

윈도우즈 기반 영상 감시 시스템에서의 Tearing 현상 개선

정희원 정 언 권*, 이 동 학 **, 정 선 태 ***

A Study on the Improvement of Tearing Artifact for Windows-Based Visual Monitoring Systems.

Yeon-Gweon Jeong*, Dong-Hak Lee**, Sun-Tea Chung*** *Regular Members*

요 약

아날로그 모니터를 사용하는 디스플레이 시스템의 경우, 프레임 버퍼의 영상 정보 갱신 주기와 모니터 주사 주기가 맞지 않는 경우, 장면 변화가 심한 동영상 디스플레이시에 화면이 상단과 하단으로 다른 영상이 나타나는 tearing 현상이 발생한다. 다중 카메라 채널을 디스플레이하여야 하는 영상 감시 시스템인 DVR(Digital Video Recorder) 시스템에서는 프레임 버퍼 갱신율이 매우 높기 때문에 tearing 현상이 심하게 된다.

본 논문에서는 윈도우즈 기반 DVR에서 디스플레이 속도 성능의 저하가 별로 없으면서, tearing 현상을 방지한 개선된 디스플레이 시스템 구조를 제안한다. 제안된 디스플레이 시스템의 효율성은 실험을 통하여 입증하였다.

ABSTRACT

In display systems employing analog monitors, the tearing artifact such that an window screen is divided into two parts showing different scenes can occur when the change of scenes in the moving pictures is very fast, but the frame buffer's refresh rate does not match the monitor's scanning frequency. It is especially noticeable at high frame rate. DVR system is a recently popularized visual monitoring system. The tearing artifacts becomes more serious since the frame buffer's refresh rate is very high due to the requirement of multi channel display in the DVR.

In this paper, we propose an improved display system for windows-based DVR systems which prevents the tearing artifacts without deterioration of display speed performance. The efficiency of the proposed display system is verified through experiments.

I. 서 론

영상 감시 시스템은 카메라에서 획득한 영상을 통하여 현장을 실시간으로 감시하고 추후 검색을 위해 획득된 비디오 데이터를 저장하며, 원격 감시를 지원한다. 현재 영상 감시시스템으로 각광을 받고 있는 DVR(Digital Video Recorder)은 종래의 아날로그 기반 시스템과 달리 CCTV 카메라에서 획득한 아날로그 영상을 디지털 영상으로 변환하고

이를 컴퓨터 시스템으로 받아들여 모니터에 디스플레이하고 원격감시를 위해서 전송하며, 추후 검색을 위해 저장하는 새로운 기술의 영상 감시 시스템으로 각광을 받고 있다^{1,2)}.

일반적으로 DVR에서 디스플레이 모니터로 아날로그 모니터(CRT 및 아날로그 지원 LCD)를 많이 사용한다. 아날로그 모니터를 사용하는 경우, 그래픽 어댑터는 프레임 버퍼(비디오 램)에서 컬러 정보를 취하여 이를 RAMDAC을 통해 아날로그 신호로

* 송실대학교 정보통신전자공학부 (jygyj@syscon.soongsil.ac.kr),

** (주)Yoos Technology 연구소

*** 송실대학교 정보통신전자공학부 (cst@syscon.soongsil.ac.kr)

※ 본 논문은 2002년 4월 JCCI 학술대회에서 우수논문으로 선정되어 게재 추천된 논문입니다.

변환하고 이를 아날로그 모니터에 제공한다. 아날로그 모니터는 이 정보를 이용하여 고정 리플레쉬 주기(50~120 Hz)로 한 화면을 주사(raster scan)하여 모니터 화면에 영상이 디스플레이 되도록 한다^[3]. 따라서, 프레임 버퍼 내용의 변경은 다음 리플레쉬 주기(period)전에는 모니터 스크린에서 즉시 갱신되어 보이지 않게 되지는 않는다. 프레임 버퍼 내용의 갱신 시점과 아날로그 모니터에서의 화면 디스플레이 시점이 동기화되지 않은 경우에, 처리하는 영상이 이전 영상에 비해 급격하게 변하는 경우에는 모니터에 출력되는 화면이 찢겨 나타나는 tearing 현상(artifact)이 나타날 수 있다^[4]. Tearing 현상의 구체적인 예로는 화면 내에 화면을 수평으로 가로 지르는 경계선이 나타나고 이를 경계로 상단과 하단에 다른 영상이 나타나는 경우나, 움직이는 스프라이트(작은 물체)의 상단은 이전 위치에 하단은 새 위치에 보이는 경우를 들 수 있다. DVR에서는 다채널 영상을 동시에 모니터에 디스플레이 될 것을 요구하며 초당 최대 240프레임 이상의 화면의 변화가 이루어지는 경우도 있어 프레임 버퍼 내용의 갱신이 보다 빠르게 이루어지게 되므로, 이러한 tearing 현상이 더 심해진다.

Tearing 현상을 방지하기 위해서, 윈도우즈 운영체제의 DirectX 에서는 수직 공백 구간(Vertical Blank Interval)에 동기화 할 수 있는 메소드를 제공하고 있다^[5]. 이를 잘 이용하는 경우, 프레임 버퍼의 갱신이 수직 공백 구간에서 이루어지게 하여, 모니터의 주사 주기에 동기화를 이룰 수 있으며, 따라서 tearing 현상을 방지 할 수 있다. 그러나, 디스플레이 시스템의 구조가 잘 설계되지 않으면, 수직 공백 구간의 동기화를 위한 기다림으로 인해 최대 디스플레이 속도가 모니터 주사 주기에 한정되어, 다중 채널 디스플레이를 요구하는 DVR 에서는 요구되는 디스플레이 속도 성능을 만족할 수 없게 된다.

