# 압축된 공간 히스토그램을 이용한 선택율 추정 기법

지 정 희[†]·이 진 열[††]·김 상 호[†]·류 근 호[†††]


## 요 약

공간 질의에 대한 선택율 추정은 가장 효율적인 실행 계획을 찾는데 이용되는 매우 중요한 과정이다. 공간 도메인이 큰 경우, 기존 연구의 요약정보는 상대적으로 적은 정보로 선택율을 추정하기 때문에 좋은 선택율을 유지하기 어렵다. 따라서, 이 논문에서는 작은 저장공간에 공간 요약정보를 압축하는 새로운 기법인 MW 히스토그램을 제안한다. 이 히스토그램은 MinSkew 분할 알고리즘과 웨이블릿 변환이 결합되어 적은 저장공간에서도 타당한 선택율과 압축효과를 얻을 수 있고, 동적 갱신에 대해 효율적으로 대처할 수 있는 구조를 가진다. 실험 결과를 통하여, 버켓 수가 0.3M/6인 MW 히스토그램이 5%~20% 질의에서 평균적으로 좋은 성능을 보이고 있어, MW 히스토그램이 적은 저장공간에서 더 좋은 선택율을 얻을 수 있음을 확인시켜주었다.


# Selectivity Estimation Using Compressed Spatial Histogram

Jeong Hee Chi[†]·Jin Yul Lee[††]·Sang Ho Kim[†]·Keun Ho Ryu[†††]


## ABSTRACT

Selectivity estimation for spatial query is very important process used in finding the most efficient execution plan. Many works have been performed to estimate accurate selectivity. Although they deal with some problems such as *false-count*, *multi-count*, they can not get such effects in little memory space. Therefore, we propose a new technique called *MW Histogram* which is able to compress summary data and get reasonable results and has a flexible structure to react dynamic update. Our method is based on two techniques : (a) MinSkew partitioning algorithm which deal with skewed spatial datasets efficiently (b) Wavelet transformation which compression effect is proven. The experimental results showed that the MW Histogram which the buckets and wavelet coefficients ratio is 0.3 is lower relative error than MinSkew Histogram about 5%~20% queries, demonstrates that MW histogram gets a good selectivity in little memory.


**키워드 : 질의 처리(Query Processing), 선택율 추정(Selectivity Estimation), 압축(Compression), 공간 히스토그램(Spatial Histogram), 웨이블릿(Wavelet)**

## 1. 서 론

공간 데이터베이스 시스템은 사용자 질의의 효율적인 처리를 위해 여러 모듈들을 가지고 있다. 특히, 신속하고 효율적인 질의 처리에 가장 중요한 역할을 하는 질의 최적화기는 지리 정보의 크기가 급격히 증가함에 따라 더욱 중요하게 다루어지고 있다. 또한, 일반적으로 공간 질의는 포인터 셋 기반의 방법인 9-IM(Intersection Method)과 계산기반 방법인 CBM(Calculus-Based Method)을 이용하여 overlap, intersect, contain, disjoint, meet, equal 등 위상 연산자로 표현되고 공간 질의 처리기에 의해 처리된다[24]. 이러한 공간 질의 처리는 일반 데이터에 보다 복잡한 계산과 많은 I/O가 필요하다. 따라서, 질의 처리시 빠른 실행을 위한 최적의 질의 수행 계획(Query Execution Plan : QEP)을 수립하는 것이 중요하다.

이 논문에서는 질의 최적화기에 중요한 구성 요소인 공간 선택율 추정기법에 관하여 연구한다. 공간 선택율 추정은 공간 객체의 분포를 함축한 요약정보를 기반으로 빠른 시간 내에 실제 질의 결과에 대한 근사한 값을 계산하는 과정이다. 이 과정에 이용되는 공간 요약정보는 점, 선, 영역을 갖는 공간 객체의 위치 속성 값에 대한 데이터 분포를 함축적으로 표현한 것이다. 위치 속성은 평면상에 하나 이상의 좌표로 구성되기 때문에 일반 속성과 달리 공간 요약정보를 구성하기 위해서는 제한된 공간을 넘어 더 큰 저장공간을 요구하게 된다. 특히, 공간 도메인이 큰 데이터베이스의 공간 요약정보는 제한된 저장공간에 적합하도록 압축시킬 필요가 있다. 따라서, 이 논문의 목적은 저장공간을 최소화하는 요약정보의 생성 및 관리 방안과 이를 기반한

※ 이 연구는 대학 IT연구센터 육성·지원사업 및 2003년도 건교부 국가 GIS사업(국토연구원)의 연구비 지원으로 수행되었음.
† 준 회 원 : 충북대학교 대학원 전자계산학
†† 준 회 원 : 포인트 아이
††† 종신회원 : 충북대학교 전기전자 및 컴퓨터공학부 교수
논문접수 : 2003년 11월 3일, 심사완료 : 2003년 12월 24일

타당한 선택율을 추정하는 것이다.

