(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0082768
(43) 공개일자 2022년06월17일

(51) 국제특허분류(Int. Cl.)
H04N 21/231 (2011.01) *H04N 21/218* (2011.01)
H04N 21/222 (2011.01) *H04N 21/81* (2011.01)
(52) CPC특허분류
H04N 21/23109 (2013.01)
H04N 21/2181 (2013.01)
(21) 출원번호 10-2021-0176051
(22) 출원일자 2021년12월09일
심사청구일자 2021년12월09일
(30) 우선권주장
1020200172409 2020년12월10일 대한민국(KR)

(71) 출원인
경희대학교 산학협력단
경기도 용인시 기흥구 덕영대로 1732 (서천동, 경희대학교 국제캠퍼스내)
(72) 발명자
이영구
경기도 수원시 영통구 청명북로 33, 431동 1704호(영통동, 청명마을삼성아파트)
아프탑 알람
경기도 용인시 기흥구 덕영대로 1732 (서천동)
(74) 대리인
유미특허법인

전체 청구항 수 : 총 19 항

(54) 발명의 명칭 **클라우드에서의 지능형 비디오 빅데이터 분석 시스템 및 그 방법**

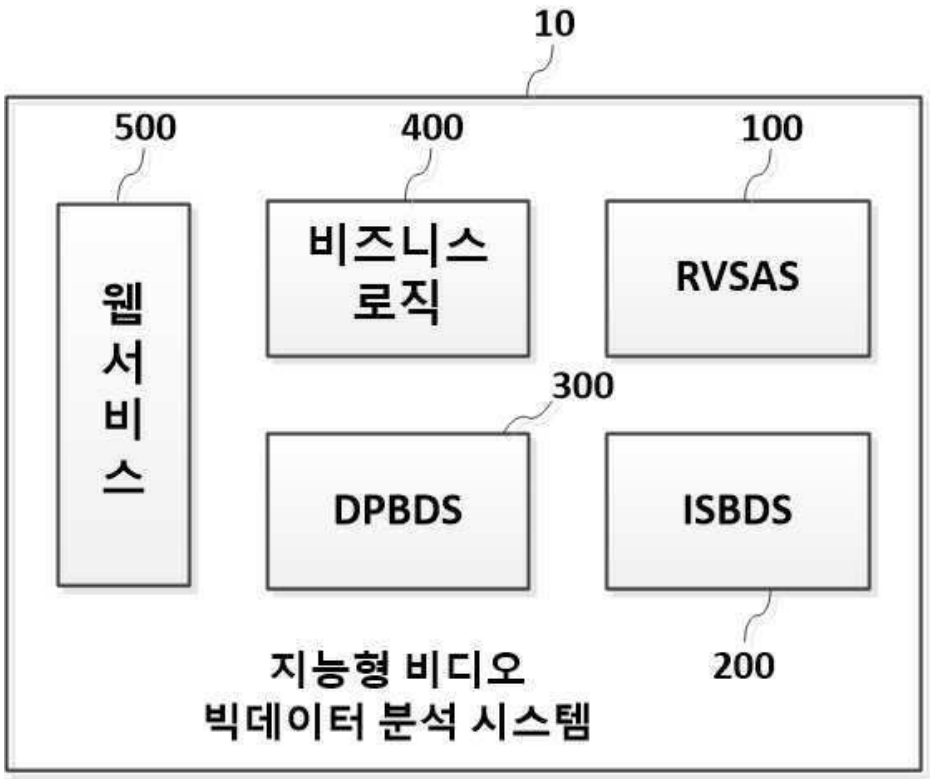
(57) 요 약

클라우드에서의 지능형 비디오 빅데이터 분석 시스템 및 그 방법이 개시된다.

이 시스템은 클라우드에서 지능적으로 비디오 빅데이터를 분석하는 시스템으로서, 외부의 실시간 비디오 수집 도구로부터 수집되는 실시간 비디오 스트림을 수집하는 실시간 비디오 스트림 획득부를 포함한다. 또한, 이 시스

(뒷면에 계속)

대 표 도 - 도1



템은 영구적이고 분산된 빅데이터 저장소를 제공하며, 실시간 비디오 스트림의 원시 비디오 스트림을 저정하는 제1 빅데이터 저장소, 분산 환경에서 대규모의 구조화된 데이터를 관리하며, 실시간 비디오 스트림에서 처리된 메타 데이터와 실시간 비디오 스트림에 기반한 실시간 지능형 비디오 분석(RIVA) 중에 생성되는 중간 결과(IR)를 저장하는 제2 빅데이터 저장소, 그리고 실시간 비디오 스트림 획득부, 제1 빅데이터 저장소 및 제2 빅데이터 저장소의 추상화로서, 웹 서비스를 통한 사용자의 데이터 액세스를 제공하는 비즈니스 로직을 포함한다. 실시간 비디오 스트림 획득부는 RIVA 중에 생성되는 중간 결과를 읽어서 재컴파일을 피하는 재사용을 위해 제2 빅데이터 저장소에 저장하는 중간 결과 관리기를 포함한다.

(52) CPC특허분류
H04N 21/2223 (2013.01)
H04N 21/8173 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711126043
과제번호	2016-0-00406-006
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	SW컴퓨팅산업원천기술개발(R&D)
연구과제명	(기반SW-창조씨앗2단계) SIAT형 CCTV 클라우드 플랫폼 기술 개발
기 여 율	1/1
과제수행기관명	숙명여자대학교 산학협력단
연구기간	2021.01.01~2021.12.31

명 세 서

청구범위

청구항 1

클라우드에서 지능적으로 비디오 빅데이터를 분석하는 시스템으로서,

외부의 실시간 비디오 수집 도구로부터 수집되는 실시간 비디오 스트림을 수집하는 실시간 비디오 스트림 획득부,

영구적이고 분산된 빅데이터 저장소를 제공하며, 상기 실시간 비디오 스트림의 원시 비디오 스트림을 저정하는 제1 빅데이터 저장소,

분산 환경에서 대규모의 구조화된 데이터를 관리하며, 상기 실시간 비디오 스트림에서 처리된 메타 데이터와 상기 실시간 비디오 스트림에 기반한 실시간 지능형 비디오 분석(Real-time Intelligent Video Analytics, RIVA) 중에 생성되는 중간 결과(Intermediate Result, IR)를 저장하는 제2 빅데이터 저장소, 그리고

상기 실시간 비디오 스트림 획득부, 상기 제1 빅데이터 저장소 및 상기 제2 빅데이터 저장소의 추상화로서, 웹 서비스를 통한 사용자의 데이터 액세스를 제공하는 비즈니스 로직

을 포함하며,

상기 실시간 비디오 스트림 획득부는 상기 RIVA 중에 생성되는 중간 결과를 읽어서 재컴파일을 피하는 재사용을 위해 상기 제2 빅데이터 저장소에 저장하는 중간 결과 관리기를 포함하는,

지능형 비디오 빅데이터 분석 시스템.

청구항 2

제1항에 있어서,

상기 실시간 비디오 스트림 획득부는,

분산 메시징 시스템을 사용하여 생산자의 클러스터에서 미니 배치 형식으로 상기 실시간 비디오 스트림을 수집하고 사용자 클러스터로 라우팅하는 브로커 클라이언트 서비스 모듈,

클라이언트 어플리케이션 프로그래밍 인터페이스로서, 대규모 비디오 스트림을 처리하기 위해 상기 생산자 클러스터에 구성되는 비디오 스트림 획득 서비스 모듈,

클라이언트 어플리케이션 프로그래밍 인터페이스로서, 상기 비디오 스트림 획득 서비스 모듈에서 레코드를 수신하고 수신된 레코드를 처리하여 미니 배치를 구성하는 비디오 스트림 프로듀서 모듈, 그리고

상기 RIVA 중에 발생되는 이상에 대해 알림을 구성하는 라이프롱(lifelong) 비디오 스트림 모니터

를 포함하는, 지능형 비디오 빅데이터 분석 시스템.

청구항 3

제2항에 있어서,

상기 브로커 클라이언트 서비스 모듈은,

새로운 RIVA 서비스 생성 시 브로커 클러스터에서 동적으로 새로운 토픽을 생성하는 토픽 관리기, 그리고

비디오 스트림 데이터 소스의 비디오 스트림과 소비자 그룹의 스트림 소비에 따라 토픽의 파티션 개수를 자동으로 조정하는 파티션 관리기

를 포함하는, 지능형 비디오 빅데이터 분석 시스템.

청구항 4

제3항에 있어서,

상기 토픽 관리기는 상기 새로운 RIVA 서비스가 생성되는 경우 상기 브로커 클러스터 내에, 상기 RIVA 서비스를 나타내는 RIVA_ID 토픽, 중간 결과를 나타내는 RIVA_IR_ID 토픽 및 이상 발생을 나타내는 RIVA_A_ID 토픽을 구성하는,

지능형 비디오 빅데이터 분석 시스템.

청구항 5

제2항에 있어서,

상기 비디오 스트림 획득 서비스 모듈은,

장치 독립적인 비디오 스트림 소스를 위한 인터페이스를 제공하는 비디오 스트림 획득 어댑터,

비디오 스트림을 디코딩하고 프레임을 감지한 다음 메타 데이터 추출을 수행하는 비디오 스트림 이벤트 프로듀서,

추출된 메타 데이터에 대해 프레임 크기 조정을 수행하는 프레임 전처리기, 그리고

상기 메타 데이터에 대해 비디오 스트림 소스와 통신하기 위한 JSON(JavaScript Object Notation) 개체를 정의하는 레코드 구성기

를 포함하는, 지능형 비디오 빅데이터 분석 시스템.

청구항 6

제2항에 있어서,

상기 비디오 스트림 프로듀서 모듈에서의 레코드는 기본 메시지 형식을 가지며, 토픽 이름(Topic Name), 파티션 번호(Partition Number), 키(Key) 및 값(Value)의 필드를 포함하는,

지능형 비디오 빅데이터 분석 시스템.

청구항 7

제2항에 있어서,

상기 중간 결과 관리기는 상기 브로커 클러스터 내의 RIVA_IR_ID 토픽과 중간 결과를 송수신하여 분산된 환경에서의 관리를 수행하는,

지능형 비디오 빅데이터 분석 시스템.

청구항 8

제2항에 있어서,

상기 라이프롱 비디오 스트림 모니터는 상기 RIVA 중에 이상이 발생되는 경우 상기 브로커 클러스터 내의 RIVA_A_ID로 전달하는,

지능형 비디오 빅데이터 분석 시스템.

청구항 9

제1항에 있어서,

상기 제1 빅데이터 저장소는,

서비스를 등록한 사용자에 대응하는 사용자 공간, 그리고

상기 사용자 공간과 상기 제1 빅데이터 저장소 내의 데이터를 관리하기 위한 액세스 제어기

를 포함하며,

상기 액세스 제어기는,

상기 비즈니스 로직에 따라 파일 작업 및 관한 제어를 관리하는 파일 시스템 핸들러, 그리고

실시간 및 오프라인 읽기-쓰기 작업을 수행하는 능동 데이터 액세스기

를 포함하는, 지능형 비디오 빅데이터 분석 시스템.

청구항 10

제1항에 있어서,

상기 제2 빅데이터 저장소는,

상기 지능형 비디오 빅데이터 분석 시스템에서 사용되는 데이터를 저장하는 복수 유형의 저장소, 그리고

상기 비즈니스 로직에 따라 상기 복수 유형의 저장소에 대한에 대한 읽기-쓰기 액세스를 제공하는 액세스 제어기

를 포함하는, 지능형 비디오 빅데이터 분석 시스템.

청구항 11

제10항에 있어서,

상기 복수 유형의 저장소는,

사용자에게 역할 기반 액세스를 제공하며, 사용자 로그 및 역할 정보가 사용자 카탈로그를 통해 유지되는 사용자 프로파일 및 로그,

RIVA 서비스에 가입할 수 있는 비디오 스트림 소스와 배치 IVA(Batch IVA, BIVA) 서비스에 가입할 수 있는 비디오 데이터 세트를 관리하는 비디오 데이터 소스 메타 스토어,

IVA 알고리즘과 서비스를 관리하는 알고리즘 및 서비스 메타 스토어,

상기 중간 결과 관리기에 의해 전달되는 중간 결과를 관리하는 중간 결과 미들웨어, 그리고

RIVA 서비스에 대한 비디오 스트림 소스를 구독하는 사용자의 구독 정보를 관리하는 구독 메타 스토어

를 포함하는, 지능형 비디오 빅데이터 분석 시스템.

청구항 12

제10항에 있어서,

상기 액세스 제어기는,

클러스터 구성, 연결 정보, 드라이버 및 보안 프로토콜 파라미터와 같은 구성 파라미터를 유지 관리하는 연결 구성기,

상기 제2 빅데이터 저장소의 스키마 구조를 유지하는 스키마 핸들러,

상기 제2 빅데이터 저장소에서 지속되는 데이터에 대한 생성, 읽기, 업데이트 및 삭제 작업을 수행하는 CRUD 오퍼레이터(Create, Read, Update, Delete Operator), 그리고

대량의 데이터를 로드하고 분산 테이블에 동일하게 유지하기 위한 읽기-쓰기 작동을 제공하는 수동 액세스기

를 포함하는, 지능형 비디오 빅데이터 분석 시스템.