본 논문에서는 윈도우즈 운영체제 기반 DVR의 경우에, 디스플레이 속도 성능을 별로 저하시키지 않으면서, tearing 현상을 방지한, 개선된 DVR 디스플레이 시스템 구조를 제안한다. 본 논문의 구성은 다음과 같다. 제2절에서 윈도우즈 운영체제 기반 DVR 시스템의 디스플레이 구조를 기술하고, DVR 시스템에서 더 심화되는 tearing 현상의 이유를 분석한다. 제3절에서는 이러한 분석을 통하여, 추가적인 임시 오버레이 표면(surface)를 추가로 설계하고, 이 곳에 다채널 영상을 갱신하고, 이를 프레임 버퍼에 아날로그 모니터 리플레쉬 주기에 맞게 갱신하

도록 하여 tearing 현상을 개선한 결과를 설명한다. 제안된 디스플레이 구조의 효용성은 실험을 통해 검증하며, 이를 제4절에 기술한다. 마지막으로 제 5 절에서 결론이 기술된다.

II. DirectDraw, 영상 감시 시스템(DVR) 디스플레이 및 tearing 현상

이 절에서는 먼저 윈도우즈의 디스플레이 시스템의 주요한 요소인 DirectDraw 와 이에 관련하여 본 논문을 이해하는 데 필요한 주요 개념이 기술된다. 이후, 최근 영상 감시 시스템으로 각광을 받고 있는 DVR에 대해 간단히 소개가 주어지며, 기존 윈도우즈 기반 DVR 의 디스플레이 시스템 구조가 설명된다. 그리고, 아날로그 모니터를 사용하는 경우 나타날 수 있는 tearing의 현상 발생 원인이 설명된다. 또한, 윈도우즈 기반 디스플레이 구조에서의 tearing 현상 방지 방법인 수직 공백 구간 동기화 기법과, 이 방법을 사용한 기존 DVR의 디스플레이 구조에서의 디스플레이 속도 성능 저하 문제가 설명된다.

2.1 DirectDraw

윈도우즈 운영체제 환경에서는 영상의 빠른 처리를 위해 프레임 버퍼로의 직접적인 작업을 지원하는 DirectDraw를 제공하고 있다. DirectDraw 는 GDI(Graphic Device Interface)가 갖는 상위레벨 드로잉 함수(문자 출력, 2D 그리기 함수 등)를 제공하지 못하지만, 응용이 직접 디스플레이 메모리에 접근할 수 있도록 허락하며 GDI 에서는 가능하지 않은, 그래픽 하드웨어의 고급 특징(하드웨어 브리팅(blitting), 스트레칭, 오버레이 등)들을 이용할 수 있도록 지원한다. DirectDraw 는 DirectX API 세트의 일부이며, 모든 윈도우 운영체제에서 지원된다. 현재 버전은 8.1이다.

DirectDraw을 이용하여 구현한 본 논문의 디스플레이 시스템의 이해를 위해 디스플레이 및 Direct Draw 관련용어를 여기에 간단히 정리한다^[5].

- 1) 스캐닝(scanning) : 그래픽 어댑터의 RAMDAC 은 프레임 버퍼(비디오 메모리)에 있는 영상 데이터를 읽어 아날로그 신호로 변환하여 이를 아날로그 모니터에 전송한다. CRT 모니터의 경우, 이 아날로그 신호에 따라 CRT의 전자총에서 전자빔 (Cathode Ray)의 강도 및 방

향을 조정하고, 이에 따라 전자빔이 스크린에 코팅된 형광물질을 발광시켜 컬러가 보이도록 한다. 그래픽 어댑터는 프레임 버퍼의 영상 데이터를 왼쪽 상단에서부터 오른쪽 하단까지 한줄 단위로 읽게 되는 데, 이를 래스터 주사 (raster scan) 방식이라 한다.

- 2) VBI(Vertical Blank Interval) : 래스터 스캐닝 할 때, 주사선이 오른쪽 하단에 도달했을 때에 이로부터 다시 왼쪽 상단까지 돌아가는 데 걸리는 시간 간격
- 3) Tearing artifacts : 프레임 버퍼의 영상정보 갱신 시간과 아날로그 모니터 화면에서의 영상 갱신 시간의 불일치가 발생하여 영상 화면에 선이 그어지거나, 한 프레임에서 영상의 상하가 불일치하게 되는 현상을 말한다.
- 4) DirectDraw Surface : 디스플레이 메모리의 선형 메모리 영역을 말하며, 시스템 메모리나 비디오 메모리에 존재한다.
- 5) Blit : Bit block transfer의 약자로서, 메모리의 한 주소 공간에서 다른 주소공간으로 데이터 블록을 전송하는 오퍼레이션을 말한다. 보통 영상 블록 데이터를 이동할 때 사용되는 오퍼레이션이다. DirectDraw 가 지원하는 블리팅에는 Blt 와 BltFast 가 있으며, 후자는 소스 표면과 목적지 표면의 크기가 같을 때, 빠른 속도로 블리팅되도록 지원되는 오퍼레이션이다.
- 6) Primary Surface : 모니터에 디스플레이되는 표면을 말하며, 그래픽 어댑터는 프라이머리 표면의 영상 데이터를 스캐닝하여 모니터에 디스플레이한다.
- 7) Paging flipping : 연속적인 표면들의 체인을 구성하여 디스플레이하는 방법으로, 처음 표면은 front surface, 그 뒤의 표면들은 back surface 들이라 하며, 플립시에 전방 표면뒤의 후방 표면이 전방 표면으로 바뀌게 된다. 따라서, 전방 표면이 디스플레이되는 동안 후방 표면에 디스플레이될 영상을 준비시킨 후에, 전방 표면의 디스플레이가 끝나면 플립하여 후방 표면이 디스플레이 되도록 할수있어 빠른 영상 디스플레이 처리가 가능해진다. 또한 DirectDraw 에서는 플립시에 수직 공백 구간 (VBI; Vertical Blank Interval) 동기를 달성하도록 지원하고 있다. VBI 동기는 주사선이 VBI에 주사하기 시작하는 시점을 기다려서,

