타일드 디스플레이 기록기를 위한 병합 알고리즘

최기석° 낭종호
서강대학교 컴퓨터공학과
brix@sogang.ac.kr, jhnang@sogang.ac.kr

Merge Algorithm for Tiled-Display Recorder

Giseok Choe° Jongho Nang
Dept. of Computer Science and Engineering, Sogang University

타일드-디스플레이 시스템은 다수의 디스플레이 디바이스를 그리드 형태로 연결하여 큰 화면과 높은 해상도를 제공해줄 수 있는 시스템이다.[1] 타일드-디스플레이 시스템은 공동 협업 분야에서 다양하게 응요할 수 있는데, 그러한 시스템들은 일반적으로 사용 로그 정보를 기록한다. 이러한 로그 정보는 시스템의 유지 및 관리 보수를 위한 용도 뿐만 아니라, 공동 협업 상에서의 진행 상황을 다시 열람할 수 있는 훌륭한 회의록이 된다. 하지만 타일드-디스플레이 시스템 상에서의 로그 정보를 저장하는 방법에 대해서 관련 연구가 진행되고 있지 않다. 본 논문에서는 타일드-디스플레이 시스템의 특성이 개인용도보다는 공동협업 분야에서 응용된다는 점에 착안하여 타일드-디스플레이 기록기를 제안한다.

타일드-디스플레이에 대한 로그를 저장하는 방법은 크게 두가지로 나누어질 수 있다. 화면을 직접 저장하는 방식과 이벤트 정보만을 기록하여 재현하는 방식이다. 화면을 직접 저장하는 방식은 이벤트 저장방식에 비해 많은 데이터를 제어해야 하는 어려움은 있지만, 응용의 종류에 관계 없이 로그를 저장할 수 있으므로 다양한 응용이 동작하는 타일드-디스플레이 시스템 상에 적합하다. 화면을 직접 캡쳐하는 방식은 화면을 캡쳐하는 주기와 캡쳐된 화면의 해상도에 따라서 영상의 퀄리티가 크게 달라질 수 있는데, 영상을 캡쳐하여 메모리로 가져오는 명령은 실시간 처리가 되지 않는다. 이를 확인 하기 위한 실험에서 화면 캡쳐 명령은 최소한 $12ms$ 이상의 화면 오차가 발생하였으며, 좀더 나은 기록 영상을 얻으려면 화면 병합 알고리즘이 필요하다는 것을 알 수 있다.

N개의 클라이언트로 이루어진 타일드-디스플레이 시스템에서 i번째 클라이언트로부터 받은 이미지는 받은 순서대로 $Buffer_i$에 저장된다고 가정한다. i번째 버퍼에 저장된 이미지 중에서 j번째 이미지를 $Image_i^j$로 표현하며, 이 이미지에 붙어있는 타임 스탬프를 TS_i^j로 표현한다. 출력 이미지를 만들기 위해서는 각 버퍼에 있는 이미지중 하나씩을 고르는 과정을 거쳐야 한다. 이때 $Buffer_i$에서 선택한 이미지 $Image_i^j$의 버퍼 인덱스 j를 s_i로 표현한다. 즉 s_i는 $Buffer_i$에서 선택한 이미지의 버퍼 인덱스이며, 선택한 이미지는 $Image_i^{s_i}$로 표현 할 수 있다. 이러한 선택을 $Buffer_0$부터 $Buffer_{N-1}$까지 하게 되면, $s_0, s_1, ..., s_{N-1}$과 같은 인덱스 리스트를 생성할 수 있다. 이 인덱스 리스트를 S로 표현한다. 조합에 포함된 각각의 이미지들은 서로 다른 타임스탬프(Time Stamp)를 가지고 있다. 병합 서버는 N개 버퍼에서 뽑아온 입력 이미지를 조합하여 출력 이미지를 만들게 된다. N개의 입력 이미지 타임스탬프로부터 출력 이미지의 타임스탬프를 만든다. 이렇게 하여 만들어진 출력 이미지의 $TimeStamp$를 $ATS(ActualTimeStamp)$라고 정의한다. 이것을 기본으로 하여 Delay, Jitter, Skew를 다음과 같은 식으로 정의할 수 있다.