청구항 13

제1항에 있어서,

상기 비즈니스 로직은,

새로운 사용자 계정 생성, 액세스 역할 할당 및 세션 관리와 같은 모든 사용자 관련 작업을 캡슐화하는 사용자 관리기,

사용자가 상기 지능형 비디오 빅데이터 분석 시스템의 배포 클라우드에서 데이터를 관리할 수 있게 하는 데이터

소스 관리기,

새로운 IVA 알고리즘 및 서비스의 관리, 개발 및 지원을 수행하는 IVA 알고리즘 및 서비스 관리기,

개발자가 의사 결정을 위한 중간 결과를 얻을 수 있게 하는 온톨로지 데이터 관리기, 그리고

시스템 관리자가 상기 지능형 비디오 빅데이터 분석 시스템의 상태와 기능을 모니터링할 수 있게 하는 클러스터 관리기

를 포함하는, 지능형 비디오 빅데이터 분석 시스템.

청구항 14

비디오 빅데이터 분석 시스템이 클라우드에서 지능적으로 비디오 빅데이터를 분석하는 방법으로서,

스피드 계층 실행 시나리오에 따라, 외부 비디오 스트림 소스에 대한 인터페이스를 제공하는 제1 클러스터를 통해 비디오 스트림을 획득하는 단계 ,

획득되는 비디오 스트림을 브로커 클러스터인 제2 클러스터의 RIVA_ID 토픽에 로드하는 단계, 그리고

실시간 지능형 비디오 분석(RIVA) 서비스를 사용하면서 실시간으로 비디오 스트림을 처리하는 제3 클러스터가 상기 제2 클러스터의 RIVA_ID 비디오 스트림에 대해 RIVA 서비스를 제공하는 단계

를 포함하며,

상기 RIVA 서비스를 제공하는 단계에서, 상기 RIVA 서비스 중에 생성되는 중간 결과는 상기 제2 클러스터의 RIVA_IR_ID 토픽으로 전달되고, 중간 결과를 저장 관리하는 제4 클러스터가 상기 제2 클러스터의 RIVA_IR_ID 토픽에서 중간 결과를 지속적으로 읽어서 저장 관리하는,

지능형 비디오 빅데이터 분석 방법.

청구항 15

제14항에 있어서,

상기 제2 클러스터의 RIVA_ID 토픽에 로드하는 단계 후에,

비디오 스트림을 저장하는 제5 클러스터가 상기 제2 클러스터의 RIVA_ID 토픽에서 비디오 스트림을 읽는 단계,

상기 비디오 스트림에서 메타 데이터를 처리하는 단계, 그리고

상기 비디오 스트림을 제1 빅데이터 저장소에 저장하고, 상기 메타 데이터를 제2 빅데이터 저장소에 각각 저장하는 단계

를 더 포함하는 지능형 비디오 빅데이터 분석 방법.

청구항 16

제15항에 있어서,

상기 RIVA 서비스를 제공하는 단계에서, 상기 RIVA 서비스 중에 이상 현상이 발생되는 경우, 발생된 이상 현상을 상기 제2 클러스터의 RIVA_A_ID 토픽으로 전달하고, 이상 발생을 통지하는 제6 클러스터가 상기 제2 클러스터의 RIVA_A_ID 토픽에서 이상 현상을 읽어서 상기 제2 빅데이터 저장소에 저장하고 웹 서비스를 통해 상기 비디오 스트림 소스의 소유자에게 실시간으로 전달하는,

지능형 비디오 빅데이터 분석 방법.

청구항 17

비디오 빅데이터 분석 시스템이 클라우드에서 지능적으로 비디오 빅데이터를 분석하는 방법으로서,

배치(batch) 계층 실행 시나리오에 따라, 배치 비디오 획득 서비스를 제공하는 제1 클러스터를 통해 외부 배치 데이터를 획득하는 단계,

획득되는 배치 데이터를 처리하여 배치 비디오 데이터와 메타 데이터를 추출하는 단계,

상기 배치 비디오 데이터를 제1 빅데이터 저장소에 저장하고, 상기 메타 데이터를 제2 빅데이터 저장소에 저장하는 단계, 그리고

배치 지능형 비디오 분석(Batch Intelligent Video Analytics, BIVA) 서비스를 사용하는 제2 클러스터가 상기 제1 빅데이터 저장소에 저장된 배치 비디오 데이터에 대해 BIVA 서비스를 제공하는 단계

를 포함하며,

상기 BIVA 서비스를 제공하는 단계에서, 상기 BIVA 서비스 중에 생성되는 중간 결과는 상기 제2 빅데이터 저장소에 지속적으로 저장되고 다시 상기 제2 빅데이터 저장소로부터 읽혀져서 상기 BIVA 서비스 제공시 사용되는,

지능형 비디오 빅데이터 분석 방법.

청구항 18

제17항에 있어서,

상기 BIVA 서비스를 제공하는 단계 전에,

모델 저장 서비스를 제공하는 제3 클러스터를 통해 외부 배치 모델 데이터를 획득하는 단계,

획득되는 배치 모델 데이터를 처리하여 모델 데이터와 파일 메타 데이터를 추출하는 단계, 그리고

상기 모델 데이터를 상기 제1 빅데이터 저장소에 저장하고, 상기 파일 메타 데이터를 상기 제2 빅데이터 저장소에 저장하는 단계

를 더 포함하고,

상기 IVA 서비스를 제공하는 단계에서, 상기 제2 클러스터가 상기 제1 빅데이터 저장소에 저장된 배치 비디오 데이터와 모델 데이터에 대해 상기 BIVA 서비스를 제공하는,

지능형 비디오 빅데이터 분석 방법.

청구항 19

제17항에 있어서,

상기 BIVA 서비스를 제공하는 단계에서, 상기 BIVA 서비스 중에 이상 현상이 발생되는 경우, 발생된 이상 현상을 상기 제2 빅데이터 저장소에 전달하여 저장하는,

지능형 비디오 빅데이터 분석 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 클라우드에서의 지능형 비디오 빅데이터 분석 시스템 및 그 방법에 관한 것이다.

배 경 기 술

[0002] 비디오는 정기적으로 녹화되고 클라우드에 업로드된다. 영상 생성에 적극적으로 기여하고 있는 소스로는 CCTV, 스마트폰, 드론 등이 있으며, 이는 영상 관리 시스템에서빅데이터 혁명을 일으켰다.

[0003] 대규모의 복잡한 비디오 데이터를 처리하는 것은 기존의 데이터 분석 접근 방식을 사용할 합당한 이유가 없다. 비디오 빅데이터 분석은 비디오관리 처리 마이닝 및 조작에 대한 문제를 제기한다. 따라서, 대규모의 비정형 영상 데이터를 관리하고 분석하기 위해서는 보다 포괄적이고 정교한 솔루션이 필요하다. 또한, 비디오 데이터의 양이 많아 가능한 의사 결정을 위해 저장과 마이닝할 수 있는 유연한 솔루션이 필요하다.

[0004] 그런데, 빅데이터 분석 및 클라우드 컴퓨팅 기술의 부상으로 대규모 지능형 비디오 분석(Intelligent Video Analytics, IVA)가 현실이 되었다.

[0005] 하둡(Hadoop) 또는 스파크(Spark) 에코 시스템과 같은 빅데이터 기술은 확장 가능하고 안정적인 방식으로 대규모 데이터 세트에서 가치있는 인사이트를 처리 분석 및 추출하기 위해 분산 컴퓨팅을 위해 설계된 소프트웨어

플랫폼이다. 또한, 클라우드 컴퓨팅은 최소한의 관리 노력이나 서비스 제공자 상호작용으로 신속하게 프로비저닝 및 릴리스할 수 있는 모델이다.

[0006] 최근, 비디오 빅데이터 분석을 위해 클라우드에 분산 컴퓨팅 기술을 배포하는것이 학계와 산업계의 주목을 받고 있다. 그런데, 이러한 연구에서는 비디오 분석 가치 추출에 더 중점을 두고 클라우드에서 IVA의 수명 주기 전반에 걸쳐 가치, 볼륨, 속도 및 다양성 관리 등의 데이터 큐레이션 문제를 간과하고 있다. 마찬가지로, 클라우드에서 IVA를 위한 빅데이터 기술의 이해, 구성 및 운영은 특히 교육 환경에서 지루하고 시간 소모적이다. 또한, 클라우드의 IVA 수명 주기가 IVA를 중심으로 접근하고 있으며, 기존 솔루션은 IVA 알고리즘을 배포하는 동안 상위 수준 및 하위 수준 기능 관리와 같은 요소를 고려하지 않고 있다.

발명의 내용

해결하려는 과제

[0007] 본 발명이 해결하고자 하는 과제는 IVA 수명 주기 동안 IR을 효율적으로 관리할 수 있는 클라우드에서의 지능형 비디오 빅데이터 분석 시스템 및 그 방법을 제공하는 것이다.

과제의 해결 수단

[0008] 상기한 바와 같은 본 발명의 과제를 달성하고, 후술하는 본 발명의 특징적인 효과를 실현하기 위한, 본 발명의 특징적인 구성은 하기와 같다.

[0009] 본 발명의 일 측면에 따르면, 지능형 비디오 빅데이터 분석 시스템이 제공되며, 이 시스템은,

[0010] 클라우드에서 지능적으로 비디오 빅데이터를 분석하는 시스템으로서, 외부의 실시간 비디오 수집 도구로부터 수집되는 실시간 비디오 스트림을 수집하는 실시간 비디오 스트림 획득부, 영구적이고 분산된 빅데이터 저장소를 제공하며, 상기 실시간 비디오 스트림의 원시 비디오 스트림을 저정하는 제1 빅데이터 저장소, 분산 환경에서 대규모의 구조화된 데이터를 관리하며, 상기 실시간 비디오 스트림에서 처리된 메타 데이터와 상기 실시간 비디오 스트림에 기반한 실시간 지능형 비디오 분석(Real-time Intelligent Video Analytics, RIVA) 중에 생성되는 중간 결과(Intermediate Result, IR)를 저장하는 제2 빅데이터 저장소, 그리고 상기 실시간 비디오 스트림 획득부, 상기 제1 빅데이터 저장소 및 상기 제2 빅데이터 저장소의 추상화로서, 웹 서비스를 통한 사용자의 데이터 액세스를 제공하는 비즈니스 로직을 포함하며, 상기 실시간 비디오 스트림 획득부는 상기 RIVA 중에 생성되는 중간 결과를 읽어서 재컴파일을 피하는 재사용을 위해 상기 제2 빅데이터 저장소에 저장하는 중간 결과 관리기를 포함한다.

[0011] 여기서, 상기 실시간 비디오 스트림 획득부는, 분산 메시징 시스템을 사용하여 생산자의 클러스터에서 미니 배치 형식으로 상기 실시간 비디오 스트림을 수집하고 사용자 클러스터로 라우팅하는 브로커 클라이언트 서비스 모듈, 클라이언트 어플리케이션 프로그래밍 인터페이스로서, 대규모 비디오 스트림을 처리하기 위해 상기 생산자 클러스터에 구성되는 비디오 스트림 획득 서비스 모듈, 클라이언트 어플리케이션 프로그래밍 인터페이스로서, 상기 비디오 스트림 획득 서비스 모듈에서 레코드를 수신하고 수신된 레코드를 처리하여 미니 배치를 구성하는 비디오 스트림 프로듀서 모듈, 그리고 상기 RIVA 중에 발생되는 이상에 대해 알림을 구성하는 라이프롱(lifelong) 비디오 스트림 모니터를 포함한다.

[0012] 또한, 상기 브로커 클라이언트 서비스 모듈은, 새로운 RIVA 서비스 생성 시 브로커 클러스터에서 동적으로 새로운 토픽을 생성하는 토픽 관리기, 그리고 비디오 스트림 데이터 소스의 비디오 스트림과 소비자 그룹의 스트림 소비에 따라 토픽의 파티션 개수를 자동으로 조정하는 파티션 관리기를 포함한다.

[0013] 또한, 상기 토픽 관리기는 상기 새로운 RIVA 서비스가 생성되는 경우 상기 브로커 클러스터 내에, 상기 RIVA 서비스를 나타내는 RIVA_ID 토픽, 중간 결과를 나타내는 RIVA_IR_ID 토픽 및 이상 발생을 나타내는 RIVA_A_ID 토픽을 구성한다.

[0014] 또한, 상기 비디오 스트림 획득 서비스 모듈은, 장치 독립적인 비디오 스트림 소스를 위한 인터페이스를 제공하는 비디오 스트림 획득 어댑터, 비디오 스트림을 디코딩하고 프레임을 감지한 다음 메타 데이터 추출을 수행하는 비디오 스트림 이벤트 프로듀서, 추출된 메타 데이터에 대해 프레임 크기 조정을 수행하는 프레임 전처리기, 그리고 상기 메타 데이터에 대해 비디오 스트림 소스와 통신하기 위한 JSON(JavaScript Object Notation) 개체를 정의하는 레코드 구성기를 포함한다.

