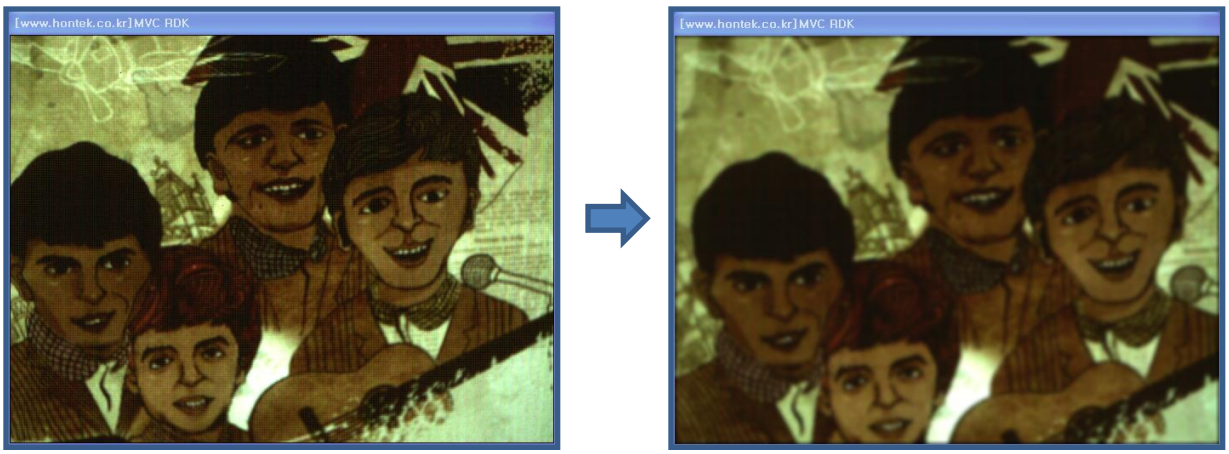
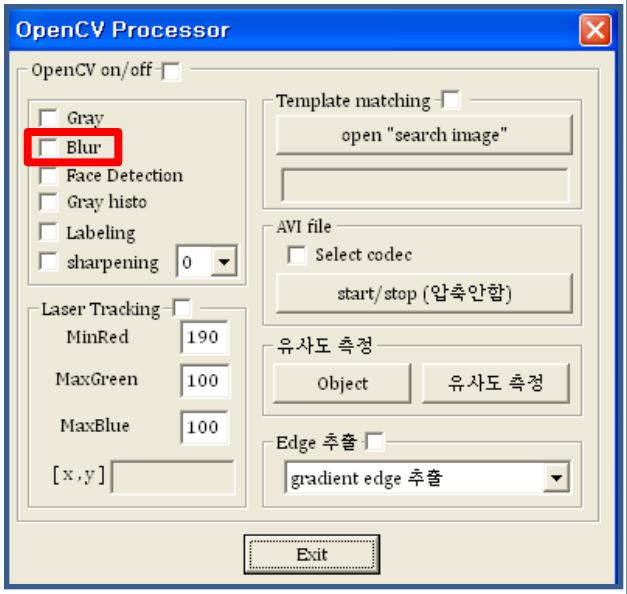


Fig 16. OpenCV Control Dialog - Blur

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



45.4 Face Detection

화면에서 얼굴을 찾아낸다.

IplImage *

MVC_OpenCV_FaceDetection(

IplImage* image);

Image : 원본 영상

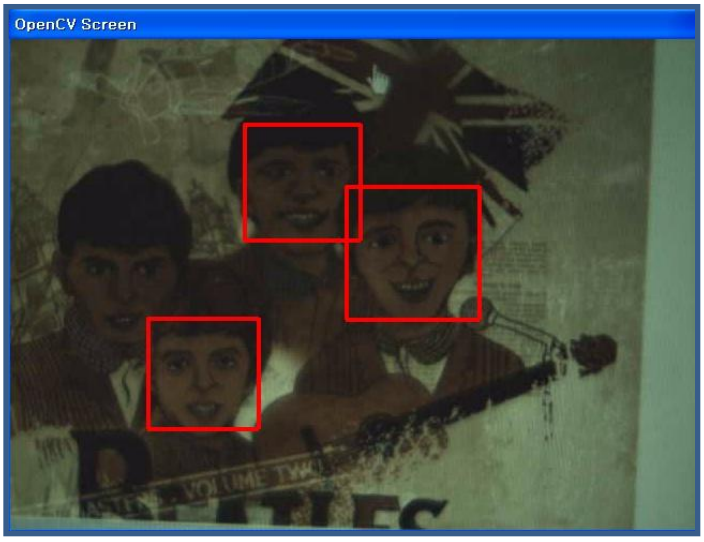
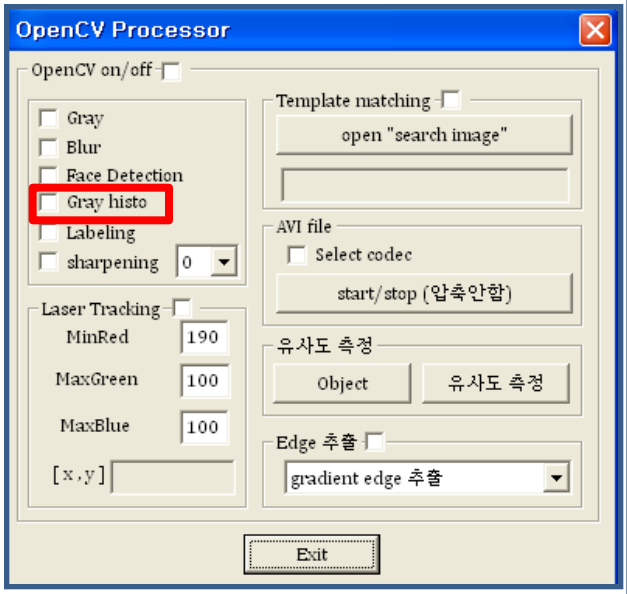


Fig 17. OpenCV Control Dialog – Face Detection

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.5 Gray Histo

영상의 밝기분포를 보여준다.