이때부터 주사선이 좌측 상단에 갈 때까지의 시간에 영상정보를 갱신하는 것을 말하며, 이 VBI 동기는 tearing 현상을 해결하는 중요한 해법이다⁴⁾.

- 8) Overlay surface : 오버레이 표면은 프라이머리 표면에 오버레이하는 데 사용된다. 오버레이가 셋팅되는 경우, 프라이머리 표면위에 오버레이 될 사각형 영역이 정의된다. 그래픽 어댑터가 모니터에 디스플레이하기 위해 프라이머리 표면에 있는 영상 데이터를 스캐닝할 때, 오버레이로 정의된 사각형 영역에 도달하게 되면, 영상 데이터를 프라이머리 표면에 있는 영상 데이터가 아니라 오버레이 표면에 있는 영상 데이터를 스캐닝한다.
DirectDraw는 오버레이 표면 지원뿐만 아니라 프라이머리 표면에 어떻게 오버레이 할 것인가도 지원한다. 예를 들어, 컬러 키를 이용하면, 프라이머리 표면 중 원하는 색상에만 오버레이 할 수 있다. 이러한 오버레이 기능은 그래픽 하드웨어 레벨에서 제공하기 때문에 성능이 저하되지 않고 수행되며, DirectDraw 는 그래픽 하드웨어가 제공하는 이러한 능력들을 지원한다.
- 9) Off-screen Surface : 디스플레이되기전의 영상 데이터를 가지고 있는 표면으로, 오프 스크린의 영상 데이터는 후에 오버레이 표면이나 프라이머리 표면으로 블리팅 된다.

2.2 영상 감시 시스템(DVR)

DVR(Digital Video Recorder)은 종래의 아날로그 기반 시스템과 달리 CCTV 카메라에서 획득한 아날로그 영상을 디지털 영상으로 변환하고 이를 컴퓨터 시스템으로 받아들여 모니터에 디스플레이하고 원격감시를 위해서 전송하며, 추후 검색을 위해 저장하는 새로운 기술의 영상 감시 시스템이다. DVR에서는 영상을 디지털화하기 때문에, 이미 잘 발전된 디지털 신호처리 기술을 사용할 수 있게 되어 다양한 영상 처리가 가능해지고, 더 많은 카메라 채널을 화면에 디스플레이할 수 있으며, 압축기법을 이용하여 정해진 저장 공간에 보다 많은 영상 데이터를 저장할 수 있게 되므로, 최근에 들어와 매우 각광을 받고 있는 영상 감시 시스템이다^{1,2)}.

현재 출시되고 있는 DVR은 PC 기반으로 설계 제작된 제품과 독립적인 임베디드 시스템으로 설계 제작된 제품으로 대별된다. PC 기반 제품의 경우,

운영체제로 윈도우즈(윈도우즈 95, 98, me, NT, 2000, 2001 등)를 사용하는 경우와 리눅스를 사용하는 경우로 대별되며, 임베디드 DVR은 임베디드 리눅스, 실시간 운영체제 등을 사용한다.

다음 그림 1에는 DVR을 이용한 영상 감시 시스템의 예를 나타낸다.

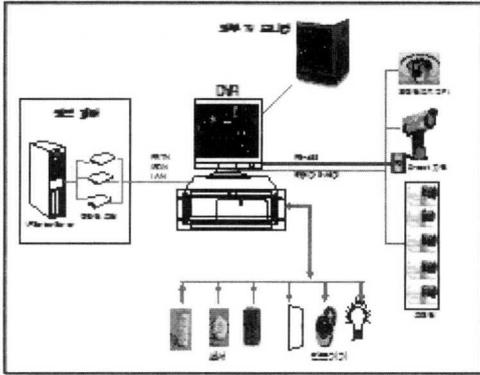


그림 1. DVR을 이용한 영상 감시 시스템의 예.

현재 DVR에 요구되는 주요 기능들은 다음과 같다[2].

- 1) 다중 카메라 채널(최대 16채널) 동시 실시간 디스플레이 지원
- 2) 16 채널 영상 데이터 실시간 저장
- 3) 다양한 화면 분할 지원
- 4) 저장된 영상 검색 지원
- 5) 카메라 Pan/Tilt/Zoom 제어 및 디지털 입출력(8개의 입력과 8개의 출력) 지원
- 6) 원격 모니터링 및 원격 검색 지원

2.3 윈도우즈 기반 DVR의 디스플레이 시스템 구조

DVR은 TV 수신 시스템이나 기타 다른 영상 시스템과는 달리 다중 채널 디스플레이를 지원해야 하는 요구 조건을 가지고 있다. 또한 디스플레이 되는 영상을 1채널 Full Screen, 4, 9, 10, 13, 16 채널 화면등의 다양한 모드로 디스플레이해야 하며 또한 추가적인 각종 시스템 정보 등을 디스플레이 되는 화면에 영상과 같이 나타내야만 한다.