[0015] 또한, 상기 비디오 스트림 프로듀서 모듈에서의 레코드는 기본 메시지 형식을 가지며, 토픽 이름(Topic Name), 파티션 번호(Partition Number), 키(Key) 및 값(Value)의 필드를 포함한다.

[0016] 또한, 상기 중간 결과 관리기는 상기 브로커 클러스터 내의 RIVA_IR_ID 토픽과 중간 결과를 송수신하여 분산된 환경에서의 관리를 수행한다.

[0017] 또한, 상기 라이프롱 비디오 스트림 모니터는 상기 RIVA 중에 이상이 발생되는 경우 상기 브로커 클러스터 내의 RIVA_A_ID로 전달한다.

[0018] 또한, 상기 제1 빅데이터 저장소는, 서비스를 등록한 사용자에 대응하는 사용자 공간, 그리고 상기 사용자 공간과 상기 제1 빅데이터 저장소 내의 데이터를 관리하기 위한 액세스 제어기를 포함하며, 상기 액세스 제어기는, 상기 비즈니스 로직에 따라 파일 작업 및 관한 제어를 관리하는 파일 시스템 핸들러, 그리고 실시간 및 오프라인 읽기-쓰기 작업을 수행하는 능동 데이터 액세스기를 포함한다.

[0019] 또한, 상기 제2 빅데이터 저장소는, 상기 지능형 비디오 빅데이터 분석 시스템에서 사용되는 데이터를 저장하는 복수 유형의 저장소, 그리고 상기 비즈니스 로직에 따라 상기 복수 유형의 저장소에 대한에 대한 읽기-쓰기 액세스를 제공하는 액세스 제어기를 포함한다.

[0020] 또한, 상기 복수 유형의 저장소는, 사용자에게 역할 기반 액세스를 제공하며, 사용자 로그 및 역할 정보가 사용자 카탈로그를 통해 유지되는 사용자 프로파일 및 로그, RIVA 서비스에 가입할 수 있는 비디오 스트림 소스와 배치 IVA(Batch IVA, BIVA) 서비스에 가입할 수 있는 비디오 데이터 세트를 관리하는 비디오 데이터 소스 메타 스토어, IVA 알고리즘과 서비스를 관리하는 알고리즘 및 서비스 메타 스토어, 상기 중간 결과 관리기에 의해 전달되는 중간 결과를 관리하는 중간 결과 미들웨어, 그리고 RIVA 서비스에 대한 비디오 스트림 소스를 구독하는 사용자의 구독 정보를 관리하는 구독 메타 스토어를 포함한다.

[0021] 또한, 상기 액세스 제어기는, 클러스터 구성, 연결 정보, 드라이버 및 보안 프로토콜 파라미터와 같은 구성 파라미터를 유지 관리하는 연결 구성기, 상기 제2 빅데이터 저장소의 스키마 구조를 유지하는 스키마 핸들러, 상기 제2 빅데이터 저장소에서 지속되는 데이터에 대한 생성, 읽기, 업데이트 및 삭제 작업을 수행하는 CRUD 오퍼레이터(Create, Read, Update, Delete Operator), 그리고 대량의 데이터를 로드하고 분산 테이블에 동일하게 유지하기 위한 읽기-쓰기 작동을 제공하는 수동 액세스기를 포함한다.

[0022] 또한, 상기 비즈니스 로직은, 새로운 사용자 계정 생성, 액세스 역할 할당 및 세션 관리와 같은 모든 사용자 관련 작업을 캡슐화하는 사용자 관리기, 사용자가 상기 지능형 비디오 빅데이터 분석 시스템의 배포 클라우드에서 데이터를 관리할 수 있게 하는 데이터 소스 관리기, 새로운 IVA 알고리즘 및 서비스의 관리, 개발 및 지원을 수행하는 IVA 알고리즘 및 서비스 관리기, 개발자가 의사 결정을 위한 중간 결과를 얻을 수 있게 하는 온톨로지 데이터 관리기, 그리고 시스템 관리자가 상기 지능형 비디오 빅데이터 분석 시스템의 상태와 기능을 모니터링할 수 있게 하는 클러스터 관리기를 포함한다.

[0023] 본 발명의 다른 측면에 따르면, 지능형 비디오 빅데이터 분석 방법이 제공되며, 이 방법은,

[0024] 비디오 빅데이터 분석 시스템이 클라우드에서 지능적으로 비디오 빅데이터를 분석하는 방법으로서, 스피드 계층 실행 시나리오에 따라, 외부 비디오 스트림 소스에 대한 인터페이스를 제공하는 제1 클러스터를 통해 비디오 스트림을 획득하는 단계, 획득되는 비디오 스트림을 브로커 클러스터인 제2 클러스터의 RIVA_ID 토픽에 로드하는 단계, 그리고 실시간 지능형 비디오 분석(RIVA) 서비스를 사용하면서 실시간으로 비디오 스트림을 처리하는 제3 클러스터가 상기 제2 클러스터의 RIVA_ID 비디오 스트림에 대해 RIVA 서비스를 제공하는 단계를 포함하며, 상기 RIVA 서비스를 제공하는 단계에서, 상기 RIVA 서비스 중에 생성되는 중간 결과는 상기 제2 클러스터의 RIVA_IR_ID 토픽으로 전달되고, 중간 결과를 저장 관리하는 제4 클러스터가 상기 제2 클러스터의 RIVA_IR_ID 토픽에서 중간 결과를 지속적으로 읽어서 저장 관리한다.

[0025] 여기서, 상기 제2 클러스터의 RIVA_ID 토픽에 로드하는 단계 후에, 비디오 스트림을 저장하는 제5 클러스터가 상기 제2 클러스터의 RIVA_ID 토픽에서 비디오 스트림을 읽는 단계, 상기 비디오 스트림에서 메타 데이터를 처리하는 단계, 그리고 상기 비디오 스트림을 제1 빅데이터 저장소에 저장하고, 상기 메타 데이터를 제2 빅데이터 저장소에 각각 저장하는 단계를 더 포함한다.

[0026] 또한, 상기 RIVA 서비스를 제공하는 단계에서, 상기 RIVA 서비스 중에 이상 현상이 발생되는 경우, 발생된 이상 현상을 상기 제2 클러스터의 RIVA_A_ID 토픽으로 전달하고, 이상 발생을 통지하는 제6 클러스터가 상기 제2 클러스터의 RIVA_A_ID 토픽에서 이상 현상을 읽어서 상기 제2 빅데이터 저장소에 저장하고 웹 서비스를 통해 상기

비디오 스트림 소스의 소유자에게 실시간으로 전달한다.

[0027] 본 발명의 또 다른 측면에 따르면, 지능형 비디오 빅데이터 분석 방법이 제공되며, 이 방법은,

[0028] 비디오 빅데이터 분석 시스템이 클라우드에서 지능적으로 비디오 빅데이터를 분석하는 방법으로서, 배치(batch) 계층 실행 시나리오에 따라, 배치 비디오 획득 서비스를 제공하는 제1 클러스터를 통해 외부 배치 데이터를 획득하는 단계, 획득되는 배치 데이터를 처리하여 배치 비디오 데이터와 메타 데이터를 추출하는 단계, 상기 배치 비디오 데이터를 제1 빅데이터 저장소에 저장하고, 상기 메타 데이터를 제2 빅데이터 저장소에 저장하는 단계, 그리고 배치 지능형 비디오 분석(Batch Intelligent Video Analytics, BIVA) 서비스를 사용하는 제2 클러스터가 상기 제1 빅데이터 저장소에 저장된 배치 비디오 데이터에 대해 BIVA 서비스를 제공하는 단계를 포함하며, 상기 BIVA 서비스를 제공하는 단계에서, 상기 BIVA 서비스 중에 생성되는 중간 결과는 상기 제2 빅데이터 저장소에 지속적으로 저장되고 다시 상기 제2 빅데이터 저장소로부터 읽혀져서 상기 BIVA 서비스 제공시 사용된다.

[0029] 여기서, 상기 BIVA 서비스를 제공하는 단계 전에, 모델 저장 서비스를 제공하는 제3 클러스터를 통해 외부 배치 모델 데이터를 획득하는 단계, 획득되는 배치 모델 데이터를 처리하여 모델 데이터와 파일 메타 데이터를 추출하는 단계, 그리고 상기 모델 데이터를 상기 제1 빅데이터 저장소에 저장하고, 상기 파일 메타 데이터를 상기 제2 빅데이터 저장소에 저장하는 단계를 더 포함하고, 상기 IVA 서비스를 제공하는 단계에서, 상기 제2 클러스터가 상기 제1 빅데이터 저장소에 저장된 배치 비디오 데이터와 모델 데이터에 대해 상기 BIVA 서비스를 제공한다.

[0030] 또한, 상기 BIVA 서비스를 제공하는 단계에서, 상기 BIVA 서비스 중에 이상 현상이 발생되는 경우, 발생된 이상 현상을 상기 제2 빅데이터 저장소에 전달하여 저장한다.

발명의 효과

[0031] 본 발명에 따르면, IVA 수명 주기 동안 IR을 효율적으로 관리할 수 있다.

[0032] 또한, IVA 알고리즘을 쉽게 파이프라인하여 도메인별 IVA 서비스를 생성할 수 있다.

[0033] 또한, 데이터 소스 독립적인 시스템으로의 구현으로 더 확장 가능하고, IoT와 호환된다.

[0034] 또한, 도메인 전문가가 IVA 서비스에 정의한 규칙에 따라 비디오 스트림을 모니터링하는 LVSM을 통해 촉진된다.

[0035] 또한, 분산 빅데이터 지속성은 방대한 양의 원시 비디오 데이터 모델 구조화된 데이터 IR을 지원하기 위해 제공되며 이를 통해 시스템은 데이터 기반 지식 생성 기술 및 예측 분석 시각화를 지원할 수 있다.

도면의 간단한 설명

[0036] 도 1은 본 발명의 실시예에 따른 지능형 비디오 빅데이터 분석 시스템의 개략적인 블록도이다.

도 2는 도 1에 도시된 RVSAS의 구체적인 구성을 도시한 도면이다.

도 3은 도 2에 도시된 VSAS 및 VSP의 작동 예를 도시한 도면이다.

도 4는 본 발명의 실시예에 따른 비디오 스트림 획득과 동기화의 흐름을 도시한 도면이다.

도 5는 도 1에 도시된 ISBDS의 구체적인 구성을 도시한 도면이다.

도 6은 도 1에 도시된 DPBDS의 구체적인 구성을 도시한 도면이다.

도 7은 도 6에 도시된 DPBDS의 사용자 공간의 계층 구조를 도시한 도면이다.

도 8은 도 1에 도시된 비즈니스 로직의 구체적인 구성을 도시한 도면이다.

도 9는 본 발명의 실시예에 따른 사용자 역할 및 사용 사례를 도시한 도면이다.

도 10은 본 발명의 실시예에 따른 지능형 비디오 빅데이터 분석 방법의 개략적인 흐름도이다.

도 11은 본 발명의 실시예에 따른 지능형 비디오 빅데이터 분석 시스템의 클러스터 예를 도시한 도면이다.

도 12는 본 발명의 실시예에 따른 지능형 비디오 빅데이터 분석 시스템의 VSAS 및 VSP의 성능 테스트 결과를 도시한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0037] 아래에서는 첨부한 도면을 참고로 하여 본 발명의 실시예에 대하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다

[0038] 명세서 전체에서, 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다. 또한, 명세서에 기재된 "…부", "…기", "모듈" 등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어나 소프트웨어 또는 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다.

[0039] 명세서 전체에서, "및/또는"이라는 용어는 관련된 객체를 기술하기 위한 연관 관계만을 설명하고 세 가지 관계가 존재할 수 있음을 나타낸다는 것이 이해되어야 한다. 예를 들어, A 및/또는 B는 A만 존재하고, A와 B가 모두 존재하며, B만 존재하는 세 가지 경우를 나타낼 수 있다. 또한, 명세서 전체에서 문자 "/"는 일반적으로 연관된 객체 사이의 "또는" 관계를 지시한다.

[0040] 본 발명에서 설명하는 장치들은 적어도 하나의 프로세서, 메모리 장치, 통신 장치 등을 포함하는 하드웨어로 구성되고, 지정된 장소에 하드웨어와 결합되어 실행되는 프로그램이 저장된다. 하드웨어는 본 발명의 방법을 실행할 수 있는 구성과 성능을 가진다. 프로그램은 도면들을 참고로 설명한 본 발명의 동작 방법을 구현한 명령(instructions)을 포함하고, 프로세서와 메모리 장치 등의 하드웨어와 결합하여 본 발명을 실행한다.

[0041] 이하, 도면을 참조하여 본 발명의 실시예에 따른 클라우드에서의 지능형 비디오 빅데이터 분석을 위한 시스템에 대해 설명한다.