CV_INLINE void cvCalcHist(
IplImage** image,
CvHistogram* hist,
int accumulate CV_DEFAULT(0),
const CvArr* mask
CV_DEFAULT(NULL))

Image : 원본데이터
Hist : 히스토그램 데이터 구조체
Accumulate : 각 빈도 누적 여부
Mask : 마스크 영상

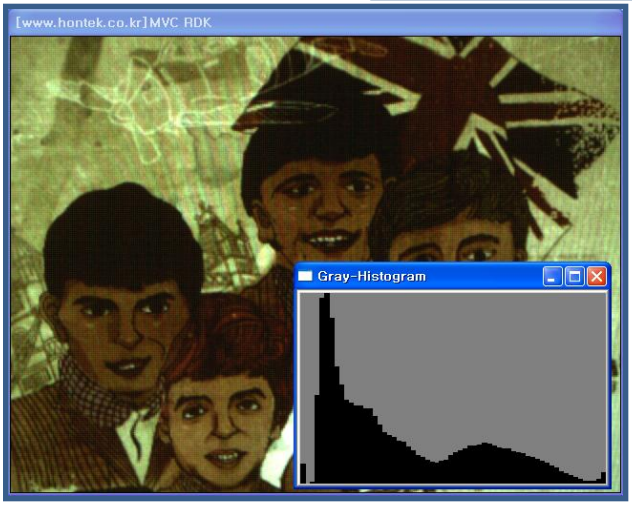
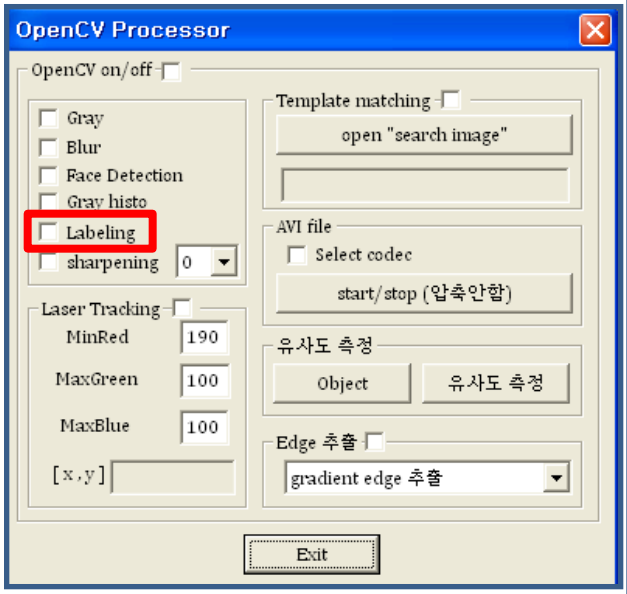


Fig 18. OpenCV Control Dialog – Gray Histo

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.6 Labeling

레이블링 기능

레이블링 : 인접한 화소에 모두 같은 번호(Label)를 붙이고 연결되지 않은 다른 성분에는 다른 번호를 붙이는 것

MVC_OpenCV_Labeling(
IplImage *imgIntP,
IplImage *imgGray);

imgIntP : 원본 영상 데이터
imgGray : 효과를 저장할 버퍼

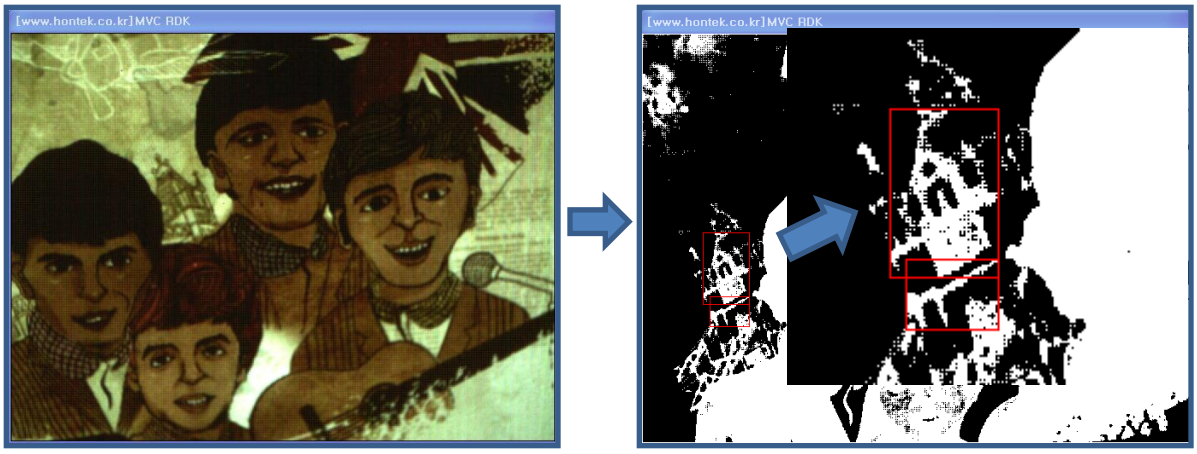
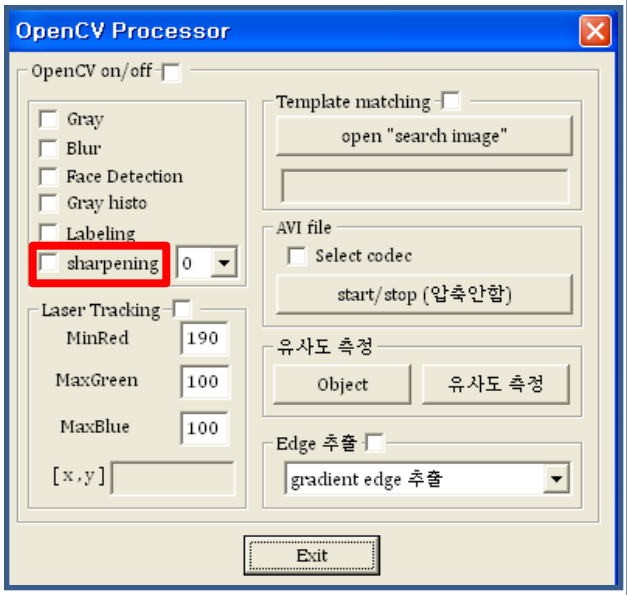


Fig 19. OpenCV Control Dialog - Labeling

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.7 Sharpening

영상을 선명하게 만든다.