기존 상용 데이터베이스 시스템에서의 선택율 추정 기법은 히스토그램 기법, 샘플링 기법과 파라미터 기법 등이 사용되어 왔다[6]. 이 여러 기법 중에 히스토그램은 매우 적은 공간을 사용하고, 사전 데이터 분포를 알고 있지 않아도 되므로 많은 상용 데이터베이스 시스템에서 사용되고 있다. 이 히스토그램 기법은 우리의 연구의 목적에 가장 효율적인 기법이다. 그러나, 일반 선택율 추정과 공간 선택율 추정은 많은 차이가 있다. 먼저 원본 데이터의 형태가 다르다. 공간 객체는 영역을 가지고 있으며 크기가 각 객체마다 다르고 입력 값이 도메인상에 크게 다양하지 않으며, 특정 부분 공간에 편중되어 분포한다. 이러한 공간 객체의 특성을 고려하여 요약정보를 생성할 수 있는 방법은 공간 분할을 이용한 히스토그램 기법, 그래프 이론을 이용하는 히스토그램 기법, 차원 변환 방법을 이용한 기법과 트리를 이용한 기법 등이 있다. 이 중에 공간 분할 히스토그램은 가장 단순한 방법으로 공간을 어떤 조건에 만족하도록 분할하여 생성된 버켓들을 유지하는 방법이다. 또한, 다른 방법들보다 작은 저장공간을 필요로 한다. 특히, MinSkew 히스토그램은 편중된 공간 분포를 고려한 히스토그램이며, 적은 수의 버켓들로 비교적 정확한 선택율을 얻을 수 있다[8].

이 논문의 목적에 부합하고 공간 객체의 특성을 고려한 요약정보를 생성하기 위해 새로운 기법인 MW 히스토그램을 제안한다. 먼저, 편중된 공간 분포를 해결하기 위해 적절한 버켓들을 편중된 공간 영역에 많이 할당하여 버켓내의 객체의 분포가 최대한 일정하도록 분할한 후, 각 버켓 영역내의 공간 분포정보들에 대해 웨이블릿 변환을 수행하여 버켓내의 정보들을 압축하고 유지한다. 웨이블릿 변환은 버켓내에 공간 객체들이 균일하게 분포될 때 매우 높은 압축효과를 생성한다[7]. MW 히스토그램의 구조는 이진 분할 트리 형태이다. 이 구조는 전체적으로 계층적인 구조를 가지게 되어, 갱신에 의한 공간 객체의 분포 변화를 요약정보에 동적으로 반영할 수 있다. 여러 실험결과를 통해서, 이 논문에서 제시한 방법이 적은 저장공간을 사용하고도 낮은 오차를 가진다는 사실을 보여준다.

이 논문의 구성은 다음과 같다. 2장에서는 이 논문에서 제기한 문제점들을 살펴보고 3장에서는 기존 공간 히스토그램과 웨이블릿에 대해 간단히 살펴본다. 4장에서는 공간 히스토그램을 일반화하여 정의하고, 5장에서는 MW 히스토그램의 구조를 기술하고, 생성 절차와 선택율을 추정하는 방법과 동적 갱신에 대한 처리 방법을 설명한다. 6장에서는 여러 실험 결과를 통하여 제안된 기법의 타당성을 검증한다. 마지막으로 7장에서는 결론과 향후 연구를 보인다.

## 2. 문제 정의

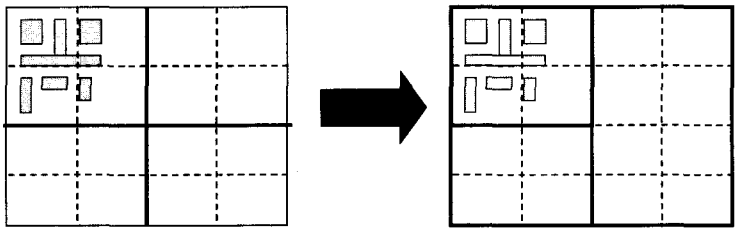
이 절에서는, 이 논문의 연구 동기가 되는 기존 공간 히스토그램의 저장공간에 관한 문제점에 대해 기술한다.