[0042] 도 1은 본 발명의 실시예에 따른 지능형 비디오 빅데이터 분석을 위한 시스템의 개략적인 블록도이다.

[0043] 도 1에 도시된 바와 같이, 본 발명의 실시예에 따른 지능형 비디오 빅데이터 분석을 위한 시스템(10)은 실시간 비디오 스트림 획득 및 동기부(Real-Time Video Stream Acquisition and Synchronization, RVSAS)(100), 즉시 구조화된 빅데이터 저장소(Immediate Structured Big Data Store, ISBDS)(200), 분산 지속 빅데이터 저장소(Distributed Persistent Big Data Storage, DPBDS)(300), 비즈니스 로직(Business Logic)(400) 및 웹 서비스(Web Service)(500)를 포함한다.

[0044] RVSAS(100)는 외부, 예를 들어 외부의 실시간 비디오 수집 도구로부터 수집되는 실시간 비디오 스트림을 수집하여 처리한다.

[0045] RVSAS(100)는, 예를 들어 분산 메시징 시스템(Distributed Messaging System)(101)인 아파치 카프카(Apache Kafka)를 활용하여 생산자의 클러스터에서 미니 배치 형식으로 비디오 스트림을 수집하고, 카프카 브로커(Kafka Broker)에서 버퍼링한 다음 미니 배치를 사용자 클러스터로 라우팅한다.

[0046] ISBDS(200)는 분산 환경에서 대규모의 구조화된 데이터를 관리한다. 데이터 집약적인 작업과 다른 계층의 요구사항에 따라 아파치 HBase라는 분산 NoSQL 빅데이터 저장소를 지원한다.

[0047] DPBDS(300)는 HDFS 위에 구축되며, 영구적이고 분산된 빅데이터 저장소를 제공한다.

[0048] 비즈니스 로직(400)은 RVSAS(100), ISBDS(200) 및 DPBDS(300)에 대한 높은 수준의 추상화이며, 본 발명의 시스템(10)의 설계 철학에 따라 데이터 액세스를 사용자가 정의하기 위해 제공된다.

[0049] 웹 서비스(500)는 본 발명의 실시예에 따른 시스템(10) 프레임워크의 기능을 웹을 통해 제공하기 위해 최고의 기능을 단순한 통합 역할 기반 웹 서비스에 통합한다. 이러한 웹 서비스(500)는 전술한 비즈니스 로직(400)을 기반으로 구축된다.

[0050] 도 2는 도 1에 도시된 RVSAS(100)의 구체적인 구성을 도시한 도면이다.

[0051] 도 2를 참조하면, RVSAS(100)는 브로커 클라이언트 서비스(Broker Client Services, BCS)(110), 비디오 스트림 획득 서비스(Video Stream Acquisition Service, VSAS)(120), 비디오 스트림 프로듀서(Video Stream Producer, VSP)(130), 중간 결과 관리기(Intermediate Result-Manager, IR-Manager)(140), 및 라이프롱 비디오 스트림 모니터(Lifelong Video Stream Monitor, LVSM)(150)를 포함한다.

[0052] BCS(110)는 아파치 카프카(Apache Kafka)를 활용하여 생산자의 클러스터에서 미니 배치 형식으로 비디오 스트림을 수집하고, 카프카 브로커(Kafka Broker)에서 버퍼링한 다음 미니 배치를 사용자 클러스터로 라우팅한다. BCS(110)는 대규모 실시간 비디오 스트림 관리를 위한 카프카 브로커를 관리하기 위해 제안된 프레임워크의 비즈니스 로직(400)에 따라 비디오 스트림 수집을 수행한다.

[0053] 이러한 BCS(110)는 토픽 관리기(Topic Manager)(111), 파티션 관리기(Partition Manager)(112) 등을 포함한다.

[0054] 토픽 관리기(111)는 새로운 RIVA(Real-time IVA) 서비스 생성시 카프카 브로커 클러스터에서 동적으로 새로운 카프카 토픽을 생성하는 데 사용된다. 새로운 RIVA 서비스가 생성되면 카프카 브로커 클러스터에서 RIVA_ID, RIVA_IR_ID, RIVA_A_ID라는 이름 규칙을 사용하여 세 가지 유형의 카프카 토픽이 자동으로 생성된다. 여기서, ID는 서비스의 고유 식별자이다. 이러한 항목은 실제 비디오 스트림 IR 및 IVA 서비스에서 감지한 이상 현상을 감지하는 데 사용된다.

[0055] 토픽 관리기(111)는 관리자 역할이 모든 토픽을 나열하고 해당 구성을 확인하고 구성을 재정의할 수 있도록 허용한다. 각 토픽은 비디오 스트림 레코드가 배포되는 파티션으로 구성된다. 일반적으로, 병렬 처리 정도와 높이 처리량은 토픽의 파티션 개수에 비례한다.

[0056] 파티션 관리기(112)는 비디오 스트림 데이터 소스의 비디오 스트림과 소비자 그룹의 스트림 소비에 따라 토픽의 파티션 개수를 자동으로 조정한다.

[0057] 다음, VSAS(120)는 클라이언트 API(Application Programming Interface)로서, 대규모 비디오 스트림을 취하기 위한 생산자의 클러스터에서 구성될 수 있다.

[0058] 이러한 VSAS(120)는 도 3을 참조하면, 비디오 스트림 획득 어댑터(Video Stream Acquistion Adapter)(121), 비디오 스트림 이벤트 프로듀서(Video Stream Event Producer)(122), 프레임 전처리기(Frame Preprocessor)(123) 및 레코드 구성기(Record Composer)(124)를 포함한다.

[0059] 비디오 스트림 획득 어댑터(121)는 장치 독립적인 비디오 스트림 소스를 위한 인터페이스를 제공한다. 특정 비디오 소스가 RIVA 서비스에 대해 구독되는 경우 스트림 획득 서비스는 ISBDS(200)의 비디오 데이터 소스(VDS)에서 구성 메타 데이터를 가져오고 비디오 스트리밍을 위해 소스 장치를 구성한다.

[0060] 비디오 스트림 이벤트 프로듀서(122)는 비디오 스트림을 디코딩하고 프레임을 감지한 다음 메타 데이터 추출을 수행한다.

[0061] 프레임 전처리기(123)는 추출된 메타 데이터에 대해 프레임 크기 조정을 수행한다.

[0062] 레코드 구성기(124)는 메타 데이터에 대해 비디오 스트림 소스와 통신하기 위해 JSON(JavaScript Object Notation) 개체를 정의한다. 이 개체의 내용은 데이터소스 ID, 프레임의 열(column) 개수 및 행(row) 개수, 데이터 소스에서 데이터가 시작된 타임 스탬프 및 페이로드가 5개 필드로 구성된다. 이러한 JSON 객체는 레코드로서, VSP(130)로 전달된다.

[0063] VSP(130)도 클라이언트 API로서, VSAS(120)에서 레코드를 수신하고 수신된 레코드를 직렬화하고 미니 배치를 구성한 다음 카프카 브로커로 보내는 작동을 수행한다.

[0064] 비디오 스트림 소스가 RIVA 서비스에 등록된 경우 프로듀서 핸들러(170)는 미니 배치를 브로커 클러스터의 토픽 RIVA_1로 라우팅한다. 마찬가지로, 여러 RIVA 서비스에 가입한 경우 스트림은 해당 토픽으로 라우팅된다.

[0065] VSP(130)에서의 카프카 프로듀서 레코드는 기본 메시지 형식을 가지며 4개의 필드, 즉, 토픽 이름(Topic Name), 파티션 번호(Partition Number), 키(Key) 및 값(Value)으로 구성된다. 여기서, 토픽 이름은 ISBDS(200)에 저장된 가입 데이터 소스에 의해 동적으로 제공된다. 파티션 및 키 필드는 비디오 스트림 소스 ID(ISBDS(200)의 비디오 데이터 소스 메타 스토어에서 제공된 카메라 ID)로 설정된다. 값 필드 입력은 레코드 구성기(124)에 의해 구성되어 제공되는 비디오 스트림 레코드를 나타낸다.

[0066] VSP(130)는 네트워크를 통해 보낼 수 있도록 프로듀서 레코드를 바이트 배열로 변환하기 위해 프로듀서 핸들러(170)의 직렬 변환기(171)로 보내진다. VSP(130)는 프로듀서 레코드를 수집하여 카프카 브로커로 보내기 위해 레코드 배치에 추가한다.

[0067] 빠른 압축 알고리즘이 사용되어 이러한 미니 배치가 압축되고, VSP(130)는 새로 형성된 미니 배치를 브로커 서

버로 보낸다. 메시지 전달을 보장하기 위해 VSP(130)의 백(BACK)이 활성화된다.

[0068] 전술한 VSAS(120) 및 VSP(130)의 서브 모듈들은 아래의 알고리즘과 같이 videoStreamEventProducer 고급 API에 기술적으로 캡슐화될 수 있다.

```
Input  :topicName, cameraID, cameraURL
Output:Sent compressed video streams to the Broker Server (Topic RIVA_ID)
/* Initialization                                                    */
static loadLibrary(Core.NATIVE_LIBRARY_NAME)
VidFrameMessage vFrameMsg.set(topicName, cameraID, cameraURL)
/* Create Kafka Producer.  Parameters defined in cluster configurator.  */
Kafka producer ← ProducerCreator.createProducer()
Mat mat <  new Mat()
Gson gson ← new Gson()
/* Testing camera                                                    */
VideoCapture camera ← null;
if StringUtils.isNumeric(videoFrameMessage.getCameraURL) then
    camera ← new VideoCapture(Integer.parseInt(vFrameMsg.getCameraURL()));
else
    camera ← new VideoCapture(vFrameMsg.getCameraURL());
end
EventGeneratorCallback(cameraID) /* Callback to confirm message delivery  */
while true do
    /* Frame resize                                                  */
    Imgproc.resize(mat, mat, newSize(480,320),0,0, INTER_CUBIC)
    /* Set the changing attributed from camera stream                */
    vFrameMsg.setCols(mat.cols(), mat.rows(), mat.type())
    vFrameMsg.setData(new byte[(int) (mat.total() * mat.channels())])
    mat.get(0, 0, vFrameMsg.getData())
    vFrameMsg.setTimestamp(Timestamp(currentTimeMillis()))
    jsonObject.addProperty("cameraId", "timestamp", "rows", "cols", "type")
    jsonObject.addProperty("data", encodeToString(vFrameMsg.getData()))
    String jsonString ← gson.toJson(jsonObject)
    JsonObject jsonObject ← new JsonObject()
    /* Composing VSP Record                                          */
    ProducerRecord<String, String> messageData ← new ProducerRecord<String, String>(
    videoFrameMessage.getTopicName(), /* send to topic               */
    videoFrameMessage.getCameraId(), /* key                          */
    jsonString) /* Value                                             */
    /* Send the message to the Broker Server (topic RIVA_ID)         */
    producer.send(messageData, eventGeneratorCallback)
end
```

[0069]

[0070] 도 4를 참조하면, 획득된 비디오 스트림은 이제 미니 배치의 형태로 다양한 토픽의 카프카 브로커(161)에 상주한다. 비디오 스트림의 이러한 미니 배치를 처리하기 위해 비디오 스트림 분석 소비자 클러스터(Video Stream Analytics Consumer Cluster)(163)라는 다양한 컴퓨터 클러스터가 있다. 각각의 클러스터에는 RIVA 서비스(1631)와 VSCS(Video Stream Consumer Service, VSCS)(1633)의 두 가지 유형의 클라이언트 서비스가 구성된다. 각각의 비디오 스트림 분석 소비자 클러스터(163)에는 서로 다른 도메인별 RIVA 서비스(1631)가 있는 반면, VSCS(1633)는 모두 공통이다. VSCS(1633)는 분석을 위해 카프카 브로커(161)의 각각의 토픽에서 비디오 스트림의 미니 배치를 읽는 RIVA 서비스를 지원한다.

[0071] RIVA 서비스(1631)는 IR을 생성하고 지속되어야 한다. 비디오 분석은 항상 비용이 많이 들고 재계산을 피하기 위해 IR을 ISBDS(200)(IR 미들웨어)로 유지한다.

[0072] IR-manager(140)는 카프카 브로커 클러스터(141)의 토픽 RIVA_IR_ID와 IR을 주고받아서, 분산된 환경에서 적절한 관리를 수행한다. IR-manager(140)는 해당 토픽에서 IR을 읽는 역할도 하며 재컴파일을 피하는 재사용(예를 들어, 비디오 온톨로지에 대한 매핑, 검색을 위한 인덱싱)을 위해 ISBDS(200)의 IR 데이터 저장소에 유지된다.

[0073] 도메인별 RIVA 서비스는 이상 또는 비정상적인 활동에 대한 비디오 스트림을 처리한다.