IplImage*
MVC_OpenCV_Sharpening(
IplImage* imgIntP,
int m_shapeninglevel);

imgIntP : 원본데이터
m_shapeninglevel : level 0~5

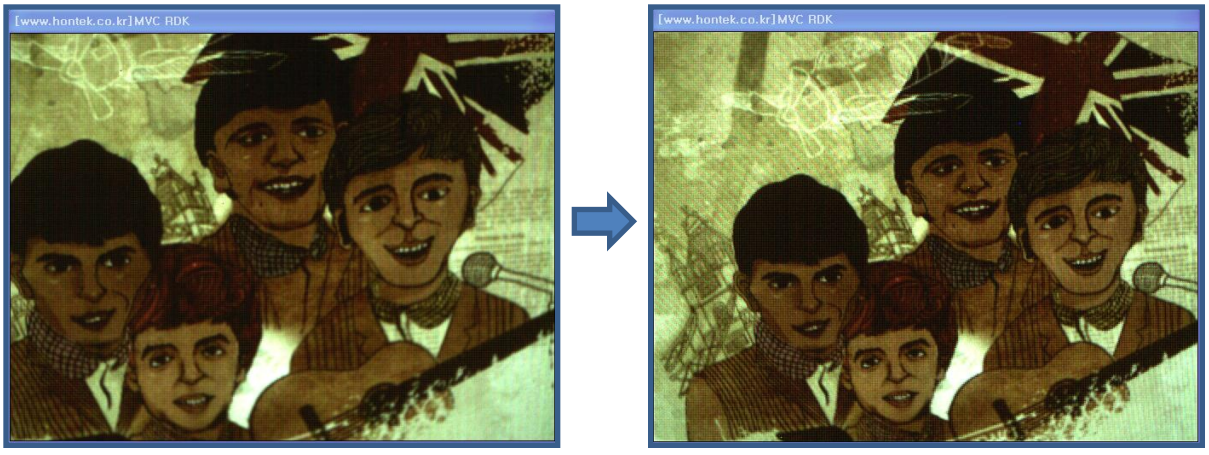
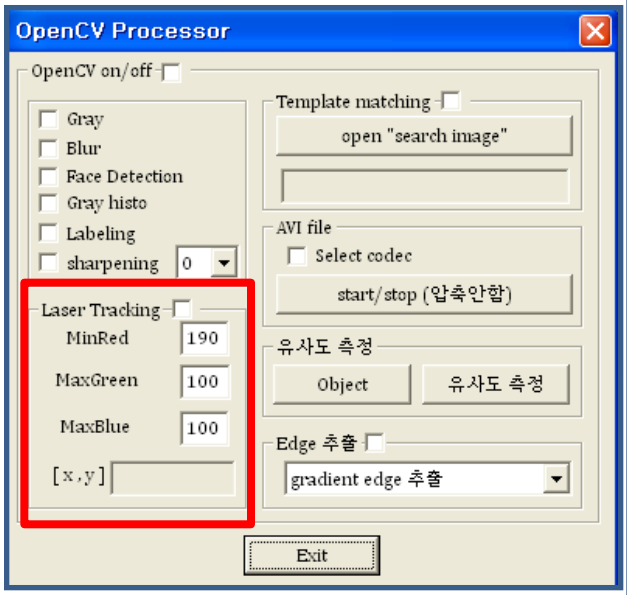


Fig 20. OpenCV Control Dialog - Sharpening

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.8 Laser Tracking

영상에서 붉은색을 찾는다.

```
char * __stdcall
MVC_OpenCV_convertRGB(
IplImage* img, // 원본 이미지
LPBYTE pBuf,   // 결과 저장 버퍼
int iWidth,    // 영상 너비
int iHeight,   // 영상 높이
int minred,    // Min Red Color
int maxgreen,  // Max Green Color
int maxblue    // Max blue Color
);
```

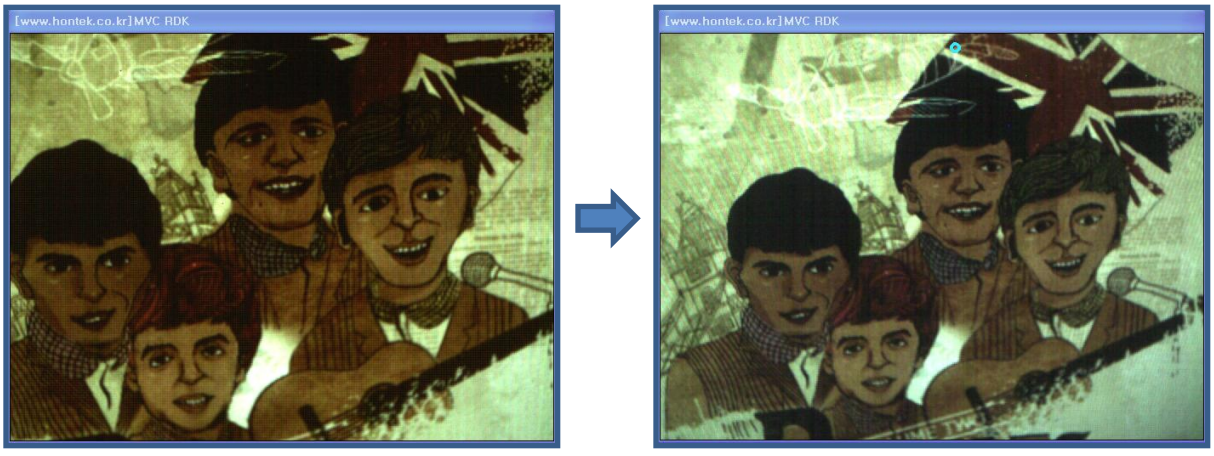
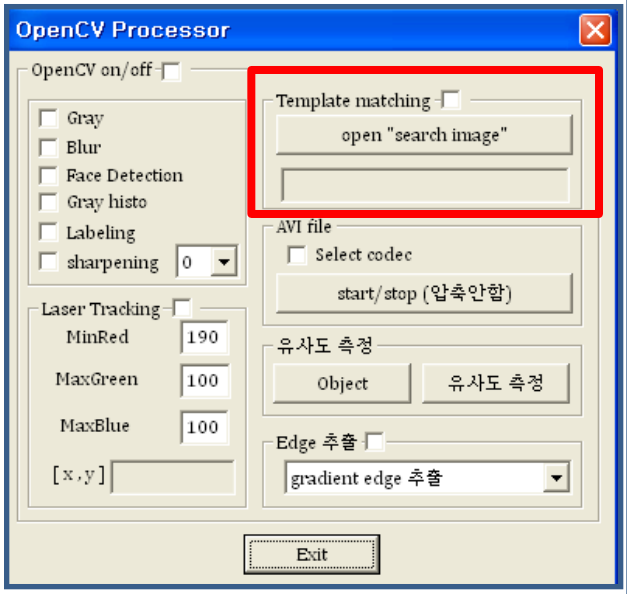


Fig 21. OpenCV Control Dialog – Laser Tracking

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.9 Template Matching

영상에서 특정영상을 찾는다.

