
다이렉트쇼 기반의 필터차단기술 설계

Design of Filter Cut-off Technology based on Directshow

김대진, 최홍섭
대전대학교 전자공학과

Dae-Jin Kim(sampoo00@hanmail.net), Hong-Sub Choi(hschoi@daejin.ac.kr)

요약

멀티미디어가 일반화됨에 따라 이 콘텐츠 이용을 위한 수많은 소프트웨어가 개발되었다. 다이렉트쇼(Directshow)는 보다 쉽고, 빠르면서, 범용성 있는 멀티미디어 소프트웨어 개발을 위한 미디어 프레임워크(Framework)이다. 다이렉트쇼에서는 필터(Filter)간 연결을 통해 데이터 처리를 할 수 있다. 일반적으로 필터 간 자동연결기법을 이용하여 필터그래프(Filtergraph)를 구성하였다. 그러나 필터 간 자동연결시 원하지 않는 필터들이 연결되어 예상치 못한 결과가 출력되거나 필터 핀(Pin)간 연결에 실패할 수 있다. 따라서 본 논문에서는 원하는 필터들만 연결하기 위하여 불필요한 필터들을 차단하는 방법에 대하여 제안한다. 차단방법에는 필터 내부 차단과 필터 외부 차단이 있으며, 내부 차단은 연결 호환성이 좋고 필터연결시간이 빠르고 외부차단은 필터 소스가 불필요하며 연결의 정확도가 높다. 결론적으로 이들 필터 차단 방법의 장점을 조합하여 원하는 필터 간 연결을 구성하고, 오류를 최소화 할 수 있는 다이렉트쇼 기반의 필터차단기법 모델을 제안한다.

■ 중심어 : | 다이렉트쇼 | 필터 | 필터차단 |

Abstract

As the multimedia technology has become in popular, many software packages that are for using those multimedia contents have also been developed. Directshow is a media framework which makes the development of such multimedia software simple, quick and regular process. In the Directshow, data are processed through the connections between filters. In general, Filtergraph is utilized to be set as 'Intelligent Connection' for filter connection. However, when we use this 'Intelligent Connection' option, they can show non-expected results or fail to connect with filter's pins together. So in this paper, we propose non-necessary filter cut-off method for connecting only just wanted filters. There are two methods such as inner filter cut-off method and outer filter one. The inner filter cut-off method is good at connection's compatibility and able to quickly connect with each other filter. But in the outer filter cut-off method, filter source is not necessary and connection is exactly done with other filter. In the end, we propose Directshow based filter cut-off model which compose of wanted filter connection and minimize error probability, through mixing up the advantages of two filter cut-off methods.

■ keyword : | Directshow | Filter|Filter Cut-off |

I. 서론

광대역 네트워크의 발달과 함께 멀티미디어 산업의 발달은 디지털 콘텐츠 시장의 확산과 함께 이 콘텐츠 이용을 위한 수많은 소프트웨어가 개발되었다. 그러나 소프트웨어 개발을 위해 많은 시간이 소요되고 복잡하여 어려움 있었다. 특히 복잡한 데이터 전송방식, 하드웨어 특성, 데이터 동기화(Synchronization) 등 많은 문제를 드러내고 있다.

이러한 배경 속에서 보다 쉽고, 빠르면서, 편리하게 멀티미디어 소프트웨어를 개발할 수 있는 개발 환경이 필요해졌다. 따라서 마이크로소프트사에서는 ‘다이렉트쇼(DirectShow)’라는 멀티미디어 프레임워크를 제공한다. 다이렉트쇼는 고품질의 멀티미디어 스트림을 재생 가능하고 캡처(Capture) 및 편집까지 가능하도록 구조화 되어있다[1][2]. 이러한 특성을 이용하여 VOD(Video On Demand), IPTV, 트랜스코딩(Transcoding)과 같은 응용 솔루션 개발 분야에 많이 이용할 수 있다[3-5]. 또한 ASF(Advanced Streaming Format), MPEG(Motion Picture Experts Group), AVI(Audio-Video Interleaved), MP3(MPEG Audio Layer-3) 등과 같은 다양한 포맷(Format)을 모두 지원한다. 이렇게 다양한 소스, 포맷, 하드웨어를 다루기 위해서 다이렉트쇼에서는 ‘필터(Filter)’ 기술을 제공한다. 필터는 COM(Component Object Model) 기반으로 구성되며 필터 그래프 상에 필터 간 연결을 통하여 데이터 흐름을 제어할 수 있다.