### 2.1 버켓 카운팅(counting)

버켓 카운팅은 임의의 버켓과 위상관계가 참인 객체의 수를 그 버켓의 값으로 할당하는 과정이다. 이 과정은 공간 분포를 저장공간에 얼마나 잘 반영하는가에 대한 평가를 결정짓는 중요한 과정이다. 그러나, 작은 저장공간에 공간 도메인상의 공간 객체의 분포를 완전하게 반영하는 것은 어렵다. 따라서, 비슷한 분포를 갖는 영역을 클러스터링하여 하나의 버켓에 할당하는 작업이 필요하다. 좋은 클러스터링은 선택율 추정시 발생하는 잘못된 카운팅(false counting)과 중복 카운팅(multi-counting)을 줄이게 된다. 잘못된 카운팅은 버켓 영역내에 객체의 실제 분포가 균일하지 않을 때 실제 질의에 만족하지 않는 객체들을 선택율 계산시 포함시키는 문제점을 말한다. 중복 카운팅은 좋지 않은 클러

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 20 | 25 | 1 | 1 | 2 | 15 | 0 | 2 |
| 15 | 10 | 1 | 1 | 0 | 25 | 10 | 2 |
| 5 | 7 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 3 | 5 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 10 | 10 | 0 | 1 | 1 | 2 | 8 |
| 100 | 98 | 19 | 0 | 0 | 0 | 1 | 7 |
| 150 | 100 | 15 | 0 | 0 | 0 | 1 | 10 |

(a) 공간 요약 정보




(b) 축분할

(그림 1) 저장공간의 낭비 요인

스터에 의해 하나의 객체가 서로 인접해 있는 두 버켓에 카운팅되는 문제점을 말한다. 따라서, 좋은 클러스터링은 크기가 작은 요약정보에 공간 분포를 최대한 반영하게 하여 좋은 선택율을 얻을 수 있다.

### 2.2 저장공간의 낭비

공간 분포의 빈도 값은 다른 분포와는 다르게 그 차이가 작다. 즉, 인접한 버켓들 사이의 빈도 차가 크지 않다. Equi-width 공간 히스토그램은 같은 빈도를 갖는 버켓들을 중복 저장하여 저장공간을 효율적으로 사용하지 못하고 있다((그림 1)(a)). 같은 값을 갖는 인접 버켓들은 하나의 버켓으로 클러스터링될 수 있다. 최상의 조건일 경우, 즉, 대부분의 버켓들의 값이 똑같을 경우, 소수의 버켓들만 저장하여도 좋은 선택율을 추정할 수 있다.

일반적인 공간 히스토그램의 축 분할을 통해 비슷한 분포를 갖는 영역을 클러스터링한다. 그러나, 축 분할은 이 과정에서 (그림 1)(b)와 같이 4개의 버켓을 생성한다. 그림과 같이 한 버켓을 제외한 나머지 버켓들은 같은 값을 갖는다. 따라서, (그림 1)(b)의 오른쪽과 같이 전체 버켓과 객체들를 포함하는 버켓만 유지하여도 정보의 손실 없이 저장될 버켓의 수를 줄일 수 있다. 정보 손실 없이 버켓들을 줄이는 것은 작은 저장공간에서 좋은 선택율을 유지할 수 있는 최선의 방법이다.

## 3. 관련 연구

이 절에서는, 지금까지 제안된 공간 히스토그램을 검토한 후, 이 논문에서 요약정보를 압축하기 위해 사용한 Haar 웨이블릿 변환(Wavelet transform)에 대해 살펴본다.