CVAPI(void) cvMatchTemplate(
const CvArr* image,
const CvArr* templ,
CvArr* result,
int method);

Image : 영상 데이터
Templ : 검색할 영상
Result : 결과 영상
Method : 검출방법

CV_TM_SQDIFF	0
CV_TM_SQDIFF_NORMED	1
CV_TM_CCORR	2
CV_TM_CCORR_NORMED	3
CV_TM_CCOEFF	4
CV_TM_CCOEFF_NORMED	5

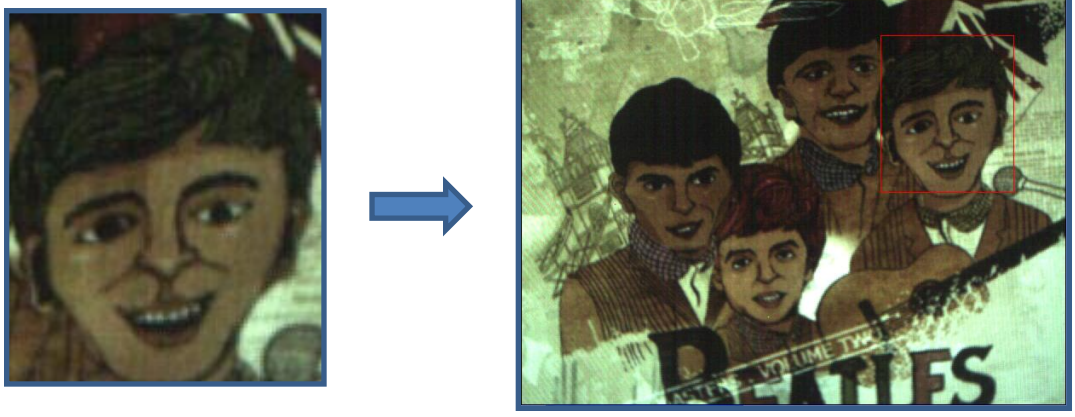
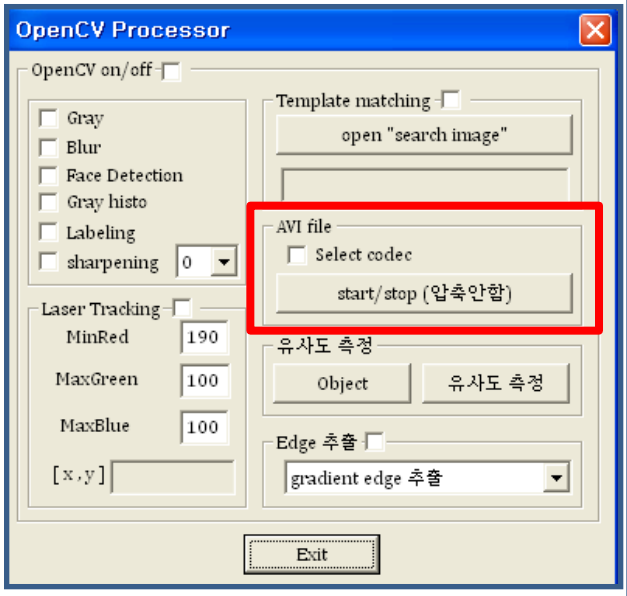


Fig 22. OpenCV Control Dialog – Template Matching

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.10 동영상 저장

동영상으로 저장한다.

```
// 저장시작
VideoOut =
cvCreateVideoWriter(
tmp,
-1,
value-5,
cvGetSize(imgGray),
1);
// 저장
cvWriteFrame(VideoOut, imgIntP);
// 저장 종료
cvReleaseVideoWriter(&VideoOut);
```

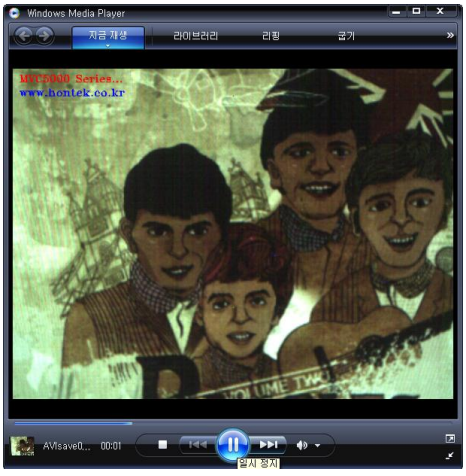
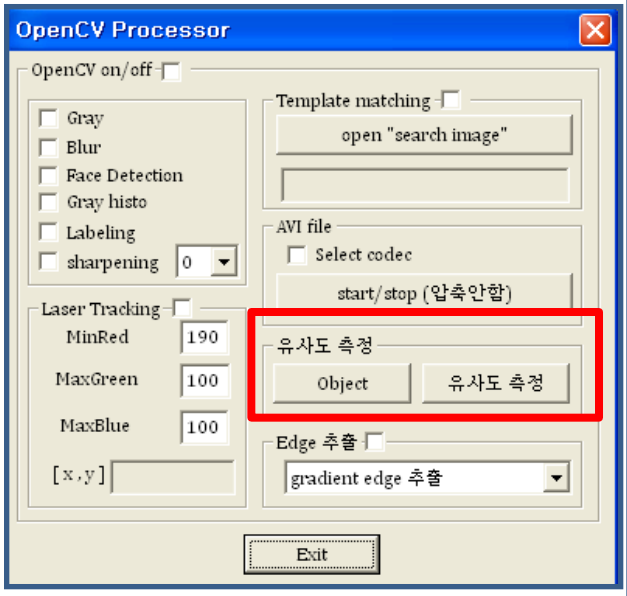


Fig 23. OpenCV Control Dialog – 동영상 저장

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.11 유사도 측정

두 개의 영상간에 유사도를 측정 한다.