그러나 수많은 필터들이 개발되고 액티브엑스(ActiveX)를 통해 자동 설치되거나 많은 멀티미디어 솔루션들이 필터기반으로 개발됨에 따라서 컴퓨터 내에 비슷한 역할을 하는 필터들이 본인도 모르게 컴퓨터 내에 설치될 수 있다. 따라서 필터 간 연결이 정상적으로 이루어 지지 않게 되고, 원하지 않는 결과가 출력되거나 멀티미디어 데이터가 전혀 출력되지 않는 결과를 가져올 수 있다.

본 논문에서는 필터차단기법을 이용하여 필요한 필터만 연결하여 원하는 데이터 출력이 가능하도록 하는 기법을 제안한다. 논문의 구성은 2장에서 기반 기술인

다이렉트쇼의 일반적인 개요에 대해 알아보고, 3장에서 필터차단의 필요성과 차단방법을 소개하고 비교하였다. 4장에서는 구현결과를 통하여 필터차단기법의 이용 여부에 따른 성능비교를 하였고, 마지막으로 5장에서 결론을 맺었다.

II. 다이렉트쇼

1. 다이렉트쇼 시스템 개요

다이렉트쇼의 주요 목표는 멀티미디어 데이터 전송, 하드웨어, 데이터 동기화를 개발자가 특별히 설계하지 않고도 멀티미디어 솔루션 개발을 쉽게 할 수 있게 한다.

다이렉트쇼에서는 비디오와 오디오 데이터 출력처리를 하기위해서 다이렉트드로우(DirectDraw) 및 다이렉트사운드(DirectSound)를 사용한다. 이것은 다이렉트엑스(DirectX) 기술의 한 부분이며, 사용자의 그래픽카드와 사운드카드를 효율적으로 렌더링(Rendering)할 수 있다. 또한 비디오와 오디오의 동기화를 위해 미디어 데이터가 포함하고 있는 DTS(Decoding Time Stamp)나 PTS(Presentation Time Stamp)정보를 이용한다. 다양한 입력방법과 멀티미디어 포맷을 지원하기 위해서 모듈화된 형태의 필터(Filter)를 사용한다. 필터는 COM개체를 이용한 개발 시 용의하게 이용할 수 있다. 따라서 이 구성요소를 조합하여 솔루션 개발에 사용한다. [그림 1]에서는 다이렉트쇼 시스템 구조도를 나타낸다.

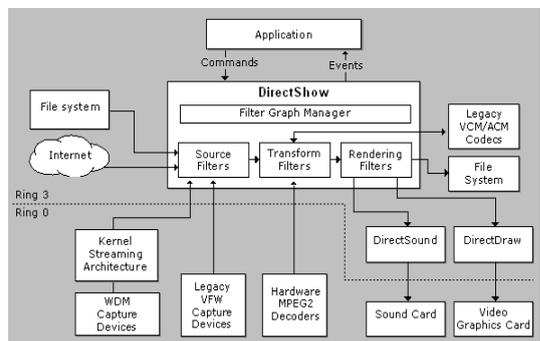


그림 1. 다이렉트쇼 시스템 구조도[2]

2. 필터 종류

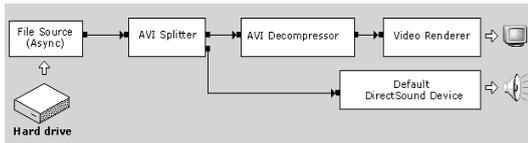


그림 2. 필터연결[2]

[그림 2]에서 보듯이 1개 또는 복수의 다른 필터에 연결되고 있다. 필터 간 연결은 핀(Pin)이라 불리는 COM 개체를 통하여 이루어진다. 필터는 핀을 통하여 데이터를 이전필터에서 다음 필터로 이동시키며, 데이터의 흐름 방향은 화살표의 방향과 같다. 다이렉트쇼에서는 연결된 필터의 집합을 필터그래프(Filtergraph)라 한다.

필터는 몇 개의 종류로 분류할 수 있다.