$$Delay(S_k, T_k) = |T_k - ATS(S_k)|$$

$$Jitter(S_k, T_k) = |Delay(S, T_k) - Delay(S_{k-1}, T_{k-1})|$$

$$Skew(S_k) = \sum_{i}^{N-1} |TS_i^{S_i} - ATS(S_k)|$$

좋은 출력 이미지를 만드는 조합은 먼저 Delay가 적을수록 좋은 조합이라 할 수 있고, 출력 이미지의 간격은 일정할 수록 좋으므로 Jitter가 적을수록 좋은 조합이 될 것이다. 또한 출력 이미지를 이루는 입력 이미지들은 서로의 타임스탬프가 가까울 수록 좋은 조합이 된다. 따라서 조합에 대한 평가 함수는 다음과 같이 정의된다.

$$f(S_k, T_k) = \alpha \times Delay(S_k, T_k) + \beta \times Jitter(S_k, T_k) + \gamma \times Skew(S_k)$$

여기서 평가함수의 특성에 의해 다음의 식이 성립함을 알 수 있다.

$$f(S_k, T_k) \leq f(S_k, T_{k+1}), (\because T_k \leq T_{k+1})$$

$$f(S_k^*, T_k) \leq f'(S_k^{END}, T_k)$$

목적 시간 T_{k-1}의 최적의 조합인 S_{k-1}을 알고 있다면, 목적 시간 T_k에 최적인 조합은 S_k은 S_{k-1}보다 이전 시간에 캡쳐된 이미지를 갖지 않는다. 따라서 S_k을 찾기 위해서 처음부터 검색할 필요는 없으며, S_{k-1}의 조합을 이용하여 검색의 시작 지점을 알 수 있다. 또한 모든 S에 대하여 $Skew(S) \geq 0$이기 때문에 검색 종료 지점 S_k^{END} 이후의 S에서는 최적해 S_k^* 가 존재할 가능성이 없다. 이를 통해 최소의 평가 함수 값을 가지는 조합의 범위를 좁힐 수 있으며, 실시간 처리가 가능한 병합 알고리즘을 설계할 수 있다.

설계한 알고리즘의 특성을 분석하기 위해 캡쳐 클라이언트에서 생성한 타임 스탬프를 저장한 뒤, 그 데이터를 이용하여 실험하였다. $5fps$와 $15fps$의 frame rate와 다른 응용 프로그램이 동작하는 시스템 환경을 가정하여 혼잡한 시스템 상태와 혼잡하지 않은 상태를 가정하여 실험에 사용할 데이터를 생성하였다. 제안한 알고리즘은 평가 함수의 세 가지 요소인 Delay, Jitter, Skew의 가중치를 조절하여 이미지 조합 전략에 변화를 줄 수 있는데, 실험에서는 두 가지 평가 가중치를 사용하였다. 첫번째 평가 전략은 Delay만을 고려하는 전략으로 가중치를 $\alpha=1,\beta=0,\gamma=0$으로 설정하였고, 두번째 평가 전략은 세 가지 평가 기준을 전부 고려하는 전략으로 가중치를 $\alpha=1,\beta=1,\gamma=15$으로 설정하였다. 실험 결과 시스템의 혼잡도가 거의 없는 상황에서는 결과에 큰 차이가 없었으나, 시스템의 혼잡도가 커서 캡쳐 명령의 오차가 커지는 상황에서는 각각의 전략 간에 차이가 드러났다. "모두 고려한 전략"은 "Delay만을 고려한 전략"에 비해 Delay와 Jitter가 높은 모습을 보여주었는데, 이는 어긋나지 않은 병합 이미지를 만들기 위하여 출력 이미지 간의 간격이 약간 불규칙해졌기 때문이라고 분석된다. 하지만 Skew가 낮아진 부분을 보았을 때, "Delay만을 고려한 전략"보다 "모두 고려한 전략"은 이미지들 사이가 조금 불안정해졌지만, 각각의 이미지는 좀더 완전하다고 기대할 수 있다. 실험에서 입출력의 frame rate가 같은 경우는 병합 알고리즘으로 인한 품질 향상이 두드러지지 않았으나, 입력 frame rate가 출력 frame rate보다 높은 경우는 품질 향상을 확인 할 수 있었다. 이는 입출력의 frame rate가 같을 경우, 알고리즘이 지능적으로 수행되더라도 실질적인 선택의 폭이 매우 제한되기 때문이라고 생각한다.