IplImage *
MVC_OpenCV_Match(
IplImage *src_image1,
IplImage *src_image2,
LPBYTE pBuf);

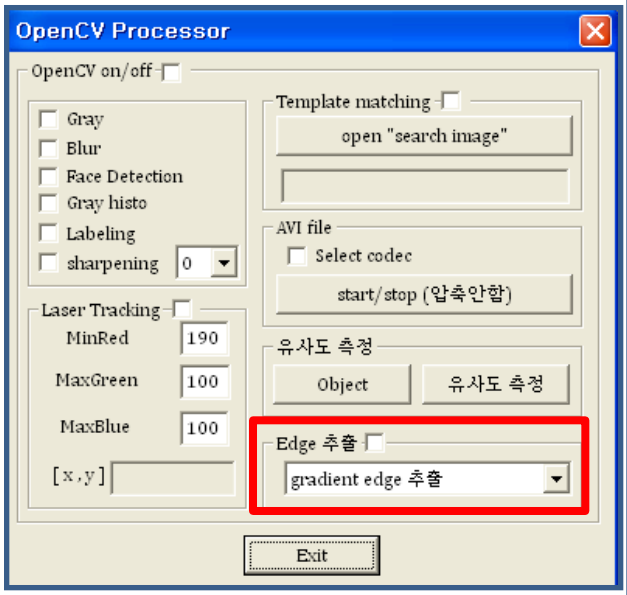
src_image1 : 원본 영상 데이터
src_image2 : 비교영상 데이터
pBuf : 비교결과 스트링

Fig 24. OpenCV Control Dialog – 유사도 측정

4. Software Architecture

4.5 OpenCV Control Dialog

OpenCV 라이브러리를 이용한 예.



4.5.12 Edge 추출

윤곽선을 추출한다.

IplImage * MVC_OpenCV_Edge(
IplImage *imgCanny,
IplImage *imgGray,
int m_seledge);

imgCanny :원본 영상 버퍼
imgGray : 수정된 영상을 담을 버퍼
m_seledge : Edge검출 방법

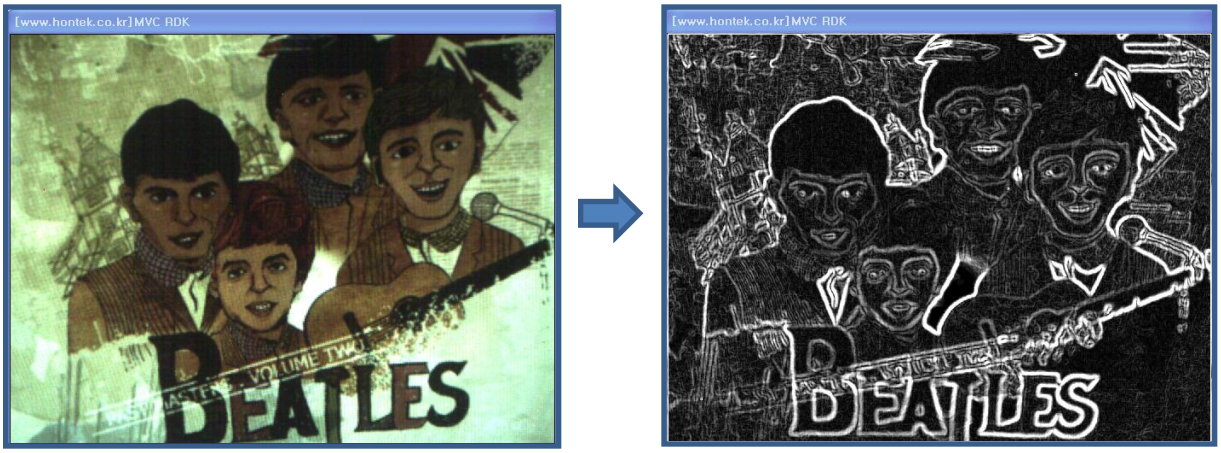
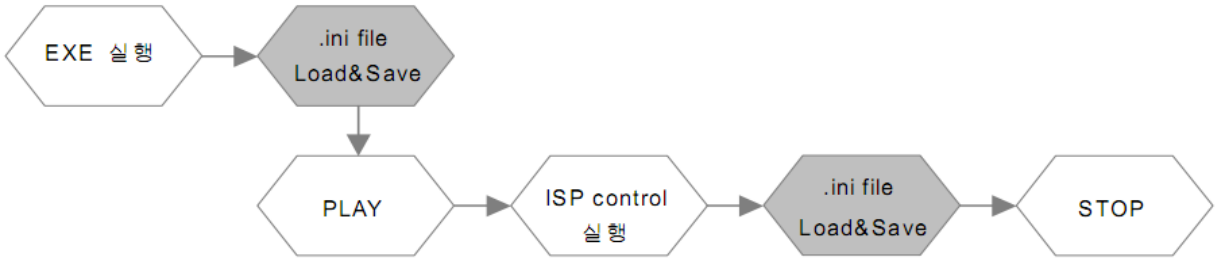


Fig 25. OpenCV Control Dialog – Edge 추출

5. System File Description

5.1 MVC_VISION.ini

실행파일과 같은 디렉토리에 위치하며, 초기 설정들의 기본값을 File Read/Write 방식으로 관리



[MVC_INIT_XXXXX] 제품 모델별로 구성됨.

Visioni_Select=0 [SensorType]에 대한 초기값

[MVC_ISPi_XXXXX] 제품 모델별로 구성됨.

AWB=vvv	[ISP control] GAIN 초기값
AEC_VALUE=vvv	[ISP control] AEC manual 초기값
AGC_ENABLE=vvv	[ISP control] AGC on/off 초기값
AGC_VALUE=vvv	[ISP control] AGC manual 초기값