DVR에 있어서의 디스플레이 방식은 크게 두 가지로 나뉘어 질 수 있다. 하나는 디스플레이를 위한 전용 영상 오버레이 하드웨어를 사용하는 경우이며, 다른 하나는 영상 캡처 하드웨어를 통해 획득한 영

상을 디스플레이 하는 경우이다. 전자의 경우, 영상 캡처 하드웨어로는 영상을 획득하여 압축 저장하는 데 사용하며, 디스플레이는 오버레이 하드웨어를 이용하게 되는데, 오버레이 하드웨어에서는 획득된 영상을 시스템에서 원하는 화면모드에 따라 한 프레임으로 구성하여 이를 프레임 버퍼에 써서 화면에 디스플레이 한다. 이 경우, 다채널 영상들이 한 프레임 화면으로 조립되고, 조립된 프레임이 30 frames/sec 로 디스플레이 된다. 이 경우, 디스플레이 속도는 매우 빠르나, 오버레이 하드웨어 추가 설계작으로 비용이 더 소요된다. 반면, 영상 캡처 카드만을 사용하여 획득된 영상을 디스플레이하고, 압축저장하는 방식의 경우에 성능은 다소 떨어지나, 오버레이 하드웨어의 추가 설계가 필요없으므로 제작 비용이 적게 들어 현재 많이 선호하는 편이다. 그런데, 이 경우에 있어 캡처 하드웨어를 통해 획득한 동영상 데이터를 디스플레이하고자 하는 모드별로 축소 및 확대를 할 수 있어야 하며, 각각의 채널 영상들이 모니터 화면의 원하는 위치에 나타나도록 적절히 배치를 해주어야 하는 등 여러 가지 고려할 점이 많다.

윈도우즈 운영체제 환경에서는 동영상의 빠른 처리를 위해 직접 프레임 버퍼에로의 작업 및 다양한 오퍼레이션 연산의 가속을 지원하는 DirectX를 제공하고 있다. 최근의 대부분의 비디오 어댑터들이 DirectX 또는 OpenGL을 하드웨어적으로 가속을 하게끔 설계가 되어 있으며, 오버레이 기능을 지원하고 있다. 따라서, 실시간 동영상 처리가 요구되는 DVR에서는 이러한 하드웨어 가속 및 오버레이 기능을 지원하는 비디오 어댑터를 채용하며, 이를 이용한 윈도우즈 기반 DVR의 기존 디스플레이 시스템은 보통 다음과 같은 구조를 갖는다[2].

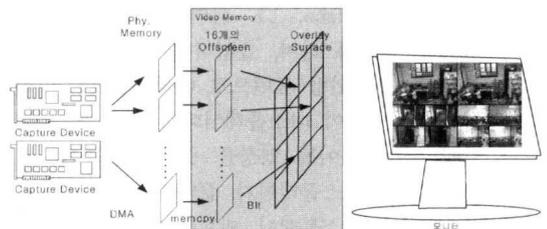


그림 2. 기존 윈도우 기반 DVR의 디스플레이 시스템 구조

캡처 장치(capture device)에서 획득한 각 카메라 채널의 영상 데이터가 물리적 메인 메모리의 버퍼에 저장되면, 디스플레이를 위해 먼저 이를 각 채널

별로 프레임버퍼(비디오 메모리)상의 오프스크린 표면(offscreen surface)에 복사한다(memcopy). 각각의 오프스크린 표면 영상은 사용자 요구 디스플레이 모드에 맞게 확대·축소되고 최종 모니터 화면 구성에 맞게 오버레이 표면에 복사(Blt)한다. 이후 오버레이 표면의 영상정보는 비디오 어댑터 콘트롤러에서 자동적으로 모니터 화면에 출력하도록 해준다.

오프스크린 표면은 메인 메모리에서도 만들 수 있는데, 비디오 메모리에 만드는 이유는 비디오 어댑터의 하드웨어 가속 및 하드웨어 오버레이 기능을 최대한 이용하기 위해서이다. 또한 오프스크린 표면을 사용하는 이유는 캡처 장치에서 가져온 영상 데이터를 프레임 버퍼에 직접 복사하게 되면 영상이 깜박거리는 현상이 있을 수 있어 이러한 현상을 최소화 시켜주기 위한 것이며, 또한 영상 사이즈 조절 및 16채널 조합을 효과적으로 하기 위해서이다. 오버레이 표면 사용은 프라이머리 표면(primary surface)을 직접적으로 사용하는 것에 비해 특정영역의 확대 및 축소가 용이하며, 오버레이시 특정 커리커를 이용하는 등의 다양한 기법을 이용하여 매 프레임 갱신될 필요가 없는 문자정보(예, 카메라 번호 등)를 처리하기가 편리하기 때문이다.

다른 운영체제하의 디스플레이 시스템에서도 이상의 디스플레이 구조와 비슷한 구조를 갖지만 Offscreen Surface의 위치는 비디오 메모리 상에 위치하거나 시스템에 따라 시스템의 메인 메모리 상에 위치 할 수 있다.