[0074] LVSM(150)은 이상이 감지되면 동일한 내용을 카프카 브로커 토픽, 즉(RIVA_A_ID)로 보낸다. 알림 기반 응답을 생성하기 위해 LVSM(150)은 표준 관찰자 기반 구현(standard observer-based implementation)을 따른다. 이러한 접근 방식을 기반으로 LVSM(150)은 해당 카프카 브로커 토픽에서 이상 현상을 읽는다. 즉, RIVA_A_ID이며, 거의 실시간으로 클라이언트에게 알리고 동시에 ISBDS에 유지한다. VSAS(1433), IR-Manager)(140) 및 LVSM(150)을 이해하기 위해 템플릿 서비스를 만든다(이하의 알고리즘 참조).

```
Input  :serviceID, cameraID
Output: Produce IR, and Anomalies to topic RIVA_IR_ID, and RIVA_A_ID, respectively
/* Initialization                                                          */
initiate Spark session
/* Schema structure definition                                             */
StructureType Schema ← DataTypes.createStructType(CameraID, timestamp, rows, cols, type,
  data)
/* VSCS abstraction to acquire mini-batches from topic RIVA_ID.            */
Dataset< VideoEventData > videoDataset ← Vscs.readVidStream(serviceID, camerID,
  Schema)
/* RIVA service logic written by developer.                                */
RIVA logic goes here...
/* IR abstraction produces IR to the topic RIVA_IR_ID                      */
IntResults.irProduce(IR-Message, serviceID, camID)
/* Condition for anomalies                                                 */
if True then
    /* LVSM abstraction produces Anomalies to topic RIVA_A_ID              */
    Lvsm.produceAnomalies(LVSM-Message, serviceID, camID)
end
/* Spark API to start the video stream continuous acquisition from the topic
   RIVA_ID                                                                 */
StreamingQuery query = processedDataset.start()
query.awaitTermination() /* Wait untill termination.                       */
```

[0075]

[0076] 이러한 서비스는 아파치 스파크 스트리밍을 기반으로 한다. RIVA 서비스에 대한 입력은 serviceID 및 구독된 cameraID이다. 비디오 스트림 미니 배치는 VSAS 상위 수준 추상화를 사용하여 토픽 RIVA_ID에서 획득하고 4단계에서와 같이 videoDataset(스파크 데이터 세트 추상화의 객체)에 할당된다. RIVA 작업(특별 또는 임시 분석)은 개발자의 논리와 요구사항에 따라 videoData에서 수행된다. IR 및 비정상 형태의 RIVA 출력은 각각 IRM(140) 및 LVSM(150) API로 전달된다.

[0077] 도 5는 도 1에 도시된 ISBDS(200)의 구체적인 구성을 도시한 도면이다.

[0078] 도 5를 참조하면, ISBDS(200)는 5가지 유형의 데이터를 호스팅한다. 5가지 유형의 데이터로는 사용자 프로파일 및 로그(User Profile and Logs)(210), 비디오 데이터 소스 메타 스토어(Video Data Source Meta-store)(220), 알고리즘 및 서비스 메타 스토어(Algorithm and Services Meta Store)(230), 중간 결과 미들웨어(Intermediate Result Middleware)(240) 및 구독 메타 스토어(Subscription Meta-store)(250)가 포함된다.

[0079] 첫 번째로, 사용자 프로파일 및 로그(210)는 사용자에게 역할 기반 액세스를 제공한다. 사용자 로그 및 해당 역할 정보는 사용자 카탈로그를 통해 유지된다. 모든 민감한 사용자 정보는 MD5 암호화 체계를 지원하여 안전하게 보호된다.

[0080] 두 번째로, 비디오 데이터 소스 메타 스토어(220)는 두 가지 유형의 비디오 소스, 즉 비디오 스트림 소스 및 비디오 데이터 세트를 관리한다. 전자는 RIVA 서비스에 가입할 수 있고, 후자는 BIVA(Batch IVA) 서비스에 가입할 수 있다. 비디오 데이터 소스 메타 스토어는 이러한 소스의 메타 정보를 액세스 권한과 함께 관리한다.

[0081] 세 번째로, 알고리즘 및 서비스 메타 스토어(230)는 IVA 알고리즘과 서비스를 관리한다. 특히, IVA 알고리즘의 메타 데이터는 알고리즘 메타 스토어를 통해 관리되며, 데이터 구조는 다음과 같은 8개의 서로 다른 데이터 필드로 구성된다. 즉, AID, UID, algoType, algoName, input, output, requiredResources, sourceCodeAddress, description. IVA 서비스의 메타 데이터는 서비스 메타 스토어를 통해 관리되며, SID, UID, dsTypeID, serviceTypeID, serviceName 및 설명과 같은 속성을 포함한다. 서비스와 알고리즘 메타 스토어 간의 관계는 SERVICE_ALGORITHM이라는 룩업 테이블을 통해 관리된다.

[0082] 네 번째로, 중간 결과 미들웨어(240)는 중간 결과를 저장하는 데 사용된다.

[0083] 다섯 번째로, 구독 메타 스토어(250)는 사용자가 RIVA 서비스에 대한 비디오 스트림 소스를 구독하는 구독 정보를 관리한다.

[0084] ISBDS(200)는 ISBDS 액세스 제어기(Access Controller)(260)를 포함한다. ISBDS 액세스 제어기(260)는 비즈니스 로직(400)에 따라 기본 데이터에 대한 읽기-쓰기 액세스를 안전하게 제공하는 역할을 한다.

[0085] ISBDS 액세스 제어기(260)는 연결 구성기(Connection Configurator)(261), 스키마 핸들러(Schema Handler)(262), CRUD 오퍼레이터(Create, Read, Update, Delete Operator)(263) 및 수동 DRW(Passive Data

Reader, Writer)(264)를 포함한다.

[0086] 연결 구성기(261)는 클러스터 구성, 연결 정보, 드라이버 및 보안 프로토콜 파라미터와 같은 구성 파라미터를 유지 관리하는 데 사용된다. 연결 구성기(261)는 데이터 액세스 요청이 있을 때 분산 데이터 저장소(Distributed Big Data Store)(270)와의 연결 세션을 설정하고 유지하는 데 사용된다.

[0087] ISBDS(200)의 기본 스키마 구조는 스키마 핸들러(262)를 통해 유지되며, 이식성을 위해서도 동일하게 사용할 수 있다.

[0088] CRUD 오퍼레이터(263)는 ISBDS(200)에 지속되는 데이터에 대한 CRUD 작업을 위해 설계 및 제공된다. CRUD 오퍼레이터(263)는 주로 분산 데이터 저장소(270)를 통해 대기 시간이 짧은 실시간 대화형 CRUD 작업을 제공하도록 설계된다. CRUD 오퍼레이터(263)의 인스턴스는 분산 RIVA 엔진과 비즈니스 로직(400)에서 사용된다. CRUD 작업에 대해 미리 정의된 일반화된 쿼리가 호스팅된다. 클라이언트에 대한 요청을 받으면 CRUD 오퍼레이터(263)는 요청 파라미터를 사용 가능한 쿼리 세트와 일치시킨다. 파라미터가 일치하는 경우 선택한 쿼리가 클라이언트에 실시간으로 응답하여 실행된다. 또한, 아파치 스파크와 같은 분산인 메모리 처리 엔진을 사용하면서 대량의 비디오에 대한 오프 라인 분석을 지원하도록 설계되었다.

[0089] 수동 DRW(264)는 아파치 스파크가 대량의 데이터를 RDD(RDD는 스파크 데이터 구조임)로 로드하고 필요에 따라 분산 테이블에 동일하게 유지할 수 있도록 제공된다. 수동 DRW(264)의 실행은 2단계 프로세스이다. 먼저. 클라이언트 응용 프로그램에서 스키마 파라미터를 선택한 다음 수동 DRW(264)에게 제출한다. 수동 DRW(264)는 파라미터 수신시 동적으로 쿼리를 생성한다. 그런 다음, 쿼리가 NoSQL 분산 데이터 저장소(270)를 통해 실행되고 응답이 생성되어 클라이언트 프로그램에 제공된다. 수동 DRW(264)는 오프라인 분석 요구사항을 충족하기 위해 제공되며, 요청된 데이터 크기에 따라 쿼리 실행 시간이 더 오래 걸릴 수 있다.

[0090] 도 6은 도 1에 도시된 DPBDS(300)의 구체적인 구성을 도시한 도면이다.

[0091] 도 6을 참조하면, 데이터는 본 발명의 시스템(10)의 비즈니스 로직(400)에 따라 체계적으로 저장되고 매핑된다. DPBDS(300)는 IVA 알고리즘의 수명 주기에서 데이터 워크로드를 효과적이고 동적으로 관리하도록 설계되었다. 본 발명의 시스템(10)에 새로 등록하면 HDFS(또는 하둡)(301) 상단에 정식 사용자 공간(310)이 생성된다. 사용자 공간(310)은 적절한 계층적 디렉토리 구조를 통해 관리되며 소유자에게 특별한 읽기 및 쓰기 액세스 권한이 부여된다. 모든 디렉토리는 사용자 프로필 로그의 사용자 ID에 따라 동기화되고 매핑된다. HDFS에서는 각 사용자 공간(310) 아래에 도 7과 같이 세 가지 유형의 분산 디렉토리가 생성된다. 즉, 원시 비디오 공간(Raw Video Space)(320), 모델 공간(Model Space)(330) 및 프로젝트 공간(Project Space)(340)이 그것이다. DPBDS(300)는 사용자 공간(310)과 DPBDS(300)의 해당 데이터를 관리 및 운용하기 위해 DPB 액세스 제어기(Access Controller)(350)를 포함하며, 이는 HDFS 핸들러(Handler)(351), 능동 DRW(Active DRW)(352) 및 수동 DRW(Passive DRW)(353)를 포함한다. HDFS 핸들러(351)는 비즈니스 로직(400)에 따라 파일 작업 및 권한 제어를 관리하기 위해 HDFS의 저수준 API를 통해 설계되었다. 능동 DRW(352) 및 수동 DRW(353)는 실시간 및 오프라인 읽기-쓰기 작업을 수행한다.

[0092] 전술한 RVSAS(100), ISBDS(200) 및 DPBDS(300)는 고급 IVA 알고리즘을 활용하면서 클라우드에서 대규모 비디오 데이터 수집 및 도메인별 IVA를 지원하도록 설계되었다.

[0093] 도 8은 도 1에 도시된 비즈니스 로직(400)의 구체적인 구성을 도시한 도면이다.

[0094] 도 8을 참조하면, 비즈니스 로직(400)은 향후 요구사항에 따라 확장 가능하지만, 본 발명의 실시예에서는 사용자 관리기(User Manager)(410), 데이터 소스 관리기(Data Source Manager)(420), IVA 알고리즘 및 서비스 관리기(IVA Algorithm and Service Manager)(430), 온톨로지 데이터 관리기(Ontology Data Manager)(440) 및 클러스터 관리기(Cluster Manager)(450)를 포함한다.

[0095] 사용자 관리기(410)는 새로운 사용자 계정 생성, 액세스 역할 할당 및 세션 관리와 같은 모든 사용자 관련 작업을 캡슐화한다. 본 발명의 실시예에 따른 시스템(10)은 사용자에 대한 역할 기반 액세스를 제공하며 도 9에서와 같이 세 가지 유형의 사용자 계정, 즉 admin, developer, consumer를 제공한다. 사용자 역할 및 액세스 권한은 도 9를 참조하며, 사용자 데이터 액세스 로직이 사용자 관리기(410)에서 구현된다.

[0096] 데이터 소스 관리기(420)는 사용자가 본 시스템(10)의 배포 클라우드에서 데이터를 관리할 수 있게 한다. 사용자는 비디오 스트림 소스, 비디오 데이터 세트 및 모델의 세 가지 유형의 데이터 소스를 관리할 수 있다.

[0097] IVA 알고리즘 및 서비스 관리기(430)는 스파크를 기반으로 새로운 IVA 알고리즘 및 서비스를 관리, 개발 및 지

원한다. IVA 알고리즘은 개발자에게 IVAAaaS(IVA-Algorithm-as-a-Service)와 같은 aaS를 제공하고, 서비스 관리는 소비자에게 IVAaaS(IVA-as-a-Service)를 제공한다. 개발자는 새로운 IVA 알고리즘을 만들고 게시할 수 있다. 이러한 IVA 알고리즘은 다른 개발자가 사용할 수 있도록 IVAAaaS로 제공된다. 관리자/개발자가 IVA 서비스를 만들고 게시하면 사용자는 제공된 IVA 서비스에 대해 비디오 소스를 구독할 수 있다.

[0098] 온톨로지 데이터 관리기(440)는 개발자가 의사 결정을 위한 중간 결과(IR)를 얻을 수 있게 한다. 온톨로지 데이터 관리기(440)는 IR을 얻는 안전한 방법을 제공하고, 지식 큐레이션 계층(Knowledge Curation Layer, KCL)의 비디오 온톨로지에 따라 매핑한다.

[0099] 마찬가지로, 클러스터 관리기(450)는 시스템 관리자가 시스템(10)의 상태와 기능을 모니터링할 수 있도록 한다.