Fig 26. MVC_VISION.ini

6. API Lists

6.1 API Access Flow

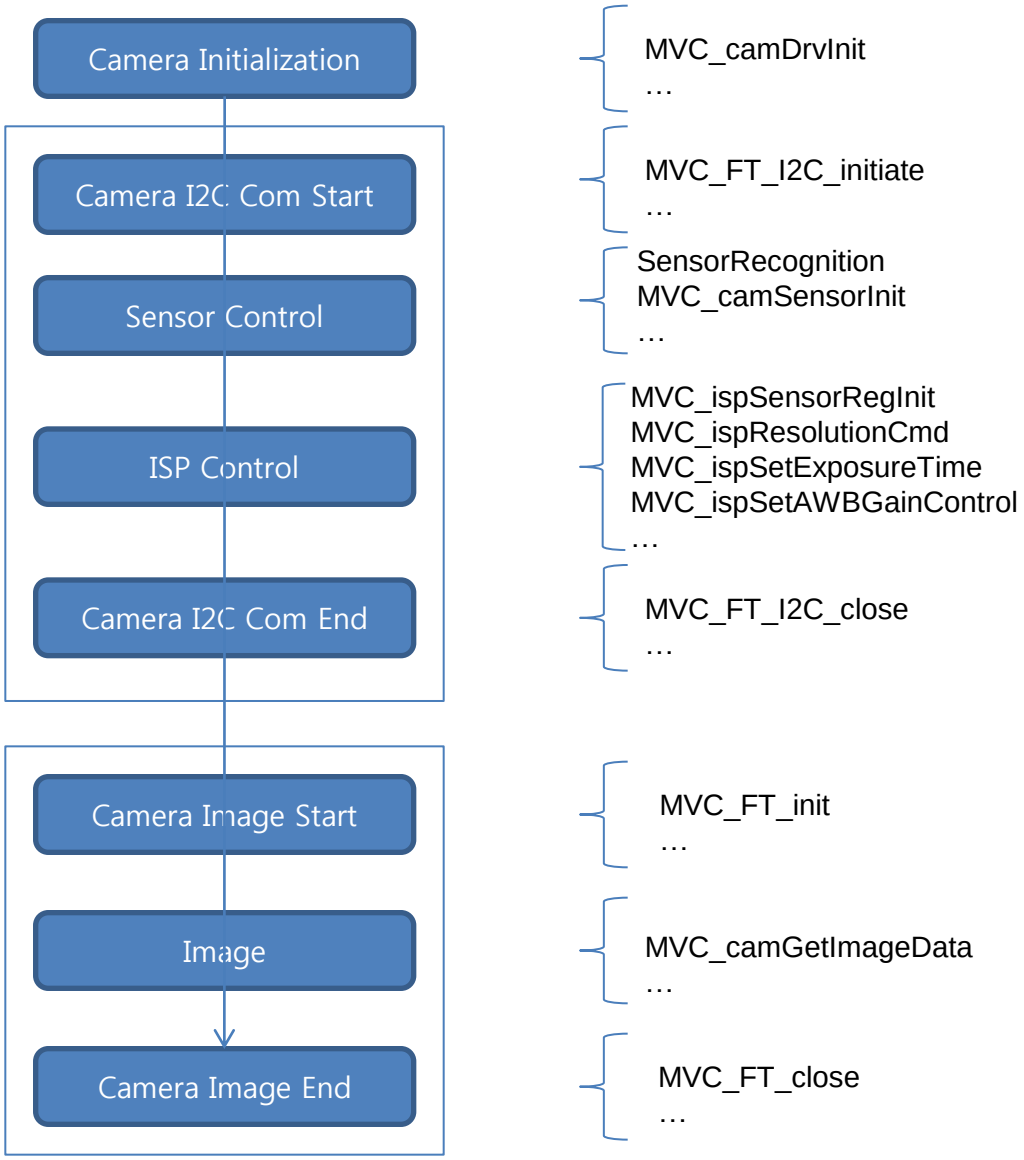


그림 4 : API acess Flow

6. API Lists

6.2 I2C Communication

API	내용
HVR_Init	카메라와 I2C통신을 시작한다
HVR_Release	카메라와 I2C통신을 종료한다. .
HVR_EmpiaispAutoGainControlEnable	Auto GAIN Control 기능 사용 유무 설정
HVR_EmpiaispGetAWBGainControlRange	Gain 값과 최대 최소값을 얻는다..
HVR_EmpiaispSetAWBGainControl	Gain 값을 설정한다.
HVR_EmpiaispSetExposureTime	Exposure Time(빛을 받아들이는 시간)을 설정한다.
HVR_EmpiaispGetExposureRange	Exposure Time(빛을 받아들이는 시간)의 최대, 최소값, 설정값을 얻는다.
IIC_Read16i	I2C를 이용해서 데이터를 읽는다
IIC_Write16i	I2C를 이용해서 데이터를 쓴다.
SoftTrigerOnOff	소프트웨어 트리거 설정을 On/Off 한다.
TriggerOnOff	하드웨어 트리거 설정을 On/Off한다.
SoftTriger	소프트웨어 트리거를 수행한다.
Slider	트리거 설정시 Exposure time 설정

6. API Lists

6.3 OpenCV 관련 함수

API	내용
HVR_OpenCV_Create	OpenCV 프로세스 시작.
HVR_OpenCV_Release	OpenCV 프로세스 종료.
HVR_OpenCV_Display	OpenCV 출력
HVR_OpenCV_StartAVI	동영상 저장 시작.
HVR_OpenCV_StopAVI	동영상 저장 종료
HVR_OpenCV_TM_targetFile	Template Matching 할 대상파일 설정
HVR_OpenCV_LoadObjectOpenCV	유사도 측정을 위한 대상 화면 설정
HVR_OpenCV_MatchingObjectOpenCV	유사도 측정
HVR_OpenCV_LaserTracking	Laser Tracking 동작
HVR_OpenCV_SelEdge	Edge 검출 수행
HVR_OpenCV_Labeling	레이블링 기능 수행
HVR_OpenCV_Sharpening	Sharpening 기능 수행

7. API 설명

7.1 HVR_Init

카메라와 I2C통신을 시작한다

구분	내용
Syntex	Int HVR_Init ()
Parameter	없음
Return Value	성공 : 1, 실패 : 0
Sample	Int m_etLib2_Init = HVR_Init();

7. API 설명

7.2 HVR_Release

카메라와 I2C통신을 종료한다.

구분	내용
Syntex	Int HVR_Release()
Parameter	없음
Return Value	성공 : 1, 실패 : 0
Sample	Int m_etLib2_Init = HVR_Init(); … if (m_etLib2_Init) HVR_Release();

7. API 설명

7.3 HVR_EmpiaispAutoGainControlEnable

카메라와 I2C통신을 시작한다.