### 3.1 공간 히스토그램

공간 선택 질의에 관한 선택율 추정 연구 중 영역 객체를 대상으로 하는 히스토그램 기반 접근 방법에는 [3-5, 8, 10, 16, 19, 25]등이 있다. Poosala[8]등은 영역 객체를 점 객체로 효율적으로 변환한 데이터 셋의 밀도 히스토그램을 기반으로 주어진 버켓에 도달할 때까지 분할하여 공간 편중을 줄이는 Min-Skew알고리즘을 제안하고 있다. 이 기법의 목적은 버켓 내의 공간 분포를 최대한 균일하게 하여 잘못된 카운팅으로 인한 선택율 오차를 줄이는 것이다. 그러나, 축 분할을 사용하기 때문에 불필요한 버켓들을 생성한다. Aboulnaga[10]등은 폴리곤 데이터 셋에 대한 선택율 추정을 위해 SQ 히스토그램을 제안하고 있다. SQ 기법은 Quad-Tree를 기반으로 버켓의 계층도를 생성하여 객체를 그들의 중심점과 면적에 따라 버켓을 할당하는 방식이다. Ji Jin[25]등은 CD 히스토그램을 제안하고 있다. 이 히스토그램은 영역 객체가 여러 버켓에 걸쳐있을 경우, 여러 번 카운트하는 문제를 해결하기 위해 제안된 기법이다. 이 기법은 영역 객체의 최소 경계 사각형(Minimum Boundary Rectangle : MBR)의 네 모서리 점을 하부 누적 히스토그램에 각각 저장한다. 따라서, 하나 객체에 대해 네 번 따로 저장해야 하는 부담을 안고 있다. Sun C.[16, 19]등은 오일러 히스토그램을 일반화하여 기하 선택을 갖는 조인의 선택율 추정에 사용하였다. 이 기법은 각 그리드 셀 뿐만 아니라 선와 점에도 버켓을 할당함으로써 여러 버켓과 겹치는 영역 객체의 중복 카운팅하는 문제를 해결하고 있다. 이 기법은 공간 기하 질의에 대한 선택율을 최초로 다루고 있다. 그러나, 이 기법들은 클러스터과정이 없기 때문에 같은 값을 갖는 버켓들을 생성한다. 이 밖에 좀 더 정확한 선택율 추정을 위해 변형된 기법으로 차원 변환 기법이 존재한다[5, 14]. 특히, Wang[14]은 웨이블릿 히스토그램과 차원 변환 기법을 결합하여 낮은 저장공간을 유지하면서 선택율 오차가 적은 히스토그램을 제안하였다.

이들 기법의 주요 관심은 공간 객체의 특성에 의해 발생되는 문제점을 최소화하여 높은 정확도의 선택율을 얻는데 있다. 그러나, 공간 도메인의 크기가 증가하게 되거나 저장공간이 작은 모바일 환경일 경우, 선택율의 정확도를 유지하기 위해서는 더 많은 저장공간을 요구하게 된다. 또한, 하나 버켓의 점유하고 있는 공간 범위가 상대적으로 매우 크므로, 이 기법들이 주장하는 효과를 얻기가 힘들다.

이런 점을 개선하기 위해 웨이블릿 변환 및 DCT(Discrete Cosine Tranform)등의 압축 기법과 결합하는 연구가 많이 수행되었다[7, 14, 21]. 특히, Haar 웨이블릿은 빠른 계산 속도와 효율적인 저장공간을 갖는다[7, 21]. 또한, 한계 값을 설정하여 신뢰할 수 있는 선택율을 얻을 수 있으며, 동적인 삽입과 갱신에 매우 유연한 구조를 갖는다[11, 22].

### 3.2 Haar 웨이블릿 변환

웨이블릿 변환 기법은 주로 신호처리와 영상처리에서 사용되어 왔다. 최근 들어, 이 기법을 데이터베이스의 근사 질의 처리와 선택율 추정기법에 이용하려는 연구가 활발히 진행되고 있다. 또한, 데이터 웨어 하우싱에서 이 기법을 이용하여 요약정보를 구성하여 근사 질의(approximate query)를 처리하는 연구가 진행되고 있다.

웨이블릿 변환은 주어진 함수를 근사하는 함수인 배율함수(scaling function)과 배율함수로 인해 생기는 오차를 보상하는 함수인 웨이블릿 기저 함수(wavelet function)의 합에 의해 변환된다. 이 논문에서는 변환 연산과 압축이 간단하여 많은 연구에 사용되고 있는 Haar 웨이블릿을 사용한다. Haar 웨이블릿은 배율 함수 계수(scaling coefficient)와 웨이블릿 계수(wavelet coefficient)를 구하는데 평균 함수(average function) 와 편차 함수(difference function)을 사용한다. 이 중에 편차함수에 의해 생성된 값을 웨이블릿 계수라 하며, 이 계수들로 웨이블릿 요약정보를 구성한다. 원

본 데이터는 분해과정을 거꾸로 진행시키는 복원과정을 통해 구하게 된다.