[0100] 마지막으로, 웹 서비스(500)는 본 발명의 실시예에 따른 시스템(10) 프레임워크의 기능을 웹을 통해 제공하기 위해 최고의 기능을 단순한 통합 역할 기반 웹 서비스에 통합한다. 이러한 웹 서비스(500)는 전술한 비즈니스 로직(400)을 기반으로 구축된다.

[0101] 이하, 본 발명의 실시예에 따른 지능형 비디오 빅데이터 분석 방법에 대해 설명한다. 이러한 방법은 전술한 도 1 내지 도 9를 참조하여 설명한 지능형 비디오 빅데이터 분석 시스템(10)에 의해 수행될 수 있다.

[0102] 본 발명의 실시예에 따른 시스템(10)을 사용하면 참여 계층이 분산 RIVA 및 BIVA 서비스를 등록할 수 있다. 본 시스템(10)은 람다 아키텍처 스타일을 따르며 실행 시나리오는 두 가지 유형의 실행 시나리오를 거친다. 여기서, 두 가지 유형의 실행 시나리오는 스피드 계층 실행 시나리오(Speed Layer Execution Scenario)와 배치 계층 실행 시나리오(Batch Layer Execution Scenario)이다. 이러한 두 시나리오의 데이터는 서빙 계층으로도 알려진 ISBDS(200)과 DPBDS(300)를 포함한 BDP(Big Data Persistence)라는 공통 계층을 통해 관리된다.

[0103] 먼저, 도 10을 참조하여 스피드 계층 실행 시나리오에 대해 설명한다.

[0104] 클러스터 'P'(901)는 VSAS(120)를 호스트하고 외부 비디오 스트림 소스(903)에 대한 인터페이스를 제공한다. 구성시 비디오 스트림은 클러스터 'K'(905)의 해당 RIVA_ID(907)에 로드된다.

[0105] 클러스터 'K'(905)는 획득한 비디오 스트림, IR 및 LVSM(150)에 의해 생성된 이상 현상이 버퍼링되는 분산 메시징 시스템(101)인 카프카를 지원한다. 여기서, 클러스터 'K'(905)는 RIVA_ID(907), RIVA_IR_ID(909) 및 RIVA_A_ID(911)로 구성된다. 이러한 토픽은 높은 처리량을 보장하기 위해 클러스터 'K'(905)에 액세스하도록 복제된다. RF(Replication Factor)는 클러스터 'K'(905)의 작업자 수 보다 작거나 같아야 한다.

[0106] 분산 메시징 시스템(101)의 브로커 클러스터(905)의 토픽 RIVA_ID(907)에 있는 비디오 스트림의 미니 배치는 DPBDS(300) 및 ISBDS(200) 데이터 저장소에 유지되어야 한다. 이를 위해 클러스터 'V'(913)는 세 가지 유형의 모듈, 즉 비디오 스트림 소비자(Video Stream Consumer, VCS) 모듈(915), 비디오 전처리기(Video Preprocessor) 모듈(917) 및 지속성(Persistence) 모듈(919)을 전개한다.

[0107] VCS 모듈(915)을 사용하면, 클러스터 'V'(913)가 클러스터 'K'(905)의 RIVA_ID(907) 토픽에서 비디오 스트림 미니 배치를 읽을 수 있다. 그런 다음, 비디오 전처리기 모듈(917)은 소비된 비디오 데이터에서 메타 데이터를 처리, 인코딩 및 추출한다. 마지막으로, 지속성 모듈(919)은 비디오 데이터와 해당 메타 데이터를 각각 DPBDS(300) 및 ISBDS(200)에 저장한다.

[0108] 한편, 클러스터 'S'(921)는 RIVA 서비스를 사용하면서 거의 실시간으로 비디오 스트림을 처리하는 역할을 한다. 아파치 스파크 스트림과 같은 다양한 스트림 처리 엔진을 RIVA 서비스 개발에 사용할 수 있다. 클러스터 'S'(921)는 네 가지 유형, 즉 비디오 스트림 소비자 서비스(Video Stream Consumer Service, VSCS) 모듈(923), RIVA 서비스 모듈(925), IR 생산자/가입자(IR Producer/Subscriber) 모듈(927) 및 LSVM 생산자(LSVM Producer) 모듈(929)을 지원한다.

[0109] VSCS 모듈(923)은 클러스터 'K'(905)에서 RIVA_ID 비디오 스트림(907)을 사용하는 데 사용된다.

[0110] RIVA 서비스 모듈(925)는 비디오 스트림을 분석하는 실제 RIVA 서비스이다(S915). 이러한 RIVA 서비스는 사용자가 체결한 RIVA 서비스 가입 계약에 따라 로드된다.

[0111] IR 생산자/가입자 모듈(927)은 클러스터 'K'(905)의 토픽 IR(909)과 애플리케이션 로직에 따라 IR을 송수신하는 데 사용된다.

[0112] 클러스터 'S'(921)에 배치된 RIVA 서비스 인스턴스는 일부 도메인 특정 목표를 가져야 하며 분석할 경우 이상을 생성해야 한다. 본 시스템(10)은 실시간 변칙 전달 시스템을 지원한다. RIVA 서비스는 계속해서 LVSM 생산자(929)를 통해 이상 현상을 클러스터 'K'(905)의 각 이상 항목 토픽 RIVA_A_ID(911)로 보낸다.

[0113] 클러스터 그룹 'I'(931)는 클러스터 'K'(905)의 토픽 RIVA_A_ID(909)에서 IR을 지속적으로 읽고 인덱싱을 위해 ISBDS(200)의 IR 미들웨어(240)로 전송한다.

[0114] 클러스터 'N'(933)은 이상 알림 클러스터이다. 이 클러스터 'N'(933)은 클러스터 'K'(905)의 토픽 RIVA_A_ID(911)에서 이상 현상을 읽고 지속성을 위해 ISBDS(200)에 동일한 내용을 보내고 경고의 비디오 스트림 소스 소유자에게 실시간으로 전달한다.

[0115] 다음, 다시 도 10을 참조하여 배치 계층 실행 시나리오에 대해 설명한다.

[0116] 배치 계층 서비스, 즉 BIVA 서비스는 일괄 비디오 분석에 사용할 수 있는 aaS이다. 일괄 비디오 데이터 세트는 오프라인에서 분석되며 실행 시간은 비디오 데이터 세트 크기 및 구독한 BIVA 서비스에 비례한다.

[0117] 본 발명의 실시예에 따른 배치 계층 실행 시나리오는 세 가지 유형의 클러스터, 즉 클러스터 'R'(941), 클러스터 'M'(943) 및 클러스터 'B'(945)를 거친다.

[0118] 클러스터'R'(941)을 통해 사용자는 배치 비디오 데이터 세트를 시스템(10)에 업로드하고 3가지 유형의 라이브러리, 즉 배치 비디오 획득 서비스(Batch Video Acquisition Service)(947), 비디오 전처리기(Video Preprocessor)(949) 및 배치 비디오 지속성(Batch Video Persistence)(951)를 구성할 수 있다.

[0119] 배치 비디오 획득 서비스(947)는 배치 비디오 데이터 세트를 획득하는 데 사용된다.

[0120] 배치 비디오 획득 서비스(947)에 의해 배치 비디오 데이터 세트가 노드 버퍼에 업로드되면, 활성화된 비디오 전처리기(949)에서 배치 데이터 세트를 처리하여 배치 비디오의 메타 데이터를 추출한다.

[0121] 그런 다음, 배치 비디오 데이터와 추출된 메타 데이터는 각각 DPBDS(300) 및 ISBDS(200)에 유지된다.

[0122] 마찬가지로, 클러스터 'M'(943)은 클러스터 'R'(941)과 동일한 방식으로 작동하지만 클러스터 'M'(943)은 모델 관리를 담당한다.

[0123] 일괄 비디오 분석에서 지원 계층은 BIVA 서비스를 지원한다. 클러스터 'B'(945)는 사용자 계약에 따라 BIVA 서비스 인스턴스를 로드하고, 비디오를 오프라인으로 처리한다. 일단 구독하면 이 클러스터 'B'(945)는 DPBDS(300)에서 배치 비디오 데이터 세트와 모델을 로드한다. 유사하게, IR 및 이상 현상은 ISBDS(200)에서 유지된다.

[0124] 다음, 본 발명의 실시예에 따른 지능형 비디오 빅데이터 분석 시스템(10)을 적용한 예시에 대해 설명한다.

[0125] 본 발명의 실시예의 적용은 두 가지 RIVA 서비스, 즉 얼굴 감지 및 동작 인식에 대해 수행되었다. 이러한 두 가지 애플리케이션은 aaS로 제공된다.

[0126] Hortonworks Data Platform(HDP) 버전 3.1.0을 배포하면서 클러스터라는 실내 분산 클라우드 환경을 구축하였다. 클러스터는 도 11에 도시된 바와 같이, 11개의 노드로 구성된다. 각 노드에는 5개의 파라미터가 있다. 맨 위행의 파라미터는 사용중인 운영체제 버전을 표시한다. 두 번째 줄은 각각 프로세서 모델, 코어 수, RAM 크기(GB) 및 하드 디스크 크기GB)를 보여준다. 네트워킹 목적의 고밀도 구리 연결을 위해 자동 감지 1000Mbps 인터페이스의 24개 및 48개 포트를 제공하는 ProSafe GSM7328S 완전 관리형 스위치가 사용되었다.

[0127] 사용자는 RIVA에 대한 실시간 비디오 스트림 소스를 노출할 수 있다. 등록시 두 가지 서비스, 세 가지 유형의 토픽은 다음의 [표 1]과 같이 토픽 관리기(Topic Manager)(111)에 의해 자동으로 생성된다. RF를 3으로 설정하고(브로커 서버가 3개 있으므로) 파티션 개수를 토픽당 140으로 설정한다(최대 카메라 스트림 개수 허용).

[0128] [표 1]

Topic Name	Description	RF	Partitions
RIVA_1	Consumes the stream, subscribed to Face Detection service.	3	140
RIVA_IR_1	Consumes the IR of Face Detection service.	3	140
RIVA_A_1	Consumes the anomalies of service Face Detection.	3	140
RIVA_2	Consumes the streams, subscribed to Action Recognition.	3	140
RIVA_IR_2	Consumes the IR of Action Recognition service.	3	140
RIVA_A_2	Consumes the Anomalies of Action Recognition service.	3	140

[0129]

[0130] 또한, 다음의 [표 2]와 같이 BCS(110)의 하위 모듈인 클러스터 구성기(도시되지 않음), ISBDS(200) 및 DPBDS(300)에서 서로 다른 파라미터의 값을 설정한다.

[0131] [표 2]

	Variable	Value
Kafka	ACKS_CONFIG	All
	BATCH_SIZE_CONFIG	20,971,520 Bytes
	COMPRESSION_TYPE_CONFIG	Snappy
	MAX_REQUEST_SIZE_CONFIG	2,097,152 Bytes
	LINGER_MS_CONFIG	5
	AUTO_COMMIT_INTERVAL_MS_CONFIG	1000
HDFS	Block replication	3
	HDFS Block Size	64 MB
	Java heap size	1 GB
HBase/Phoenix	phoenix.query.timeoutMs	1,800,000
	hbase.regionserver.lease.period	1,200,000
	hbase.rpc.timeout	1,200,000

[0132]

[0133] VSAS(120)는 장치 독립적이므로 서로 다른 이기종 장치와 오프라인 비디오 스트림 소스를 등록한다. 이기종 비디오 스트림 소스에는 IP 카메라, 깊이 카메라, RTSP, IPhone6s Plus가 포함된다. 기본적으로, 처음 세 데이터 소스의 프레임 속도는 30이고, 마지막 데이터 소스는 초당 60프레임이다.

[0134] 오프라인 비디오 스트림 소스의 경우 HDD(WDC WD10EZEX)에 있는 비디오 파일도 VSAS(120)의 요소로 구성되었다. VSAS(120)는 획득한 프레임의 해상도를 480 ㅅ 320 픽셀로 설정한다. 결과적으로, 획득한 각 프레임의 크기는 614.495KB가 되었다.

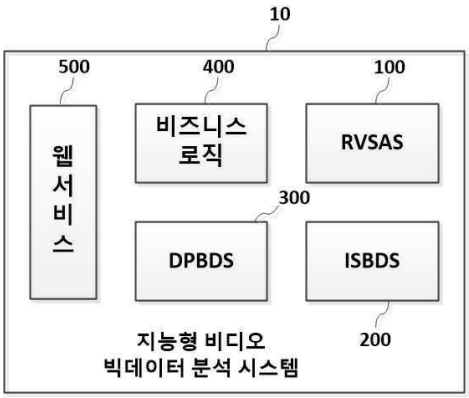
[0135] VSAS(120)는 획득한 프레임을 6MS의 속도로 형식 메시지로 변환한다. 그런 다음, VSAS(120)는 메시지를 VSP(130)로 전달한다. VSP(130)는 메시지 크기를 평균 140.538KB로 압축하고, 12MS의 속도로 브로커 서버로 전달한다. 이 두 모듈은 브로커 서버(Agent-6, Agent-7 및 Agent-8, 도 11 참조)로 스트림을 획득하고 전송하도록 구성된다. 성능 테스트 결과는 도 12에 도시된 바와 같다. 도시된 결과로부터 평균적으로 이기종 비디오 스트림 소스 및 오프라인 비디오 스트림 소스에서 각각 초당 34 및 54 프레임을 획득할 수 있음을 알 수 있다. 달성된 속도는 선호하는것 보다 36% 및 116% 더 높다. 즉 RIVA 분석의 경우 초당 25프레임이다.