일반적인 알고리즘을 사용한 시스템은 캡쳐하는 시스템의 혼잡도가 커져서 캡쳐의 오차가 발생하게 되면, 그것은 곧바로 출력 영상의 품질 저하로 이어진다. 제안한 알고리즘을 사용한 시스템 역시 캡쳐의 오차가 발생하면 출력 영상의 품질이 저하되지만, 병합 알고리즘을 통해 품질이 저하되는 것을 막을 수 있다. 특히 이 병합 알고리즘은 입력 영상의 frame rate 보다 출력 영상의 frame rate가 낮을 경우 더 높은 효율을 보인다. 기록기를 사용하는 여러가지 시나리오 중에서 PDA와 같은 제한적인 기기로 타일드-디스플레이 시스템을 열람하게 된다면, 네트워크 속도나 저장 용량과 같은 기기의 제한으로 인해 출력의 frame rate를 낮춰야 하는 상황이 있을 것이다. 그러한 경우 이 병합 알고리즘은 크게 효율이 있을 것이라 기대한다.

타일드-디스플레이 시스템은 큰 화면과 높은 해상도를 제공해주기 위한 시스템이다. 타일드-디스플레이 시스템이 제공해 줄 수 있는 높은 해상도는 여러 전문 분야에서 활용할 수 있으며, 특히 공동 연구나 협업과 같은 분야에서 주로 활용된다. 그러한 점에 착안하여 회의록이나 원격 참여와 같은 응용으로 활용할 수 있는 타일드-디스플레이 기록 시스템이 필요하며, 하드웨어와 응용에 독립으로 사용하기 위해서는 화면 저장 방식의 로그 저장 방식을 택하여야 한다. 화면 저장 방식은 화면을 캡쳐하기 위한 작업이 필요한데, 이 작업은 다른 응용 작업의 프로세스에 따라 오차가 발생하기 때문에 정확히 동기화된 시간에 캡쳐 화면을 생성할 수 없는 단점을 가지고 있다. 이러한 환경에서 좋은 품질의 로깅 화면을 얻기 위해서는 조합할 수 있는 무수히 많은 조합들을 서로 비교할 수 있는 평가 함수가 필요하다. 따라서 Delay, Jitter, Skew라는 기준에 맞추어 평가 함수를 정의하고 이에 따른 병합 알고리즘을 제안하였다. 제안한 알고리즘은 모든 조합을 비교하는 방법과 동일한 최적 조합을 갖게되며, 실시간 처리가 가능하다.

본 논문에서 제안한 알고리즘은 타임 스탬프에 기반한 방법을 사용하고 있다. 타임 스탬프는 영상을 조합하기 위한 좋은 기준이지만, 원본 화면이 정적인 모습에서 갑자기 동적인 모습을 보이거나 화면 전환을 하는 경우는 예외가 생길 수 있다. 이럴 때에 화면의 비주얼 정보까지 고려한다면 좀더 나은 결과를 얻을 수 있을 것이라 생각된다. 향후에는 타임 스탬프 이외의 정보를 함께 고려한 알고리즘을 연구할 예정이며, 좀더 정확한 화면 캡쳐를 위한 방법도 함께 연구할 예정이다.

참고문헌

[1] Giseok Choe, Jeongsoo Yu, Jeonghoon Choi, Jongho Nang, Design and Implementation of a Real-time Video Player on Tiled-Display System, *Proceedings of the IEEE 7th International conference on Computer and Information Technology*, pp. 621-626, October 2007.

[2] University Illinois, Scalable Adaptive Graphics Environment, *http://www.evl.uic.edu/cavern/sage*, 2007.

[3] R. Singh, B. Jeong, L. Renambot, A. Johnson and J. Leigh, TeraVision: a Distgributed, Scalable, High Resolution Graphics Streaming System, *in proceedings of the 2004 IEEE International Conference on Cluster Computing*, pp. 391-400, 2004.

[4] Y. Zhao and X. Zhang, Design a secure and scalabel CVE: The Access Grid, *in Proceedings of the 2001 ACM/IEEE conference on Supercomputing*, 59-59, 2001.

[5] H. Chen, D. Clark, Z. Liu, G. Wallace and K. Che, Software Environments For Cluster-Based Display Systems, *in proceedings of the 1st International Symposium on Cluster Computing and the Grid*, pp. 202, 2001.

[6] E. Magana, J. Aracil and J. Villadangos, Packet Video Broadcasting with General-Purpose Operating Systems in an Ethernet, *in proceedings of Multimedia Tools and Applications*, pp. 5-28. Sep. 2004.