구분	내용	
Syntex	int HVR_EmpiaispAutoGainControlEnable(WCHAR *Serial_Number, int Value);	
Parameter	Serial_Number	카메라 Serial_number
	Value	AGC_ENABLE:사용, AGC_DISABLE:미사용
Return Value	성공 : 1, 실패 : 0	
Sample	WCHAR Serial_Number[5]=L"1001"; Int m_etLib2_Init = HVR_Init(); HVR_EmpiaispAutoGainControlEnable(Serial_Number, AGC_ENABLE); HVR_Release();	

7. API 설명

7.4 HVR_EmpiaispGetAWBGainControlRange

Gain값과 최대 최소, 현재값을 얻는다...

구분	내용	
Syntex	Int HVR_EmpiaispGetAWBGainControlRange(WCHAR *Serial_Number, int *pMin, int *pMax, int *pValue);	
Parameter	Serial_Number	카메라 Serial_number
	pMin	GAIN 최대값
	pMax	GAIN 최소값
	pValue	GAIN 현재값
Return Value	성공 : 1, 실패 : 0	
Sample	WCHAR Serial_Number[5]=L"1001"; Int iMin, iMax, iWB_R_E; Int m_etLib2_Init = HVR_Init(); HVR_EmpiaispGetAWBGainControlRange(Serial_Number, &iMin, &iMax, &iWB_R_E); HVR_Release();	

7. API 설명

7.5 HVR_EmpiaispSetAWBGainControl

GAIN 값을 설정한다..

구분	내용	
Syntex	Int HVR_EmpiaispSetAWBGainControl(WCHAR *Serial_Number, int iGain);	
Parameter	Serial_Number	카메라 Serial_number
	Value	Exposure Time 설정값
Return Value	성공 : 1, 실패 : 0	
Sample	WCHAR Serial_Number[5]=L"1001"; Int m_etLib2_Init = HVR_Init(); VR_EmpiaispSetAWBGainControl(Serial_Number, 300); HVR_Release();	

7. API 설명

7.6 HVR_EmpiaispGetExposureRange

Exposure Time(빛을 받아들이는 시간)의 최대, 최소값, 설정값을 얻는다

구분	내용	
Syntex	int HVR_EmpiaispGetExposureRange(WCHAR *Serial_Number, int *pMin, int *pMax, int *pValue);	
Parameter	Serial_Number	카메라 Serial_number
	pMin	Exposure Time 최대값
	pMax	Exposure Time 최소값
	pValue	Exposure Time 현재값
Return Value	성공 : 1, 실패 : 0	
Sample	WCHAR Serial_Number[5]=L"1001"; Int iMin, iMax, iEC_E; Int m_etLib2_Init = HVR_Init(); HVR_EmpiaispGetExposureRange(Serial_Number, &iMin, &iMax, &iEC_E); HVR_Release();	

7. API 설명

7.7 HVR_EmpiaispSetExposureTime

Exposure Time(빛을 받아들이는 시간)을 설정한다…

구분	내용	
Syntex	Int HVR_EmpiaispSetExposureTime(WCHAR *Serial_Number, int Value);	
Parameter	Serial_Number	카메라 Serial_number
	Value	Exposure Time 설정값
Return Value	성공 : 1, 실패 : 0	
Sample	WCHAR Serial_Number[5]=L"1001"; Int m_etLib2_Init = HVR_Init(); HVR_EmpiaispSetExposureTime(Serial_Number, 20); HVR_Release();	

7. API 설명

7.8 IIC_Read16i

I2C를 이용해서 데이터를 읽는다

구분	내용	
Syntex	ULONG IIC_Read16i(WCHAR *Serial_Number, WORD wRegAddr);	
Parameter	Serial_Number	카메라 Serial_number
	wRegAddr	어드레스
Return Value	레지스트리로 부터 읽어온 데이터	
Sample	WCHAR Serial_Number[5]=L"1001"; ULONG I2cAddr = 0; Int m_etLib2_Init = HVR_Init(); I2cValue = IIC_Read16i(Serial_Number,0); HVR_Release();	

7. API 설명

7.9 IIC_Write16i

I2C를 이용해서 데이터를 쓴다

구분	내용	
Syntex	BOOL IIC_Write16i(WCHAR *Serial_Number, WORD wRegAddr, WORD wRegData);	
Parameter	Serial_Number	카메라 Serial_number
	wRegAddr	어드레스
	wRegData	데이터
Return Value	성공 : 1, 실패 : 0	
Sample	WCHAR Serial_Number[5]=L"1001"; ULONG I2cAddr = 0; Int m_etLib2_Init = HVR_Init(); IIC_Write16i (Serial_Number, 0xb, 10); HVR_Release();	

7. API 설명

7.10 SoftTrigerOnOff

소프트웨어 트리거 설정을 On/Off 한다

구분	내용	
Syntex	Void SoftTrigerOnOff(WCHAR *Serial_Number, BOOL chk);	
Parameter	Serial_Number	카메라 Serial_number
	chk	1: On, 0:OFF
Return Value	‘	
Sample	WCHAR Serial_Number[5]=L"1001"; Int m_etLib2_Init = HVR_Init(); SoftTrigerOnOff(Serial_Number, 1); HVR_Release();	

7. API 설명

7.11 TriggerOnOff

트리거 설정을 On/Off한다.

구분	내용	
Syntex	Void TriggerOnOff(WCHAR *Serial_Number, BOOL chk);	
Parameter	Serial_Number	카메라 Serial_number
	chk	1: On, 0:OFF
Return Value		
Sample	WCHAR Serial_Number[5]=L"1001"; Int m_etLib2_Init = HVR_Init(); TrigerOnOff(Serial_Number, 1); HVR_Release();	

7. API 설명

7.12 SoftTriger

소프트웨어 트리거를 수행한다

구분	내용	
Syntex	Void SoftTriger(WCHAR *Serial_Number);	
Parameter	Serial_Number	카메라 Serial_number
Return Value		
Sample	WCHAR Serial_Number[5]=L"1001"; Int m_etLib2_Init = HVR_Init(); SoftTrigerOnOff(Serial_Number, 1); SoftTriger(Serial_Number); HVR_Release();	

7. API 설명

7.13 Slider

Exposure time(노출시간) 을 설정한다.

구분	내용	
Syntex	Void Slider(WCHAR *Serial_Number, int pos);	
Parameter	Serial_Number	
	pos	
Return Value		
Sample	int pos = m_TriggerSlide.GetPos(); Slider(Serial_Number, pos);	

7. API 설명

7.14 HVR_OpenCV_Create

OpenCV 프로세스 시작.