[0136] 이상에서 설명한 본 발명의 실시예는 장치 및 방법을 통해서만 구현이 되는 것은 아니며, 본 발명의 실시예의 구성에 대응하는 기능을 실현하는 프로그램 또는 그 프로그램이 기록된 기록 매체를 통해 구현될 수도 있다.

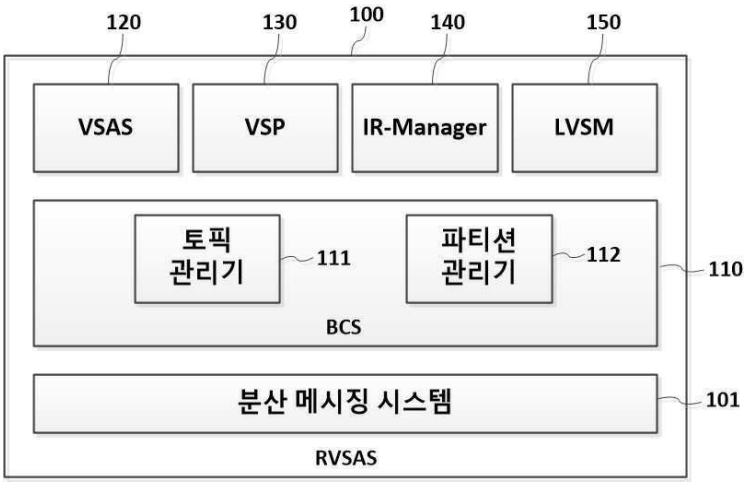
[0137] 이상에서 본 발명의 실시예에 대하여 상세하게 설명하였지만 본 발명의 권리범위는 이에 한정되는 것은 아니고 다음의 청구범위에서 정의하고 있는 본 발명의 기본 개념을 이용한 당업자의 여러 변형 및 개량 형태 또한 본 발명의 권리범위에 속하는 것이다.

도면

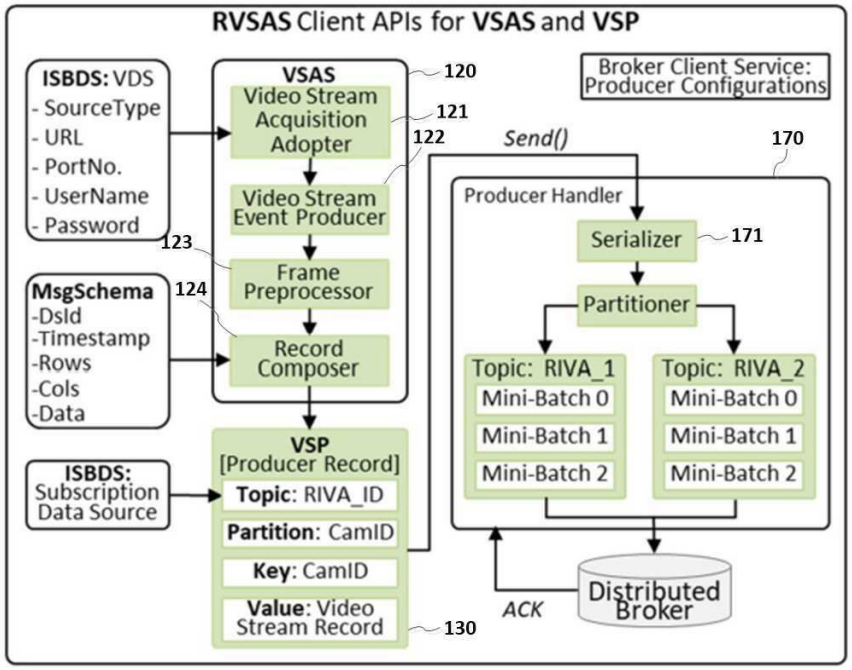
도면1


10
500
400
100
웹 서비스
비즈니스 로직
RVSAS
300
DPBDS
ISBDS
200
지능형 비디오 빅데이터 분석 시스템

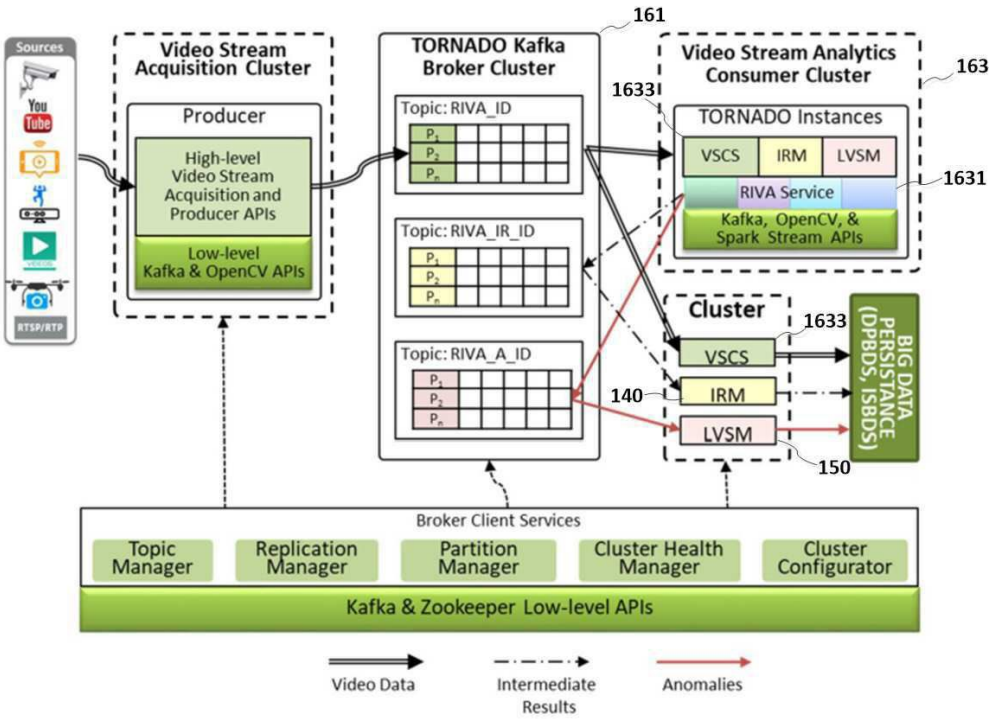
도면2


120
130
100
140
150
VSAS
VSP
IR-Manager
LVSM
토픽 관리기
111
파티션 관리기
112
110
BCS
분산 메시징 시스템
101
RVSAS

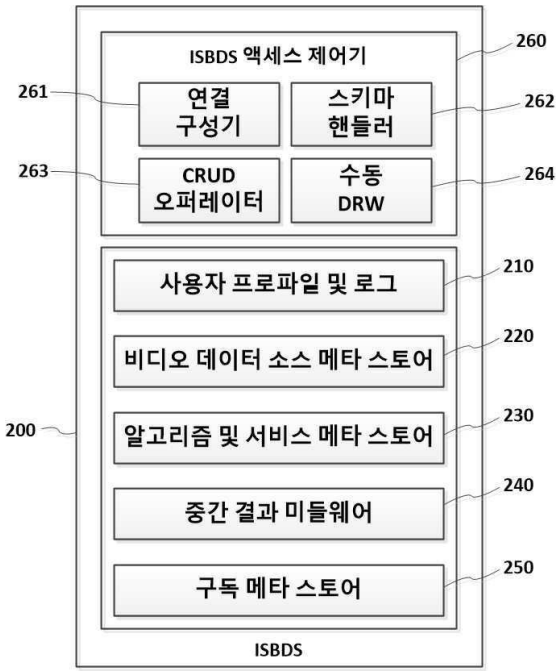
도면3


RVSAS Client APIs for VSAS and VSP
ISBDS: VDS
- SourceType
- URL
- PortNo.
- UserName
- Password
VSAS
Video Stream Acquisition Adopter
120
121
122
Video Stream Event Producer
123
Frame Preprocessor
124
MsgSchema
-DsId
-Timestamp
-Rows
-Cols
-Data
Record Composer
VSP
[Producer Record]
Topic: RIVA_ID
Partition: CamID
Key: CamID
Value: Video Stream Record
130
ISBDS: Subscription Data Source
Broker Client Service: Producer Configurations
Send()
170
Producer Handler
Serializer
171
Partitioner
Topic: RIVA_1
Mini-Batch 0
Mini-Batch 1
Mini-Batch 2
Topic: RIVA_2
Mini-Batch 0
Mini-Batch 1
Mini-Batch 2
ACK
Distributed Broker

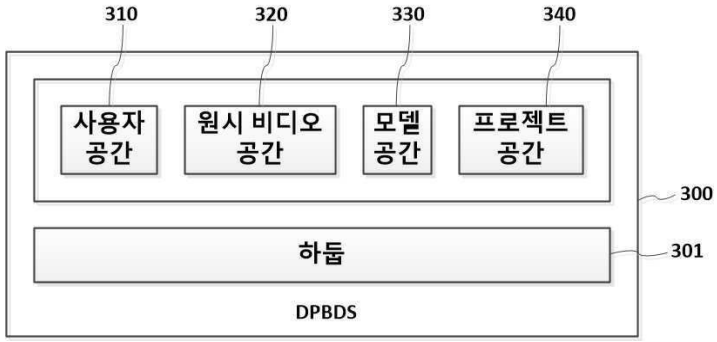
도면4


Sources
Video Stream Acquisition Cluster
Producer
High-level Video Stream Acquisition and Producer APIs
Low-level Kafka & OpenCV APIs
RTSP/RTP
TORNADO Kafka Broker Cluster
161
Topic: RIVA_ID
Topic: RIVA_IR_ID
Topic: RIVA_A_ID
Video Stream Analytics Consumer Cluster
163
1633
TORNADO Instances
VSCS
IRM
LVSM
RIVA Service
1631
Kafka, OpenCV, & Spark Stream APIs
Cluster
1633
VSCS
IRM
LVSM
140
150
BIG DATA PERSISTANCE (DPBDS, ISBDS)
Broker Client Services
Topic Manager
Replication Manager
Partition Manager
Cluster Health Manager
Cluster Configurator
Kafka & Zookeeper Low-level APIs
Video Data
Intermediate Results
Anomalies

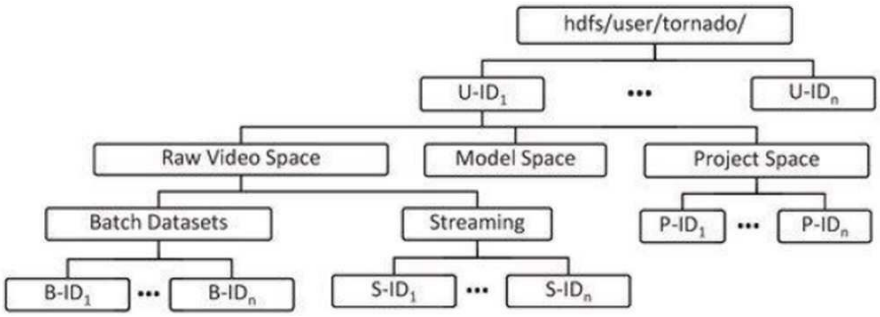
도면5


ISBDS 액세스 제어기
260
261
연결 구성기
스키마 핸들러
262
263
CRUD 오퍼레이터
수동 DRW
264
사용자 프로파일 및 로그
210
비디오 데이터 소스 메타 스토어
220
200
알고리즘 및 서비스 메타 스토어
230
중간 결과 미들웨어
240
구독 메타 스토어
250
ISBDS

도면6


310
320
330
340
사용자 공간
원시 비디오 공간
모델 공간
프로젝트 공간
300
하둡
301
DPBDS

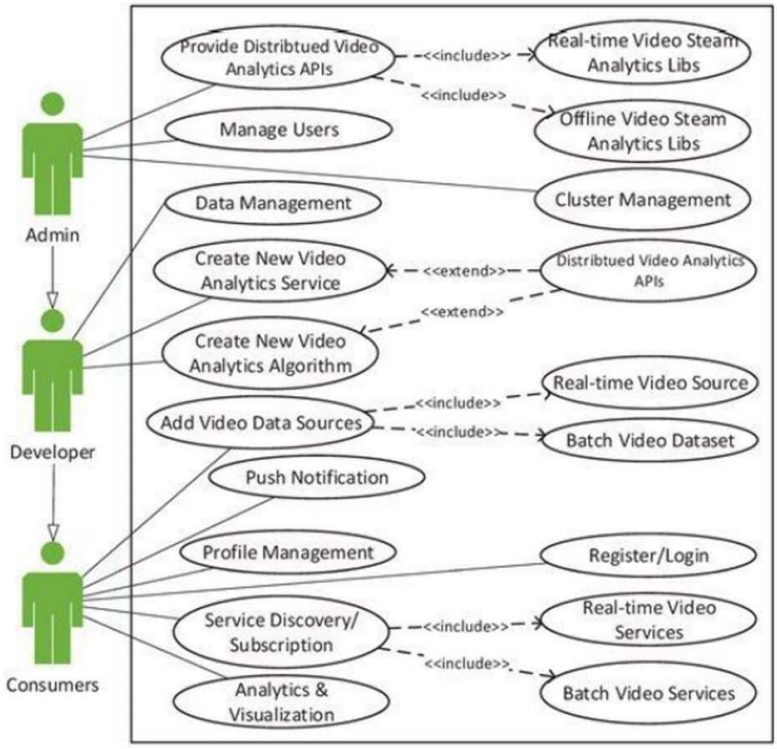
도면7


hdfs/user/tornado/
U-ID1
U-IDn
Raw Video Space
Model Space
Project Space
Batch Datasets
Streaming
P-ID1
P-IDn
B-ID1
B-IDn
S-ID1
S-IDn