- 소스필터
 - : 데이터 입력 단으로 파일, 네트워크, 카메라 등으로 부터 입력.
- 변환필터
 - : 입력 데이터를 처리하여 출력 데이터를 생성하는 엔코더(Encoder), 디코더(Decoder)와 같은 코덱.
- 렌더필터
 - : 필터 연결의 마지막에 존재하며 데이터를 출력함. 비디오 렌더러는 프레임을 화면에 나타내고, 오디오 렌더러는 사운드카드로 송신. 파일 저장필터는 출력데이터를 파일로 관리.
- 스플리터(Splitter) 필터
 - : 입력 데이터를 분석하여 복수의 출력으로 구분함. 비트스트림 데이터를 오디오, 비디오, 텍스트 데이터로 구분. 파서(Parser) 필터라고 함.
- 믹스(Mux) 필터
 - : 복수의 입력을 받아서 하나의 데이터로 합성. 스플리터 필터와 반대로 작동.

3. 필터 협상

3.1 미디어협상

미디어 타입은 멀티미디어 데이터 포맷을 기술하기 위한 범용적인 정보이다. 다이렉트쇼에서는 두 필터의 핀 연결을 위해서 상호간의 필터가 교환하려고 하는 데이터의 미디어 타입을 상호간 협상(Negotiation)하게 된다. 미디어 타입은 업-스트림(Up-Stream) 필터가 다운 스트림(Down-Stream) 필터에게 보내어 식별하게 되며, 만약 2개의 필터가 협상(미디어 타입에 대해 합의)이 이루어 지지 못한 경우 필터 연결을 할 수 없다. [그림 3]에서는 미디어 협상을 위한 작업흐름관리를 나타내며 모든 필터는 연결 시 이 과정이 이루어진다.

3.2 버퍼협상

미디어협상을 통하여 필터 간 연결이 된 후 필터 간 데이터 전달을 위한 버퍼협상을 한다. 버퍼협상 시 다음과 같은 조건을 확인한다.

- 두 핀간 데이터 전달시 메모리 공유
- 데이터 전달 주체
- 데이터 전달 측과 수신측의 동기

이와 같은 사항들을 확인한 후에 샘플 데이터를 업 스트림 필터에서 다운 스트림 필터 쪽으로 전달한다. 두 핀 사이에 데이터가 전달되기 위해서는 정해진 메모리를 출력핀에서 데이터를 복사해주고, 입력핀에서는 복사한 데이터 정보를 읽을 수 있어야 한다. 다이렉트쇼에서 출력핀과 입력핀이 공유하는 정보를 미디어샘플(Media Sample)이라 부르며, 할당자에서 미디어샘플을 할당하게 된다.

[그림 4]에서는 할당자, 미디어샘플, 필터의 관계에 대하여 나타낸다.