2.4 Tearing 현상

영상을 모니터 화면에 디스플레이 하기 위해서는 화면에 디스플레이 하고자 하는 정보가 비디오 어댑터의 프레임 버퍼로 전달이 되어야하고 비디오 어댑터의 RAMDAC은 프레임 버퍼(비디오 메모리)에 있는 디지털 영상 정보를 아날로그 신호로 변환하여 이를 아날로그 모니터에 전송한다. CRT 모니터의 경우, 이 아날로그 신호에 따라 CRT의 전자총에서전자빔(Cathode Ray)의 강도 및 방향을 조정하고, 이에 따라 전자빔이 스크린에 코팅된 형광물질을 발광시켜 컬러가 보이도록 한다. 아날로그 모니터의 디스플레이 갱신은, 전자빔이 왼쪽 상단에서부터 오른쪽 하단까지 한 줄 단위로 스캔되는 래스터 주사(raster scan) 방식으로 이루어진다. 따라서, 프레임 버퍼에 갱신된 영상정보는 즉시 모니터 화면에 갱신되는 것이 아니고, 리플레쉬 주기에 따라 갱신된다. 이 과정에서 프레임 버퍼의 영상정보 갱

신 시간과 아날로그 모니터 화면에서의 영상 갱신 시간의 불일치가 발생하게 되며, 이에 따라 화면에 선이 가거나 영상의 상하가 불일치하는 tearing 현상(artifact)이 발생하게 된다. 입력되는 동영상이 충분히 느린 경우에는, 현 입력 영상과 바로 이전 영상과의 차이가 별로 없기 때문에 tearing 현상은 잘 나타나지 않지만, 입력 되는 동영상이 매우 빠르게 변화하며 영상간의 차이가 클수록 tearing 현상이 매우 심하게 나타나게 된다. (그림 5-a 참조).

다음의 설명을 통해 이러한 tearing 현상이 발생하는 원인을 좀더 명확히 살펴 보자^[4].

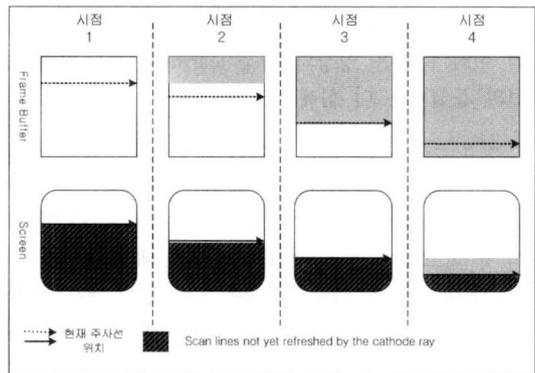


그림 3. Tearing 현상 발생의 원인 규명

그림 3에서 상단은 프레임 버퍼 상태를, 하단은 프레임 버퍼 하단을 나타낸다. 프레임 버퍼의 모든 pixel들은 처음에 흰색으로 값이 설정되어 있다고 하자. 또한, 프레임 버퍼의 갱신 주기가 스크린 주사 주기 보다 빠르다고 하자.

시점 1은 응용프로그램에서 모든 픽셀을 회색으로 변경하라고 명령을 내리고, 아날로그 모니터 스크린에서 현재 주사되는 주사선은 화면의 1/3 부분까지 와 있을 때의 프레임 버퍼와 스크린의 상태를 나타낸다. 프레임 버퍼의 점선 화살표와 스크린의 실선 화살표는 주사선의 위치를 나타낸다.

시점 2은 프레임 버퍼의 상단 1/3안에 있는 픽셀들이 회색으로 갱신되고, 스크린에서 현재 주사되는 주사선은 시점 1에 비해 몇 라인 아래로 내려와 있는 상태이다.

시점 3에서는 프레임 버퍼의 2/3안에 있는 픽셀들이 회색으로 갱신되고, 현재 주사되는 주사선도 프레임 버퍼의 갱신 부분까지 서로 일치한 상태이다. 하지만 이 시점까지는 화면에는 여전히 이전 프레임 버퍼의 영상인 흰 영상만이 나타나게 된다. 왜냐하

면 회색으로 채워진 부분은 아직 스크린 화면에서 주사에 의해 한번도 갱신되지 않았기 때문이다.

시점 3과 4 사이에 프레임 버퍼는 화면 주사에 앞서 갱신되어 시점 4에서 프레임 버퍼는 이제 전체가 회색으로 채워졌고, 이때 스크린에서는 시점 3에서부터 시점 4까지 주사된 영역을 보여주게 된다. 따라서 사용자는 실제로는 화면 전체가 회색으로 갱신된 영상을 보는 것이 아니라, 화면의 2/3 선을 경계로 전 영상(회색)과 갱신된 영상(회색)이 분리되어 나타나는 tearing 현상의 한 예를 보게 된다.

실제로는 이러한 과정은 고속으로 이루어지게 되고 앞의 예에서처럼 단순히 특정색만이 아닌 복잡한 영상을 보이게 된다. 이러한 경우에 이전영상과의 변화가 많은 영상의 경우 tearing 현상에 의한 화면의 불일치는 더 심해진다.