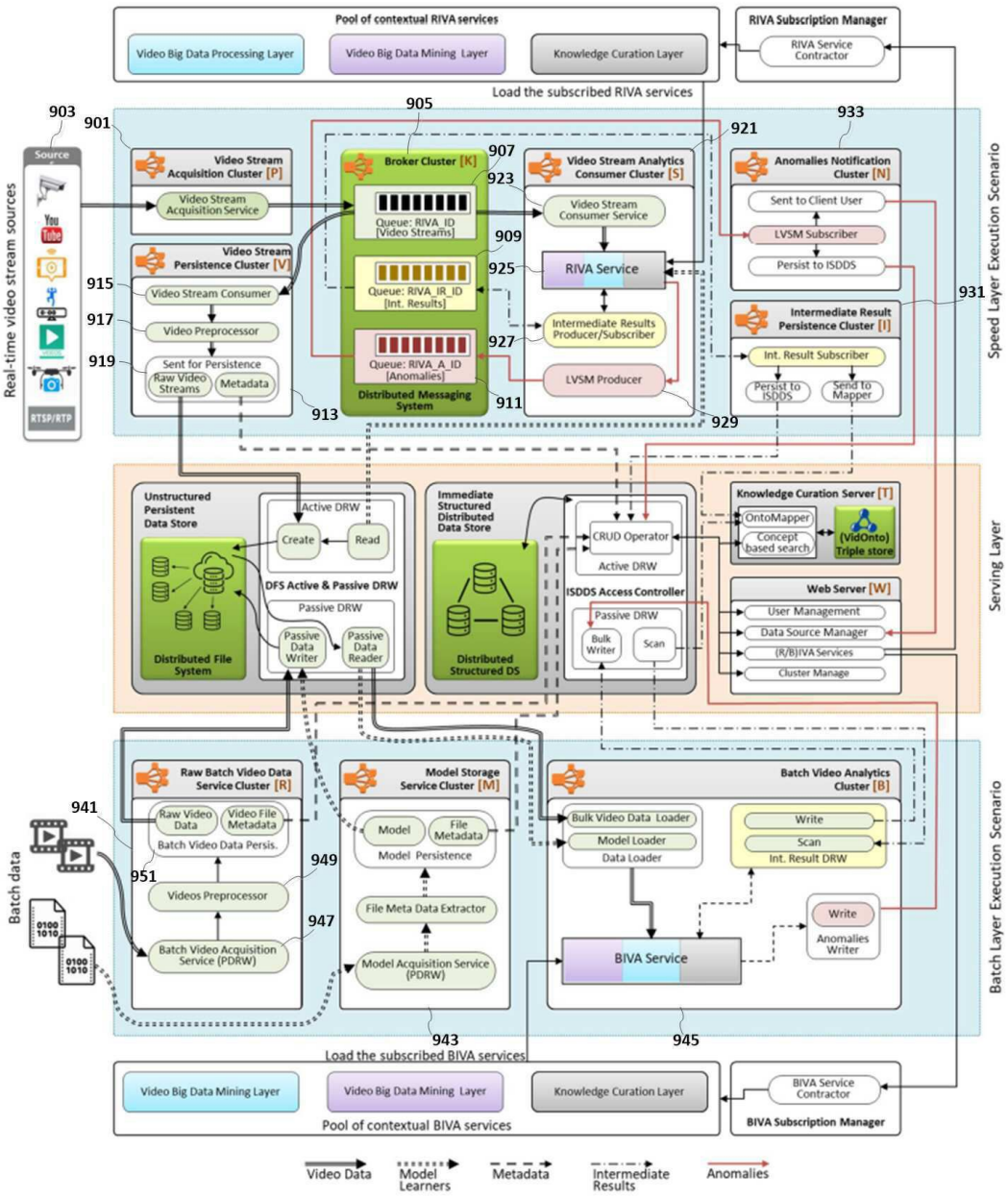
도면8


사용자 관리기
410
데이터 소스 관리기
420
IVA 알고리즘 및 서비스 관리기
430
온톨로지 데이터 관리기
440
클러스터 관리기
450
400
비즈니스 로직

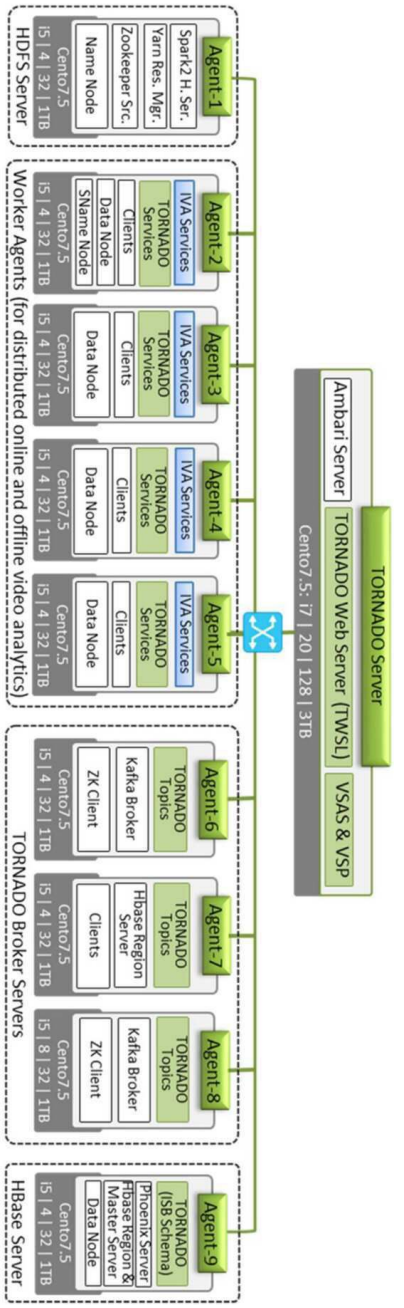
도면9


Admin
Developer
Consumers
Provide Distribtued Video Analytics APIs
Manage Users
Data Management
Create New Video Analytics Service
Create New Video Analytics Algorithm
Add Video Data Sources
Push Notification
Profile Management
Service Discovery/ Subscription
Analytics & Visualization
Real-time Video Steam Analytics Libs
Offline Video Steam Analytics Libs
Cluster Management
Distribtued Video Analytics APIs
Real-time Video Source
Batch Video Dataset
Register/Login
Real-time Video Services
Batch Video Services
<<include>>
<<extend>>

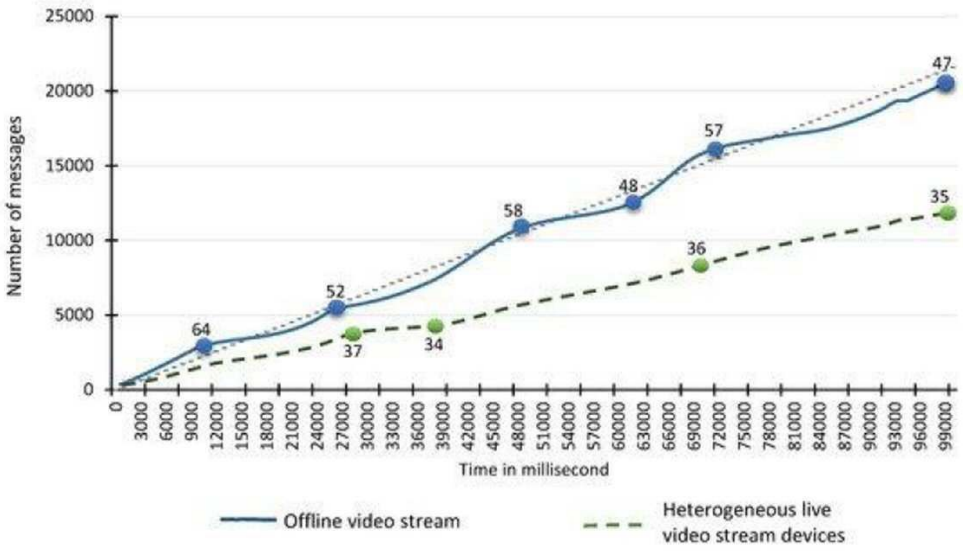
도면10


Pool of contextual RIVA services
Video Big Data Processing Layer
Video Big Data Mining Layer
Knowledge Curation Layer
RIVA Subscription Manager
RIVA Service Contractor
Load the subscribed RIVA services
Real-time video stream sources
Source
RTSP/RTP
Video Stream Acquisition Cluster [P]
Video Stream Acquisition Service
Video Stream Persistence Cluster [V]
Video Stream Consumer
Video Preprocessor
Sent for Persistence
Raw Video Streams
Metadata
Broker Cluster [K]
Queue: RIVA_ID [Video Streams]
Queue: RIVA_IR_ID [Int. Results]
Queue: RIVA_A_ID [Anomalies]
Distributed Messaging System
Video Stream Analytics Consumer Cluster [S]
Video Stream Consumer Service
RIVA Service
Intermediate Results Producer/Subscriber
LVSM Producer
Anomalies Notification Cluster [N]
Sent to Client User
LVSM Subscriber
Persist to ISDDS
Intermediate Result Persistence Cluster [I]
Int. Result Subscriber
Persist to ISDDS
Send to Mapper
Speed Layer Execution Scenario
Unstructured Persistent Data Store
Active DRW
Create
Read
DFS Active & Passive DRW
Passive DRW
Passive Data Writer
Passive Data Reader
Distributed File System
Immediate Structured Distributed Data Store
Distributed Structured DS
CRUD Operator
Active DRW
ISDDS Access Controller
Passive DRW
Bulk Writer
Scan
Knowledge Curation Server [T]
OntoMapper
Concept based search
(VidOnto) Triple store
Web Server [W]
User Management
Data Source Manager
(R/B)IVA Services
Cluster Manage
Serving Layer
Batch data
Raw Batch Video Data Service Cluster [R]
Raw Video Data
Video File Metadata
Batch Video Data Persis.
Videos Preprocessor
Batch Video Acquisition Service (PDRW)
Model Storage Service Cluster [M]
Model
File Metadata
Model Persistence
File Meta Data Extractor
Model Acquisition Service (PDRW)
Batch Video Analytics Cluster [B]
Bulk Video Data Loader
Model Loader
Data Loader
Write
Scan
Int. Result DRW
BIVA Service
Write
Anomalies Writer
Batch Layer Execution Scenario
Load the subscribed BIVA services
Video Big Data Mining Layer
Video Big Data Mining Layer
Knowledge Curation Layer
Pool of contextual BIVA services
BIVA Service Contractor
BIVA Subscription Manager
Video Data
Model Learners
Metadata
Intermediate Results
Anomalies
903
901
905
907
923
909
925
915
917
919
913
911
921
933
931
927
929
941
951
949
947
943
945

도면11


TORNADO Server
Ambari Server
TORNADO Web Server (TWSL)
VSAS & VSP
Cento7.5: i7 | 20 | 128 | 3TB
Agent-1
Spark2 H. Ser.
Yarn Res. Mgr.
Zookeeper Src.
Name Node
Cento7.5 i5 | 4 | 32 | 1TB
HDFS Server
Agent-2
IVA Services
TORNADO Services
Clients
Data Node
SName Node
Agent-3
Agent-4
Agent-5
Data Node
Worker Agents (for distributed online and offline video analytics)
Agent-6
TORNADO Topics
Kafka Broker
ZK Client
Agent-7
HBase Region Server
Clients
Agent-8
TORNADO Broker Servers
Agent-9
TORNADO (ISB Schema)
Phoenix Server
HBase Region & Master Server
HBase Server

도면12


25000
20000
15000
10000
5000
0
Number of messages
64
52
37
34
58
48
57
36
47
35
0
3000
6000
9000
12000
15000
18000
21000
24000
27000
30000
33000
36000
39000
42000
45000
48000
51000
54000
57000
60000
63000
66000
69000
72000
75000
78000
81000
84000
87000
90000
93000
96000
99000
Time in millisecond
Offline video stream
Heterogeneous live video stream devices