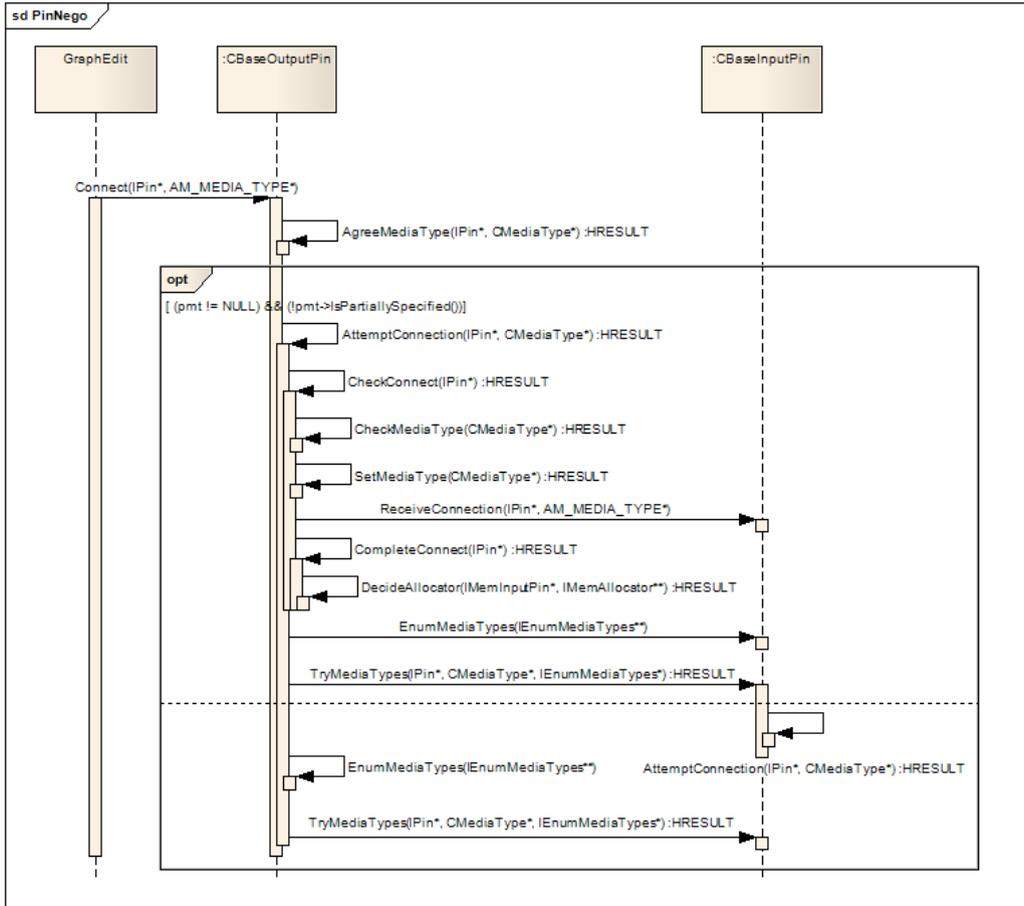


그림 3. 미디어 협상을 위한 작업흐름관리

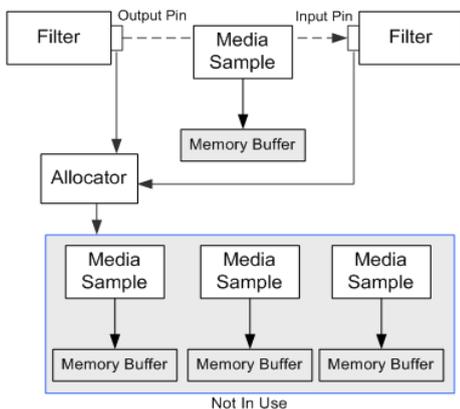


그림 4. 할당자, 미디어샘플, 필터의 관계

4. 필터연결

4.1 수동연결

수동연결은 필터 간 연결을 할 때 직접 연결하는 방법이다. 수동연결을 통하여 모든 필터들을 하나씩 연결하여 원하는 필터구성을 이룰 수 있다. 그러나 필터를 한 개씩 다 생성하고 필터그래프에 추가시켜야 하기 때문에 연결하고자 하는 필터정보를 정확히 알고 있어야 하며, 개발자들이 구현해야 할 코드들이 많아지는 단점이 있다.

4.2 자동연결

필터그래프 매니저가 자동으로 필터그래프를 생성하

기 위해서 사용하는 방법이다. 이 연결방법은 필터에 대한 상세정보 없이 자동으로 연결한다. 따라서 필터구성 시 편리하고 개발자가 소스필터와 렌더필터의 연결만으로도 플레이어와 같은 멀티미디어 솔루션 개발의 편의성을 제공한다.

자동연결 시 필터 연결을 위한 검사를 하는데 그 순서는 다음과 같다.

- 필터그래프 매니저는 필터의 연결을 IStreamBuilder::Render 함수에 위임하며, 필터들을 생성 후 렌더 필터까지 연결함.
- 캐시(Cache)된 필터들을 연결함.
- 필터그래프내 이미 추가되어 있는 필터들을 연결함.
- 레지스트리내 필터들을 검색하여 연결함.
(단 메리트(Merit)값이 MERIT_DO_NOT_USE 이 하인 카테고리에는 무시함.)

4.3 필터연결비교

다이렉트쇼 필터를 이용한 멀티미디어 솔루션 개발 시 필터연결방법에 따라 장단점이 존재하므로, 개발자는 상황에 맞게 하나를 선택하여 사용하게 된다. [표 1]에서 각 필터연결방법의 장단점을 비교, 분석하였다.

표 1. 필터연결방법의 비교

	수동연결	자동연결
코드복잡도	복잡	단순
개발편의성	어려움	편리함
필터연결속도	아주 빠름	보통 빠름
불필요필터생성	생성불가능	생성가능

사용자 컴퓨터 내에는 통합코덱, DMO(DirectX Media Objects) 코덱과 같은 수많은 필터들이 설치되어 있다. 따라서 멀티미디어 솔루션 개발 시 자동연결을 사용하면 이미 컴퓨터 내 존재하는 필터들을 사용할 수 있으므로 지원되지 않았던 코덱들이 지원되는 등 서비스 지원 범위가 광범위 해지고 개발자 입장에서도 쉽고 편리하게 솔루션을 개발할 수 있다.