2.5 기존 윈도우즈 운영체제 기반 DVR의 디스플레이 구조에서의 tearing 현상

2.3 절에서 설명한 윈도우즈 기반 DVR 디스플레이 구조(그림 2 참조)에서는 실제 프레임 버퍼에 영상 데이터가 갱신되는 비율은 각 캡처 장치로부터 영상이 들어온 프레임 비율만큼이 된다고 할 수 있다. 프레임 버퍼에 있는 오버레이 표면의 영상 정보 갱신은 각 오프스크린 표면의 갱신된 영상 정보가 오버레이 표면에 복사됨으로써 발생한다. 따라서, m 채널 디스플레이의 경우, 채널당 초당 n 프레임의 디스플레이가 요구된다면, 총 m*n 번, 오버레이 표면이 갱신된다. 보통 한 채널 풀모션 비디오가 초당 30 프레임의 프레임 버퍼의 변경이 이루어지는데 반해, 보통 16 채널 지원 DVR의 경우, 초당 240프레임의 디스플레이를 지원하여야 하기 때문에, 프레임 버퍼 갱신 비율은 스크린의 주사선 갱신 주기(보통 초당 50 ~ 120)에 비해, 훨씬 높기 때문에 빨리 변하는 동영상 디스플레이의 경우, tearing 현상이 심하게 나타나며, 따라서, 한 채널만을 사용하는 영상시스템이나 오버레이 하드웨어를 사용한 디스플레이 방식에 비해 그 현상 해결이 다소 어렵다는 것을 알 수 있다.

2.6 DVR의 기존 디스플레이 구조에서의 Tearing 현상해결의 문제점

Tearing 현상을 막는 방법의 하나는 프레임 버퍼로의 갱신을 스크린 화면의 주사선이 좌측 상단으로 가는 동안, 즉 주사선이 수직 공백 구간(vertical blank interval)에 주사하기 시작하는 시점을 기다려서, 이때부터 주사선이 좌측 상단에 갈 때까지의 시

간에 영상정보를 프레임 버퍼에 갱신하여, 스크린 주사 주기와 프레임 갱신 주기를 동기화시킴으로써, tearing 현상을 제거하는 것이다. 이를 위하여 DirectX에서는 수직 공백 구간에 동기화 할 수 있는 기능을 제공하고 있다. DirectX에서는 "IDirectDraw7::WaitForVerticalBlank", "IDirectDrawSurface7::Flip"와 같은 수직 공백 구간에 동기화를 달성할 수 있는 메소드를 제공한다^[5]. 이러한 동기화 기능은 한 채널의 풀모션 디스플레이(초당 30프레임)의 경우, 성능의 저하없이 성공적으로 적용될 수 있다.

그러나, DVR의 기존 디스플레이 구조(그림 2)에서 수직 공백 구간 동기화 기법을 이용하게 되면(예를 들어, 오프스크린 표면에서 Blt 하기전에 'WaitForVerticalBlank' 메소드를 호출하여, 수직 공백 구간과 동기화를 달성), tearing 현상은 제거 되겠지만, 스크린 주사 주기와 동기화를 위한 기다림으로 인해 오프스크린 표면에 갱신된 영상이 오버레이 표면에 갱신되지 못하고 프레임에 Drop이 발생하여 디스플레이 성능의 저하가 발생한다.

III. 제안된 윈도우즈 기반 DVR의 디스플레이 시스템 구조

앞서 설명하였듯이, DVR의 기존 디스플레이 시스템 구조에서 윈도우즈 DirectX에서 제공하는 수직 공백 구간 동기화 기법을 적용하는 경우, tearing 현상은 해결되지만, 화면에서 영상이 갱신 될 수 있는 비율은 아날로그 모니터의 주사 주기에 의해 제한되어 디스플레이 속도 성능의 저하가 초래된다(표 1 참조).

본 논문에서 tearing 현상도 해결하면서, 디스플레이 성능 저하를 초래하지 않는 윈도우즈 기반 DVR의 디스플레이 시스템 구현을 위하여 다음과 같이 디스플레이 구조를 설계하였다.

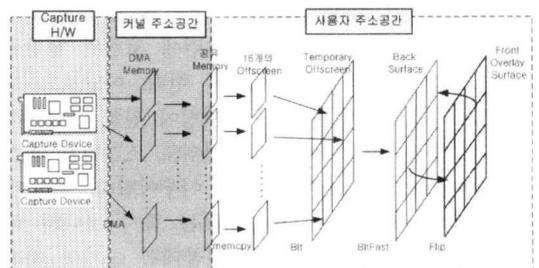


그림 4. 개선된 디스플레이 구조

제안된 DVR의 디스플레이 구조가 기존의 디스플레이 구조(그림 2 참조)와 다른 부분은 임시 오프스크린 표면과 후방(back) 오버레이 표면의 추가 설계이다. 그림 4의 종단에서 보이는 후방 오버레이 표면과 전방(front) 오버레이 표면 간의 플립(flip)은 두 표면간 전환(switching)을 의미한다. 즉, 플립 수행시에 후방 오버레이 표면은 전방 오버레이 표면으로 전환되고 이 전환 시기는 수직 공백 구간에 동기화된다⁵⁾. 이 후, 전방 오버레이 표면의 내용이 화면 주사 주기에 따라 화면에 갱신되게 된다. 또한, 임시 오프스크린 표면과 후방 오버레이 표면 간의 복사는 BltFast 라는 빠른 표면간 복사를 지원하는 DirectX 메소드를 사용하였다. BltFast는 scaling을 지원하지 않는 반면 Blt에 비해 약 10%정도 성능향상을 주는 것으로 알려져 있다⁵⁾. 플립(flip) 메소드 호출 주기는 스크린 주사 주기를 고려하여 선정하였다. 플립시에 오버레이 표면의 전환은 수직 공백 구간에 동기화되므로, 플립 메소드 호출 주기는 스크린 주사 주기보다 빠를 필요가 없으므로, 프레임 버퍼에 각 카메라 채널의 영상의 갱신이 충분히 반영되도록만 플립 메소드 호출 주기를 결정하도록 하는게, 시스템 자원의 낭비를 막을 수 있어 바람직하다.