구분	내용
Syntex	Int HVR_OpenCV_Create();
Parameter	
Return Value	성공 : 1
Sample	BOOL gAVI = 0, gGRAYhisto = 0; HVR_OpenCV_Create(); HVR_OpenCV_Release(gAVI, gGRAYhisto);

7. API 설명

7.15 HVR_OpenCV_Release

OpenCV 프로세스 종료.

구분	내용	
Syntex	Int HVR_OpenCV_Release(BOOL AVI, BOOL GRAYhisto);	
Parameter	AVI	현재 동영상 저장 유무
	GRAYhisto	현재 Grayhist 설정 유무
Return Value	성공 : 1	
Sample	BOOL gAVI = 0, gGRAYhisto = 0; HVR_OpenCV_Create(); HVR_OpenCV_Release(gAVI, gGRAYhisto);	

7. API 설명

7.16 HVR_OpenCV_Display

OpenCV 출력.

구분	내용	
Syntex	Int HVR_OpenCV_Display(BYTE *pBuffer,long lBufferLen, LPBYTE pBuf,BOOL gGRAY,BOOL gBLUR,BOOL gGRAYhisto,BOOL gCANNY,BOOL gFACEDETECTION,BOOL gTMATCHING,BOOL gAVI,BOOL bTargetFile,BOOL bLaserSearch,HWND ghwndApp);	
Parameter	pBuffer,	출력할 영상 데이터
	lBufferLen	영상 데이터 길이
	pBuf	효과를 적용한 영상데이터를 임시로 저장하기 위한 버퍼
	gGRAY	흑백효과 사용 유무
	gBLUR	Blur 효과 사용 유무
	gGRAYhisto	Gray histograph 사용 유무
	gCANNY	Edge 기능 사용 유무
	gFACEDETECTION	얼굴 검출 기능 사용 유무
	gTMATCHING	Template Matching 사용 유무
	gAVI	동영상 저장 상태
	bTargetFile	Template Matching 을 위한 비교파일 설정 유무
	bLaserSearch	Laser Tracking 기능 사용 유무
	ghwndApp	효과를 적용한 영상데이터를 출력할 화면의 핸들
Return Value	성공 : 1	
Sample	HVR_OpenCV_Create(); HVR_OpenCV_Display(pBuffer, lBufferLen, pBuf, gGRAY, gBLUR, gGRAYhisto, gCANNY, gFACEDETECTION, gTMATCHING, gAVI, bTargetFile, gLaser, ghwndApp); HVR_OpenCV_Release(gAVI, gGRAYhisto);	

7. API 설명

7.17 HVR_OpenCV_StartAVI

동영상 저장 시작.

구분	내용	
Syntex	Void HVR_OpenCV_StartAVI(bool mode, int value);	
Parameter	mode	Codec 사용 유무(1: 사용, 2:미사용)
	value	FrameRate
Return Value		
Sample	HVR_OpenCV_StartAVI(0, 20); … HVR_OpenCV_StopAVI();	

7. API 설명

7.18 HVR_OpenCV_StopAVI

동영상 저장 종료.

구분	내용
Syntex	void HVR_OpenCV_StopAVI();
Parameter	
Return Value	
Sample	HVR_OpenCV_StartAVI(0, 20); … HVR_OpenCV_StopAVI();

7. API 설명

7.19 HVR_OpenCV_TM_targetFile

Template Matching 할 대상파일 설정

구분	내용	
Syntex	BOOL HVR_OpenCV_TM_targetFile(LPCTSTR filepath);	
Parameter	filepath	비교할 대상 파일이름
Return Value	성공 : 1	
Sample	if(HVR_OpenCV_TM_targetFile(targetfileName)) bTargetFile=TRUE;	

7. API 설명

7.20 HVR_OpenCV_LoadObjectOpenCV

유사도 측정을 위한 대상 화면 설정

구분	내용
Syntex	Void HVR_OpenCV_LoadObjectOpenCV()
Parameter	
Return Value	
Sample	HVR_OpenCV_LoadObjectOpenCV(); … HVR_OpenCV_MatchingObjectOpenCV();

7. API 설명

7.21 HVR_OpenCV_MatchingObjectOpenCV

유사도 측정

구분	내용
Syntex	Void HVR_OpenCV_MatchingObjectOpenCV();
Parameter	
Return Value	
Sample	HVR_OpenCV_LoadObjectOpenCV(); … HVR_OpenCV_MatchingObjectOpenCV();

7. API 설명

7.22 HVR_OpenCV_LaserTracking

Laser Tracking 동작

구분	내용	
Syntex	Void HVR_OpenCV_LaserTracking(int red, int green, int blue);	
Parameter	Red	탐지할 위치의 red 최소값(0~255)
	Green	탐지할 위치의 green 최대값(0~255)
	Blue	탐지할 위치의 blue 최대값(0~255)
Return Value		
Sample	HVR_OpenCV_LaserTracking(m_red, m_green, m_blue);	

7. API 설명

7.23 HVR_OpenCV_SelEdge

Edge 검출

구분	내용	
Syntex	Void HVR_OpenCV_SelEdge(int no);	
Parameter	no	알고리즘 0 : 그래디언트 에지 추출 1 : 영역 필터 에지 추출 2~5 : 소벨 에지 추출 6~8 : 프리윗 에지 추출 7 : 프레이첸 에지 추출 8: 서터캐스틱 에지 추출 9 : 로버츠 에지 추출 10~11 : 라플라시안 에지 추출 12 : 라플라시안 샤프닝 13~14 : LoG 에지 추출 15~17 : 캐니 에지 추출
Return Value		
Sample	HVR_OpenCV_SelEdge(15);	

7. API 설명

7.24 HVR_OpenCV_Labeling

OpenCV 라이브러리 레이블링 기능 수행

구분	내용	
Syntex	void HVR_OpenCV_Labeling(int state);	
Parameter	state	1: 적용, 0 : 미적용
Return Value		
Sample	HVR_OpenCV_Labeling(gLabeling);	

7. API 설명

7.25 MVC_OpenCV_Sharpening

OpenCV 라이브러리 Sharpening 기능 수행

구분	내용	
Syntex	Void HVR_OpenCV_Sharpening(bool state, int level);	
Parameter	state	기능 사용 유무(1:사용, 0: 미사용)
	level	Level(0 ~ 5)
Return Value	Sharpening 영상 구조체	
Sample	HVR_OpenCV_Sharpening(gSharpening, m_Sharpening.GetCurSel());	