III. 제안하는 필터차단기술

1. 필터차단의 필요성

필터를 연결할 때 자동연결을 이용하면 개발의 편의성 뿐만 아니라 서비스 범용성의 장점이 있다. 그러나 수많은 필터들이 컴퓨터 내에 존재하고 그 수가 많아짐에 따라 필터 간 연결이 잘못될 수 있다. 따라서 원하지 않는 화면출력이 나올 수 있고, 다른 필터들의 환경 설정 때문에 원하지 않는 설정이 구성되기도 한다.



그림 5. 자막중복출력 화면

[그림 5]는 영화 “센과 치이로의 행방불명”의 한 장면이다. 이것을 보면 자막이 2개가 나오는 것을 확인할 수 있다. 이는 소스필터 내부의 자막해석기에 의해서 나오는 자막화면과 자막진용필터인 DirectVobSub를 통한 자막화면이 동시에 나오고 있다. DirectVobSub는 통합코덱이나 오픈소스형태로 널리 사용되고 있는 필터로서 자막화면 관리를 효율적으로 할 수 있는 기능을 가진다. 그러나 어떤 상황에서는 [그림 5]와 같이 오히려 불필요한 필터의 역할을 하기도 한다.

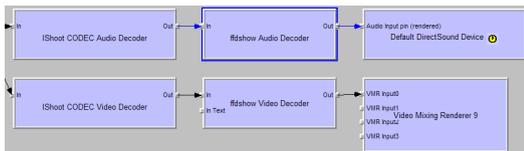
[그림 6]의 (a)에서는 그래프에디터(Grapheditor)상에서 코덱필터가 중복 연결된 것을 보여준다. 이와 같은 현상은 통합코덱이 설치된 컴퓨터에서 자주 발생된

다. 필터의 특성상 미디어 협상 시에 타입이 허용되기 때문에 코덱필터 중에 메리트가 높은 필터가 있으면 항상 발생하게 된다. 이런 경우에는 첫 번째 코덱필터의 환경설정과 그 뒤에 바로 연결된 코덱필터의 환경변수를 각각 설정 할 수 있다. [그림 6]의 (b)에서는 중복된 필터의 환경설정 프로퍼티를 보여준다. 따라서 다이렉트쇼를 이용한 멀티미디어 솔루션에서는 전에 설정해 놓은 사용자 환경설정과 상관없이 중복된 코덱필터의 환경설정에 따라 작동될 수 있다.

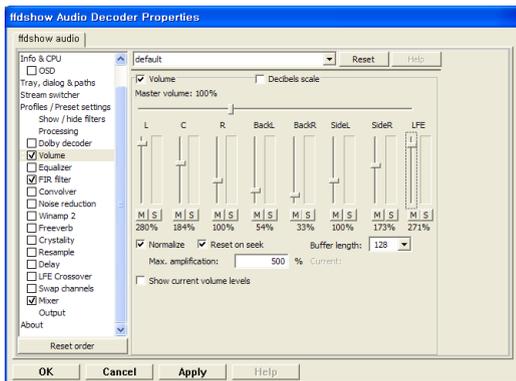
일반적으로 사용되는 필터자동연결의 문제점을 정리하면 아래와 같다.

- 원하지 않은 필터의 연결을 통한 비정상적인 작동 [그림 5]
- 메리트가 높은 중복필터 연결을 통한 원하지 않은 환경설정의 변경[그림 6]

따라서 자동연결의 장점을 그대로 유지하면서 다른 필터의 영향으로부터 독립적으로 되기 위해서는 불필요한 필터들을 차단하는 방법이 반드시 필요하다.