그림 2의 기존 구조에서는 수직 공백 구간에 모든 다중 채널 영상의 갱신이 가능하도록 하기가 어려웠으나, 그림 4의 제안된 구조에서는 오프스크린 표면과 오버레이 표면 사이에 임시 오프스크린 표면을 추가로 설계하여, 일정한 순서없이 갱신되는 다 채널 입력 영상에 대해 버퍼 역할을 수행하여 항상 화면에 나타날 완성된 다채널 영상을 먼저 구성한다. 그리고, 플립전에 'BltFast' 메소드 호출을 통해 후방 오버레이 표면에 이를 복사함으로써, 후방 오버레이 표면에는 화면에 디스플레이 될 완성된 다중 채널 화면이 미리 준비되어 있게 된다. 플립시에 이 후방 오버레이 표면은 전방 오버레이 표면으로 전환되고 이 전방 오버레이 표면의 영상 내용이 화면에 주사된다. 플립은 수직 공백 구간과 동기화가 이루어지므로, tearing 현상의 발생은 방지된다.

플립 메소드 호출 주기는 25 ms 로 셋팅 하였다. 디스플레이를 수행하는 스레드의 동작이 다른 스레드의 동작에 영향을 받기 때문에, 호출주기의 지터가 발생하게 되며, 실험적으로는 플립 메소드 호출이 25 ~ 30ms 마다 이루어지고 있음이 관찰되었다. 이런 시간 간격 (25 ~ 30ms)의 경우, 임시 오프스크린 표면에 각 카메라 채널 영상이 평균 1회 이상

갱신될 수 있으므로, 결국 화면에는 각 카메라 채널 영상이 충분한 속도로 디스플레이 되게 된다.

IV. 실험 및 검토

제안된 디스플레이 구조에서 tearing 현상이 제거되고, 또한 디스플레이 성능의 저하가 별로 없는 것을 보기 위해, DVR에 제안된 디스플레이 구조(그림 4)를 설계 구현하여, 기존 디스플레이 구조(그림 2)를 구현하는 DVR의 디스플레이와 동일한 컴퓨터 환경에서 비교하였다.

실험 환경은 다음과 같다.

1) 컴퓨터 H/W 환경

- CPU : Intel Pentium IV 1.5GHz
- Memory : RDRAM PC800 256M
- Video Adapter : ATI Radeon 7200 (DirectX 7.0a 하드웨어 가속 지원)

2) 컴퓨터 S/W 환경

- 운영체제 : Microsoft Windows2000 (Service Pack 2)
- DirectX : DirectX 8.1

3) 모니터 수직 주파수

- 모니터의 수직 주파수는 스크린 주사 주기와 같으며, 75 Hz 로 셋팅하였다.

4) 영상 캡처

- 제작된 캡처 카드 2개 이용
- 한 캡처 카드에는 4개의 비디오 디코더(Bt878) 칩을 가지고 있으며, 한 개의 비디오 디코더 칩은 2개까지의 카메라 채널을 다중화 한다. 총 2개의 캡처 카드를 사용하였으므로, 최대 16 카메라 채널에 캡처를 지원한다.

4.1 Tearing 현상 발생 비교

첫 번째 실험에서는 디스플레이 성능의 사양을 만족하는 기존 디스플레이 구조를 구현 한 DVR과 제안된 디스플레이 구조를 구현한 DVR 시스템에 대해, 16채널 디스플레이 동작 중에, 각각 1 개의 카메라 채널에 대해 카메라 앞에서 빠르게 손을 흔들어 tearing 현상이 발생하는 가의 유무를 비교하였다. 그림 5-a에서 보는 바와 같이 기존의 구조에서는 쉽게 tearing 현상을 관찰할 수 있었다. 같은 실험 환경 및 같은 실험 방법에서 제안된 디스플레이 구조를 구현한 DVR 시스템의 경우에는 tearing 현상을 볼 수 없었다(그림 5-b 참조).



그림 5-a. 기존 디스플레이 구조를 갖는 DVR 에서의 tearing 현상

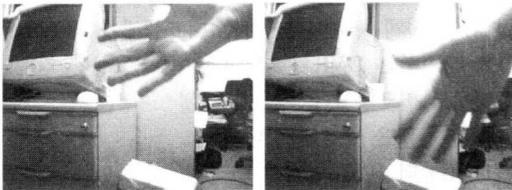


그림 5-b. 제안된 디스플레이 구조를 갖는 DVR 에서의 디스플레이

4.2 디스플레이 성능 비교

두 번째 실험에서는 기존 디스플레이 구조에서 tearing 현상을 방지하기 위해 수직 공백 구간 동기화 기법을 채용한 경우와 하지 않는 경우, 그리고 제안된 디스플레이 구조에서의 디스플레이 속도를 동일한 실험 조건에서 1분 정도 관찰하여 비교하여 보았다. 표 1은 이 실험 결과를 정리한 것이며, 각 경우의 값은 초당 디스플레이 되는 총 프레임수를 나타낸다.