이 변환은 웨이블릿 계수가 0인 값을 제외하고 나머지 값을 저장함으로써 압축효과를 얻어올 수가 있다. 또한, 오차가 최소인 계수 값을 0으로 대치함으로써 최소 오차를 갖는 웨이블릿 요약정보를 얻을 수 있다. 이 것은 원본 데이터와 추정 데이터 사이의 오차가 한계 값을 넘지 않음을 보장한다. 이 웨이블릿 압축은 변환과정에서 이루어 지며, 웨이블릿 분해와 복원은 N개의 배열에 대해 단지 O(N) CPU 시간만이 요구된다.

웨이블릿을 이용한 압축은 전체 데이터의 오차를 최소화하는 계수들을 선택하여 0으로 대치함으로써 저장할 때 제외시킨다. 이런 오차 계산법으로는 일반적으로 평균 제곱오차 $L^2$을 사용한다. 이 오차를 최소화하기 위해서는 각 웨이블릿 계수를 $\sqrt{2^{level}}$로 나누어 계산된 정규화 계수를 큰 순서로 정렬한 후, 요구하는 B개의 계수들을 선택하여 저장한다. 이런 방법에 의해 웨이블릿을 압축한다. 추정 오차를 최소화하게 압축시키는 방법으로는 전체 오차를 최소화하는 결정적 임계(deterministic threshold)과 상대 오차를 최소화하는 확률적 임계(probabilistic threshold)를 이용하는 방법이 있다. 결정적 임계 값은 $L^2$오차를 최소화하는 것으로 전체 오차를 최소화하는데 사용된다. 결정적 임계로 압축된 웨이블릿 요약정보는 범위 질의에 높은 정확도를 갖는다. 그러나, 점 질의(개개 질의)에 대해서는 큰 오차를 갖는다. 이런 점을 해결하기 위해 상대 오차를 최소화하는 확률적 임계가 제안되었다[22]. 이것은 임의의 계수가 제거될 때 각 데이터에 미치는 영향이 최소가 되는 확률에 따라 계수의 값을 조정하여 상대 오차를 최소화한다. 결정적 임계를 사용하는 방법은 계산비용이 저렴하다, 반면 확률적인 임계를 사용하는 방법은 유지될 계수를 얻기 위해서는 비싼 계산비용을 요구한다.

## 4. 공간 선택율 추정의 일반화

많은 공간 히스토그램은 다양하게 표현되고 정의되고 있다. 그러나, 이런 히스토그램은 공통적인 절차에 의해 수행된다. 따라서, 이 절에서는 히스토그램의 생성과정과 선택율 추정과정을 일반화하는 동시에 공통적인 분모에 대해 기술한다.

### 4.1 표 기

논문 전개상 필요한 표기들을 <표 1>에 기술한다. <표 1>은 공간 히스토그램 생성 및 제안된 히스토그램을 위한 표기이다.

### 4.2 일반적인 공간 선택율 추정과정

다음은 일반적인 히스토그램 생성 과정을 기술하고, 이 과정에서 필요한 요소를 정의한 것이다.

<표 1> 히스토그램 표기

| 표 기 | 의 미 |
|---|---|
| $D_X$ | 속성 X가 가질 수 있는 값 집합 |
| $V_X$ | 속성 X의 값 집합, $V_X \subset D_X$ |
| $r$ | 해상도(resolution), $r = \log_2 \|C\|$ |
| $B_i$ | 전체 공간을 B개의 버켓으로 나누었을 때, i번째 나누어진 버켓 |
| $c_{ij}$, $\|C\|$ | 전체 공간을 일정한 크기로 분할하여 생성된 i, j번째 격자, 전체 격자의 크기 |
| $f_{ij}$, $f_k$ | 격자 $c_{ij}$ 및 버켓 $B_k$와 위상관계가 참인 객체의 빈도 |
| $cf_{ij}$ | 원점 (0,0) 부터 (i, j)의 영역과의 위상관계가 참인 객체의 수 |
| $A$, $A_i$ | 전체 공간을 격자 분할하여 생성된 배열과 i번째 버켓의 원본 데이터 배열. |
| $N_z$, $N_{iz}$ | 전체 공간 및 버켓 $B_i$의 밀도 배열에서 0이 아닌 배열 요소의 수 |
| $s_{ij}$ | 격자 $c_{ij}$의 공간 편중도 |
| $ss_{ij}$ | 격자 $c_{ij}$의 객체 크기 편중도 |
| $BT$ | 전체 공간을 분할하는 인덱스를 노드로 형성된 이진 분할 트리 |
| $snode_i$ | 이진 분할 트리의 내부 노드로 분할 노드라 부른다. |
| $TF$ | 격자 $c_{ij}$ 또는 버켓 $B_i$ 값을 결정하는 요약함수 |
| $MF$ | 공간 분할에 기준이 되는 값을 측정하는 함수, 측정함수 |
| $S(Q)$ | 임의의 질의 Q에 대한 공간 선택율 |