(a) 코덱필터 중복연결



(b) 중복코덱필터 프로퍼티설정

그림 6. 코덱필터 중복연결과 프로퍼티설정

2. 필터내부차단

미디어 협상을 통하여 필터 간 출력핀과 입력핀의 협상이 이루어진다. 입력핀에서는 CPin::QueryPinInfo 함수를 통해 출력핀의 필터 정보를 얻어올 수 있다. 필터들은 16 byte의 고유의 CLSID를 가진다. 이것을 이용하여 목록 관리를 하고 해당 필터들만 연결 할 수 있다. 이 목록을 화이트리스트(Whitelist)라 부른다. 보통 FFDSHOW[6]과 같은 통합코덱필터는 프로그램단위로 화이트리스트를 관리한다. FFDSHOW에서 화이트리스트에 등록하려면 레지스트리에서 HKEY_CURRENT_USER/Software/GNU/ffdshow/whitelist 레지스트리키에 사용하고자 하는 프로그램 이름을 등록하면 해당 솔루션을 사용할 수 있다.

본 논문에서는 INonDelegatingUnknown::NonDelegatingQueryInterface 함수 내에서 화이트리스트 유무를 검사하고 사용가능한 솔루션들을 검사한다. NonDelegatingQueryInterface 함수는 인터페이스 포인터를 가져오고 레퍼런스카운트(Reference Count)를 하나 증가 시키는 역할을 한다. 사용하고자 하는 모든 필터에 이를 적용하여 해당 필터 내에서 원하는 솔루션만 사용 가능하도록 지정할 수 있다. 화이트리스트를 관리할 때 레지스트리나 ini파일로 관리를 하면 소스코드를 다시 컴파일 하지 않고 사용할 수 있다. 또한 해당 필터만 사용가능하게 하려면 다운 스트림 입력 핀의 IPin::ReceiveConnection 함수 내에서 업 스트림의 입력핀을 통해 CLSID를 검사하고, 원하지 않는 필터 접속 시 VFW_E_TYPE_NOT_ACCEPTED의 값을 리턴한다.

필터 내부차단이 가장 확실한 방법이나 필터를 제작하는 개발자만이 초기에 리스트관리를 할 수 있도록 코드설계를 해야 하는 단점이 있다. 따라서 필터 개발자가 아닌 일반 어플리케이션 개발자들은 필터 내부차단 방법을 사용할 수 없는 단점을 가진다.

3. 필터외부차단

필터개발은 비디오, 오디오 코덱과 같은 멀티미디어 기반지식과 COM을 기반으로 한 다이렉트쇼 프레임워크를 이해해야만 개발을 할 수 있다. 따라서 필터개발

과 상관없이 일반 어플리케이션 개발자들도 이미 존재하는 필터를 차단하는 방법이 필요하겠다. 이것은 핀의 “동적재접속” 기술을 이용하여 필터차단에 이용할 수 있다. 우선 필터자동연결을 통하여 렌더링이 가능하도록 필터그래프를 완성한다. 물론 연결된 필터들이 개발자가 원하는 필터들로 연결되지 않은 경우에만 “동적재접속”을 진행한다. 그러나 대부분의 컴퓨터에서는 액티브엑스나 통합코덱설치를 통하여 본인도 모르게 수많은 필터들이 설치되어 있기 때문에 원하는 필터만 연결될 가능성은 적다. 따라서 필터외부차단을 통하여 원하는 필터만 연결할 수 있다. [그림 7]은 필터연결을 설명하기 위한 필터연결 그래프를 나타낸다.



그림 7. 필터연결 그래프

동적재접속은 다음과 같은 순서로 행해진다.

- 핀 A의 데이터 스트림을 막기 위해 블록 구성.
- 새로운 중간 필터 3 추가 후 핀 A와 핀 D를 재접속.
- 데이터 흐름이 재개되도록, 핀 A의 블록 해제.

데이터 스트림에 대한 블록을 만들기 위해서, 핀 A로 IPinFlowControl::Block 함수를 호출한다. 이 함수를 비동기적으로 호출하기 위해 이벤트 핸들을 Block 함수에 전달한다. 단 동기적으로 호출할 때는 NULL값을 전달한다. 핀을 재접속하기 위해서는 IGraphConfig::Reconnect 함수를 이용한다. 이 함수를 사용하기 위해서는 필터 2를 정지한 후, 그래프로부터 필터 2를 삭제한다. 필터 2는 개발자가 연결을 원하지 않는 필터리스트로 관리하여 삭제한다. 연결된 필터를 검색하기 위해서는 IEnumFilter::Next 함수를 이용하여 하나씩 열거 후에 IGraphConfig::RemoveFilterEx 함수를 이용하여 불필요한 필터들을 삭제한다. 그리고 필요에 따라 새로운 중간 필터를 추가한 후 모든 핀을 재접속한다. 그 후 첫 번째 인자에 '0'으로 넣어 IPinFlowControl::Block을 호출하여 블록을 해제한다.