표 1. 디스플레이 속도 성능의 비교

채널수		1 채널	4 채널	8 채널	16 채널
		경우			
기존 구조	VBI 동기화 미적용	28-30	115-120	235-240	160
	채널별 동기화 VBI 적용	28-30	45-75	50-65	55-60
제안된 구조		28-30	110-120	232-239	155-160

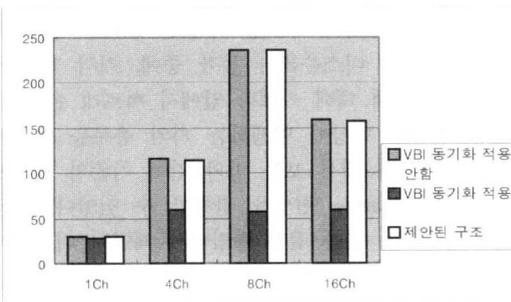


그림 6. 디스플레이 속도 성능 비교

결과에서 보면, 기존 디스플레이 시스템 구조(그림 2)에서는 수직 공백 구간 동기화를 사용하는 경우, 디스플레이 속도 성능의 저하가 심함을 알 수 있다. 모니터 수직 주파수가 75 Hz 임에도, 디스플레이 속도 성능이 초당 75 프레임에 못 미치는 것은 “WaitForVerticalBlank” 메소드 호출시에 수직 공백 구간과의 동기화를 위해 대기함으로 인해 8,16채널과 같이 초당 획득한 영상의 프레임 수가 많을수록 “WaitForVerticalBlank”의 호출이 더 많아지고 따라서 동기화를 위한 대기가 더 많아져 프레임의 Drop이 더 많이 발생하기 때문이다.

제안된 디스플레이 구조를 채용한 경우, 디스플레이 성능은 수직 공백 구간 동기화를 적용하지 않은 기존 디스플레이 구조에서의 성능에 비해 별로 저하가 없음을 알 수 있다.

V. 결론

본 논문에서는 최근 영상 감시 시스템으로 각광을 받고 있는 DVR 시스템에서 아날로그 모니터를 사용하는 경우, 디스플레이에서 심각히 발생하는 tearing 현상의 원인을 규명하고, 윈도우즈 기반 DVR에서 디스플레이 속도 성능의 저하를 별로 가져오지 않으면서, tearing 현상 발생을 방지한 새로운 디스플레이 시스템 구조를 제안하였다. 제안된 디스플레이 시스템 구조의 효율성은 윈도우즈 기반 DVR에서의 구현을 통해 실험하고 입증하였다.

본 논문에서 제안된 디스플레이 성능 열화가 없는 tearing 현상 방지 디스플레이 시스템 구조의 아이디어는 윈도우즈 운영체제 기반이 아닌 다른 DVR 시스템에서도 적용될 수 있다. 리눅스 운영체제 기반 영상 감시 시스템에서도 같은 아이디어를 적용하여 디스플레이 성능은 저하시키지 않으면서 tearing 현상은 개선함을 확인하였다.

리눅스 운영체제를 사용하는 경우 윈도우즈 DirectX 와 같이 Frame Buffer 장치 드라이버 인터페이스를 통해 직접 그래픽 어댑터의 하드웨어 능력을 제어하는 디스플레이 시스템인 DirectFB 가 개발되어 있다^[6]. 본 논문의 아이디어를 DirectFB를 사용한 리눅스 기반 DVR 시스템에서 적용하여 윈도우즈 경우와 마찬가지로 디스플레이 성능은 별로 저하시키지 않으면서 tearing 현상을 개선할 수 있음을 확인하였다(자세한 내용은 [7] 참조).

참 고 문 헌

- [1] DVR(Digital Video Recorder) 시스템 개요,
<http://www.digitalplaza.co.kr/library/cctv-dvr.htm>,
웹 문서
- [2] 이동학, “영상 감시 시스템의 효율적인 멀티미디어 스트리밍 처리 부시스템 설계 및 구현”, *숭실대학교 전자공학과 석사학위 논문*, 숭실대학교, 2001
- [3] Caroline M. Halliday, *PC Secretes*, 2nd ed., IDG, 1996.
- [4] Damon Chandler, Michael Fötsch, *Windows2000 Graphic API Black Book*, Coriolis, 2001.
- [5] DirectX7 Platform SDK , <http://download.microsoft.com/download/win98SE/DXSDK/7.0a/W9X/EN-US/dx7adxf.exe>
- [6] DirectFB reatures and concepts overview V0.1,
http://www.directfb.org/documentation/DirectFB_overview_V0.1.pdf
- [7] 정연권, 정선태, “리눅스 기반 다채널 영상 디스플레이 시스템에서의 Tearing 현상 개선”, *2002 신호처리 합동학술 대회 논문*, 2002. 9.

정 선 태(Sun-Tea Chung)



1983년 : 서울대학교 전자공학과, 공학사

1986년 : The Univ of Michigan, 전기 공학 및 전산학과, 공학석사

1990년 : The Univ. of Michigan, Ann Arbor

1991년 3월~현재 : 숭실대학교 정보통신전자공학부 교수

1999년 9월~2000년 6월 : The Univ. of Massachussts, Amherst 전산과 방문교수

<주관심 분야> 실시간 시스템, 임베디드 시스템, 멀티미디어 시스템

정 연 권(Yeon-Gwon Jeong)



2001년 2월 : 숭실대학교 정보통신전자공학부 졸업

2001년 3월~현재 : 숭실대학교 정보통신공학과 석사 과정

이 동 학(Dong-Hak-Gwon Lee)

2000년 2월 : 숭실대학교 전자공학과 졸업

2002년 2월 : 숭실대학교 전자공학과 석사

2002년 2월~현재 : (주)유스테크놀로지 연구소
선임연구원