1. 공간 데이터분포 **ST**의 형태를 결정하고, 연속적인 각 축의 값을 비연속적인 값으로 변환시키기 위해 전체 데이터 분포를 **M×N**의 해상도가 되도록 격자 분할을 수행한다.
2. 각 격자의 값은 정의된 분할 함수 **MF**의 기초함수 **h**에 의해 결정된다. 결정된 데이터분포의 형식에 맞는 전체 데이터분포의 배열 A을 생성한다. A는 배열요소 A[i, j]에 의해 구성된다($0 \le i \le M-1$, $0 \le j \le N-1$. i, j는 정수).
3. 공간 버켓 Bi을 생성하기 위해 결정된 MF함수에 의해 분할 조건에 따라 전체 데이터분포를 $\beta$개 만큼 분할한다($1 \le i \le \beta$).
4. 만일, 요약함수 **TF**와 **MF**의 **h**함수가 같지 않다면, 생성된 버켓 $\mathbf{B_i}$의 값을 TF에 의해 다시 계산한다.

(알고리즘) 일반적인 공간 요약정보 생성

공간 히스토그램의 데이터 분포 ST는 단순 데이터분포, 누적 데이터분포가 있다. 누적 데이터분포는 누적에 대한 추가 비용이 필요하지만, 선택율 추정 시 반복적인 계산비용이 줄어든다.

**[정의 1]** 요약 함수 **TF(Topology Function)**. 요약함수는 공간 위상함수로 정의되며, 여러 위상함수의 교집합 혹은 합집합으로 정의될 수 있다. 예를 들어, 요약함수는 다음과 같이 정의할 수 있다.

$$\mathbf{TF(c_{i,j}) = Count(Contain(c_{i,j}, ST) \vee Overlap(c_{i,j}, ST))}$$

**[정의 2]** 분할 함수 **MF(Measure Function)**. 공간 분할은 기초 함수, 무게 함수, 누적 함수에 의해 분할 된다[26]. 따라

서, 분할 함수 MF는 f-g-h 조합이다.

- 기초 함수(Elementary Function) $h$ : 어떤 버켓 $B_i$내 배열요소 $A_i[n, m]$를 실수로 사상시키는 함수이다. 즉, $A_i[n, m]$의 요약범위에 존재하는 공간 객체들을 실수로 사상시키는 함수이다. 일반적인 함수로 다음과 같은 것이 있다.
  - $A_i[n, m] = TF(c_{n,m})$, AVG_DIFF($A_i[n, m]$), SQR_DIFF = {AVG_DIFF($A_i[n, m]$) }$^2$
- 무게 함수(Heft Function) $g$ : 버켓 $B_i$의 대표 값을 구하는 함수이다. 주로 버켓내 배열요소들의 합(SUM)이나 최대값(MAX)로 버켓 값을 대표한다.
- 누적 함수(Cumulative Function) $f$ : 전체 버켓에 대한 합이거나 결합된 무게함수 g로 정의한다. 이것은 버켓을 분할할 때 기준이 되는 함수이다.

Equi-Count 히스토그램은 각 버켓내의 배열요소의 합이 최대값 max을 넘지 않게 전체 공간을 분할한다. 이와 같은 경우 분할함수 MF는 MAX-SUM-ID와 같다.

공간 선택율 추정 과정은 생성된 공간 히스토그램의 데이터분포 형태, 요약함수와 측정함수에 의존한다. 단순 데이터분포일 경우 윈도우 질의와 교차하는 버켓들에 대해 복원함수를 적용하여 계산한다. 그러나, 누적 데이터분포일 경우 윈도우 질의의 각 네 점에 대해 복원함수를 적용하여 계산한다. 또한, 요약함수의 정의에 따라 배열요소 A[i, j]의 값이 갖는 의미가 다르게 되므로 선택율은 공간 연산에 따라 다르게 계산된다.

다음은 일반적인 선택율 추정과정이다.

1. 요약정보의 데이터 분포 형태 ST와 요약함수 TF의 정의에 따라 복원 함수 RF와 선택율 추정 전략을 선택한다.
2. 선택된 추정 전략에 따라 임의의 질의를 해석하여 계산되어야 할 공간 버켓들을 찾는다.
3. 선택된 버켓들을 복원 함수를 통해 원본 데이터로 복원하고, 선택된 전략에 의해 선택율을 계산한다.