4. 필터차단비교

필터차단방법을 비교하면 [표 2]와 같다. 내부차단이나 외부차단과 같은 필터차단기법을 이용하면 불필요한 필터 연결에 따른 오류들을 막을 수 있다. 따라서 개발자들은 각각의 상황에 맞게 차단기법을 적용하여 멀티미디어 솔루션 개발에 이용할 수 있다.

표 2. 필터차단방법의 비교

	내부차단	외부차단
필터소스	필요	불필요
연결호환성	App지정가능, 필터지정가능	필터지정가능
정확도	불필요 필터연결가능	필요필터만연결
필터연결시간	빠름	보통

IV. 구현결과

1. 시스템 구현 환경

표 3. 시스템 구현환경

시스템 환경	상세내역
OS	윈도우즈 비스타
개발SDK	다이렉트쇼 / Platform SDK / Direct 3D
필터	파서필터 : Guliverkli2 코덱필터 : iShoot 렌더필터 : VMR9 사용필터 : FFDSHOW, 곰플레이어, Nero, Arcsoft
컴파일러	Microsoft Visual Studio 2008
개발 언어	C++ / Macro Assembler

필터차단기술을 테스트하기 위하여 다이렉트쇼를 이용한 필터기반의 플레이어를 개발하였다. OS(Opera-tion System)로 윈도우즈 비스타(Windows Vista)를 사용하였으며 여러 가지 필터들을 테스트하기 위해 FFDSHOW 통합코덱필터들을 사용하였다. 플레이어 필터 구성 시 다양한 압축포맷을 지원하기 위하여 코덱 필터(CODEC Filter)는 ffmpeg 오픈소스 기반의 iShoot[7] 코덱필터를, 파서필터로 Guliverkli2 오픈소스를 사용하였다[8]. 또한 렌더필터는 VMR9(Video

Mixer Render9)를 사용하여 플레이어의 렌더링 성능을 높였다. 또한 사용자 UI의 편의성을 제공하기 위해 Direct3D를 이용하였다.

2. 시스템 프로토타입



그림 8. 플레이어

필터 차단방법을 비교 연구하기 위하여 멀티미디어 콘텐츠 재생을 위한 플레이어를 구성하였다. 완성된 플레이어의 모습은 [그림 8]와 같다. 플레이어 내의 화면은 영화 “과속스캔들”의 한 장면이다.

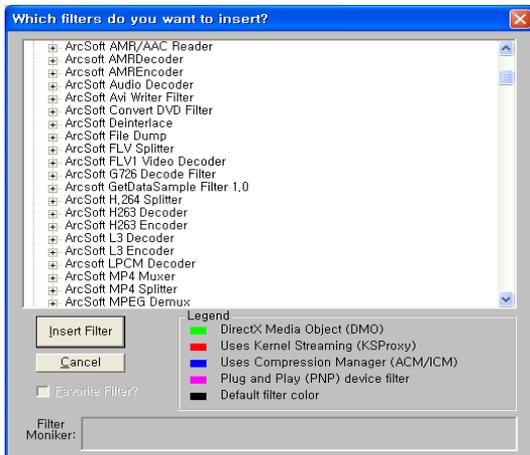


그림 9. Arcsoft 필터모음

또한 필터연결 오류를 발생시키기 위하여 다양한 FFDSHOW, 곰플레이어, Nero, Arcsoft와 같은 다수의

필터들을 미리 설치해 놓은 상태에서 테스트 하였다.

[그림 9]은 컴퓨터에 다양한 필터들이 설치된 것을 보기 위하여 그래픽에디터에 있는 다이렉트쇼 필터 모음중 Arcsoft의 멀티미디어 필터를 보여준다.

다양한 필터모음 속에서 필터차단기법을 이용하여 원하는 필터들 간의 연결을 구성할 수 있다.

3. 시스템 성능비교

필터연결방법을 비교하면 [표 4]와 같다.

표 4. 필터연결방법에 따른 비교

	자동연결	필터추가후 자동연결	필터추가후 자동연결+필터 차단
연결성	못할 수 있음	연결	연결
연결 정확도	원하는 필터접속 못할 수 있음	원하는 필터 접속	원하는 필터 접속
불필요한 필터연결	가능	가능	불가능
필터연결시간	조금빠름	빠름	빠름

위 사항을 성능비교하기 위하여 한 가지 콘텐츠를 다양한 코덱(H.264, MPEG4, AAC, MP3)등으로 부호화하였다. 또한 25개의 콘텐츠에 대해 마찬가지로 다른 코덱으로 부호화 하여, 총 100개의 필터 연결환경을 구축하였다.

자동연결만 한 경우는 연결을 못하는 경우도 생긴다. 연결하고자 하는 필터의 메리트가 낮을 경우에는 연결하고자 하는 필터를 찾을 수 없기 때문에 연결을 못하는 경우도 생긴다. 이를 방지하기 위해 필터추가 후 자동연결을 하게 된다. 이는 원하는 필터 연결이 가능하기 때문에 솔루션 개발 시 주로 이용되게 된다. 그러나 간혹 불필요한 필터가 연결되어 원하지 않는 결과를 양산하기도 한다. 그러나 필터차단방법을 이용하여 개발 솔루션이 의도한 정확한 결과를 얻어낼 수 있다.

V. 결론

다이렉트쇼를 이용하여 개발자들이 멀티미디어 데이

터 전송, 하드웨어, 데이터 동기화와 같은 것을 설계하지 않고 쉽게 솔루션들을 개발할 수 있는 환경이 제공되고 있다. 이러한 환경 속에서 수많은 다이렉트쇼 필터들이 개발 되었고 이 필터들을 요소에 맞게 정확하게 사용하기 위해서는 필터차단기법이 반드시 동반되어야 한다. 따라서 본 논문에서는 필터내부차단, 필터외부차단과 같은 기법을 이용하여 원하는 필터연결을 구축할 수 있는 방법을 제안하고 멀티미디어 플레이어를 구현하여 빠르고 정확한 필터연결이 이루어짐을 확인하였다.

참 고 문 헌

[1] A. Dasu and S. Panchanathan, "A Suvey of Media Processing Approches", IEEE Transactions on Circuit and System for Video Technology, Vol.12, pp.633-645, 2002(8).

[2] *Microsoft DirectX SDK 9.0 Document.*

[3] Z. Fan and L. Bo, "DirectShow Based Internet Video On Demand System", Microwave and Millimeter Wave Technology, ICMMT 2008 International Conference, Vol.4, pp.2077-2088, 2008(4).

[4] 김대진, 최홍섭, "휴대용 멀티미디어 디바이스를 위한 TPO(Time, Place, Occasion)-Shift 시스템 설계에 대한 연구", 한국컴퓨터정보학회논문지, 제14권, 제2호, pp.9-16, 2009(2).

[5] 김대진, 최홍섭, "사용자 맞춤형 채널 관리를 이용한 다운로드 기반의 IPTV 시스템 제안", 한국 디지털콘텐츠학회논문지, 제10권, 제1호, pp.61-71, 2009(3).

[6] <http://ffdshow-tryout.sourceforge.net>

[7] <http://www.ishoot.co.kr>

[8] <http://mpc-hc.sourceforge.net>

저 자 소 개

김 대 진(Dae-Jin Kim)

정회원



- 1998년 2월 : 대진대학교 전자공학과 졸업(공학사)
 - 2000년 2월 : 동국대학교 전자공학과 졸업(공학석사)
 - 2008년 2월 : 대진대학교 전자공학과 수료(공학박사)
 - 2000년 ~ 2003년 : 한빛소프트 주임연구원
 - 2003년 ~ 2007년 : 모토로라 코리아 전임연구원
 - 2007년 ~ 2008년 : 아이비인터넷 부장
 - 2008년 ~ 현재 : 피어컴(미디어웹) 선임연구원
- <관심분야> : 저작권 보호, 멀티미디어 시스템, 디지털 콘텐츠, 멀티미디어 검색, IPTV

최 홍 섭(Hong-Sub Choi)

정회원



- 1985년 2월 : 서울대학교 전자공학과 졸업(공학사)
 - 1987년 2월 : 서울대학교 전자공학과 졸업(공학석사)
 - 1994년 2월 : 서울대학교 전자공학과 졸업(공학박사)
 - 1995년 ~ 현재 : 대진대학교 전자공학과 정교수
- <관심분야> : 통신 및 신호처리, 음성인식, 멀티미디어 시스템, IPTV