(알고리즘) 일반적인 공간 선택율 추정

**[정의 3]** 복원 함수 **RF(Recovery Function).** 공간 요약정보를 기반으로 원본 데이터를 복원하는 함수이다. 예를 들어, $B_i$와 질의 Q가 교차할 경우 교차하는 영역에 대한 복원은 일반적으로 다음과 같이 계산된다.

$$RF(B_i \cap Q) = \frac{Area(B_i \cap Q)}{Area(B_i)} \times TF(B_i)$$

## 5. MW(MinSkew-Wavelet) 히스토그램

이 논문의 주요 목적은 공간 객체의 특성을 고려하면서 최대 압축효과를 갖는 공간 히스토그램을 이용하여 타당한 선택율을 얻는데 있다. 또한, 동적인 삽입과 갱신에 매우 유연하게 대처할 수 있는 구조를 얻는 것이다. 따라서, 이 절에서는 위 목적에 부합하는 MW 히스토그램을 제안한다.

### 5.1 MW 히스토그램의 구조

기존 연구로부터 우리는 MinSkew 히스토그램의 BSP(Binary Split Partitioning) 기법과 웨이블릿 변환의 결합이 압축 효과를 최대로 높이고 편중된 공간 분포를 해결할 수 있음을 발견하였다. 또한, 공간 분할에 의한 버켓 생성이 동적 삽입과 갱신에 매우 비효율적인 대처 능력을 가진다는 단점을 발견하였다. 따라서, 제안된 MW 히스토그램의 구조는 전체적으로 동적인 계층구조를 가지면서, 두 기법의 장점을 최대한 반영되도록 설계한다.

#### 5.1.1 기본 아이디어

이 논문에서는 MinSkew 히스토그램과 웨이블릿 변환에 대해 다음 네 가지 특징들에 초점을 맞춘다.

① MinSkew 히스토그램의 목적은 버켓 내의 공간 분포를 최대한 균일하게 하는 것이다.
② BSP 분할 기법은 편중된 공간 분포를 해결하기 위해 두 번의 축 분할한다. 이것은 총 4개의 버켓을 생성시키지만, 하나를 제외한 나머지 3개의 버켓은 같은 빈도 값을 가지므로 하나의 버켓으로 대치될 수 있다. 따라서, 실제 2개의 버켓만 유효하다.
③ 웨이블릿 변환은 공간 분포가 균일할 때, 최대한의 압축효과를 얻을 수 있다.
④ 웨이블릿은 동적 삽입과 갱신에 매우 효율적인 대처 능력을 가진다.

위 특징들은 이 논문의 목적에 매우 잘 부합된다. 특히 ①~③번의 특징은 MinSkew 히스토그램과 웨이블릿 변환을 결합 시, 최대 압축 효과를 가져올 주요특징들이다. 따라서, 이 논문의 기본 아이디어는 MinSkew 히스토그램에서 이용된 분할 알고리즘을 적용하여 버켓들을 생성 후, 각 버켓에 웨이블릿을 적용하면 높은 압축 효과를 가져올 수 있으며, 동시에 편중된 공간 분포도 해결할 수 있다는 것이다. 구체적으로 기술하면, 저 해상도 레벨에서 MinSkew 히스토그램에서 이용된 분할 알고리즘을 적용하여 버켓들을 생성한다. 생성된 버켓들의 공간 분포는 편중되지 않을 것이며, 최상의 경우에 균일하게 될 것이다. 이 분할 과정에서 분할되는 버켓 정보와 축 분할이 일어나는 위치와 축 등을 저장하는 분할 노드들을 생성한다. 이 분할은 계층적인 형태를 가지므로 분할 노드들은 이진 트리 형태로 구성된다. 분할이 완전히 끝난 후, 각 버켓 내의 격자들에 1차 Haar 웨이블릿 변환을 적용하여 웨이블릿 요약정보를 생성한다. 각 버켓은 생성된 웨이블릿 요약정보로 대치된다. 웨이블릿 요약정보는 오차 트리라는 계층적인 구조로 변환할 수 있다. 따라서, 전체적으로 MW 히스토그램은 계층적인

구조를 갖는다. 또한, 웨이블릿 요약정보의 부모는 분할 노드로 공간 분포에 대한 정보를 가진다. 이 공간 분포 정보는 동적 갱신에 의한 변화에 대처할 수 있는 기준이 된다.

5.1.2 생성 절차

MW 히스토그램의 생성은 다음과 같은 세 단계로 이루어진다.

① 공간 분할 단계(Partitioning) : 전체 공간 $|D_X \times D_Y|$을 기본 해상도 $r$의 1/2 배인 저 해상도에서 공간 분할을 시작한다. 만일 요구하는 버켓의 수가 $b$일 때, Min-Skew 분할 알고리즘에 의해 b-1개의 분할 노드 $snode_i$ $(0 \le i < b-1)$가 생성되어 이진 분할 트리를 형성한다. (그림 2)(b)는 해상도 $r=8$이고, $b=5$인 데이터 분포 (그림 2)(a)에 대한 공간 분할을 나타내고 있다. 이 과정에서 (그림 2)(c)와 같은 이진 분할 트리가 생성된다. 각 분할 노드는 다음과 같은 구조를 갖는다.

**snode = {axis, split position, spatial skew, right split node, left split node}**

② 웨이블릿 변환 단계(Wavelet Transform) : 이진 분할 트리의 단말 노드에 하나 혹은 두 개의 버켓을 생성한다. 각 생성된 버켓 $B_i$가 점유하고 있는 영역에 1차원 Haar 웨이블릿 변환를 적용하여 웨이블릿 요약정보 $W_{Ai}$를 생성한다( $0 \le i < b$).

③ 계수 축소 단계(Shrinking) : 각 웨이블릿 요약정보 $WA_i$에 유지되는 계수의 수를 제한된 저장공간을 만족할 때까지 줄인다.

**B = {spatial skew, Wavelet Synopsis ∈ {coefficient index , coefficient }}**

만일, 요구하는 버켓의 수가 b이고, 각 버켓의 평균 유지되는 웨이블릿의 계수의 수를 Ws, 각 구조의 인자를이 모두 같은 크기라면, 총 저장공간의 크기 M는 다음과 같다.

$$\mathbf{M = 5(b-1) + b(1+2Ws)} \quad (1)$$

5.2 히스토그램 생성

이 절에서는 이진 분할 트리의 생성과정과 알고리즘을 기술하고 Haar 웨이블릿 요약정보 생성과정을 논의한다.

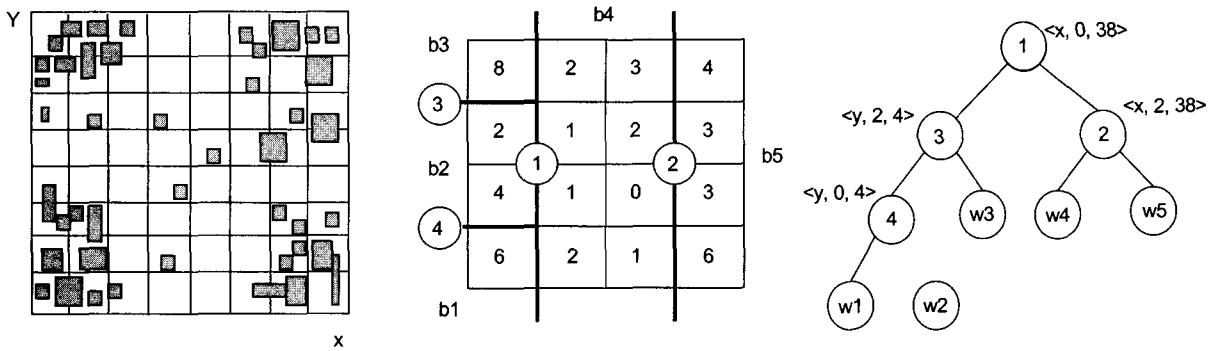
5.2.1 이진 분할 트리

공간 분할은 공간 밀집도(spatial density)와 공간 편중도(spatial skew)를 이용하여 분할을 수행한다. 공간 밀집도는 분할 함수 MF 중 기초 함수 h에 해당하며 어느 격자 $c_{i,j}$에 교차 관계(Intersect)를 갖는 객체들의 수로 표현된다. 공간 편중도는 분할 함수 중 무게함수 g에 해당하며, 어느 일정한 영역에 포함되는 격자들의 통계적 분산에 의해 계산된다. 따라서, 분할 함수 MF는 다음 함수들의 조합으로 정의한다(MF = $f-g-h$ ).

$$\begin{aligned} h(c_{i,j}) &= f_{i,j} = count(intersect(c_{ij}, A)), g(B_k) \\ &= s_k = \frac{(f_{ij}-\bar{f})^2}{|B_k|}, f(A) = MAX(s_k) \end{aligned} \quad (2)$$

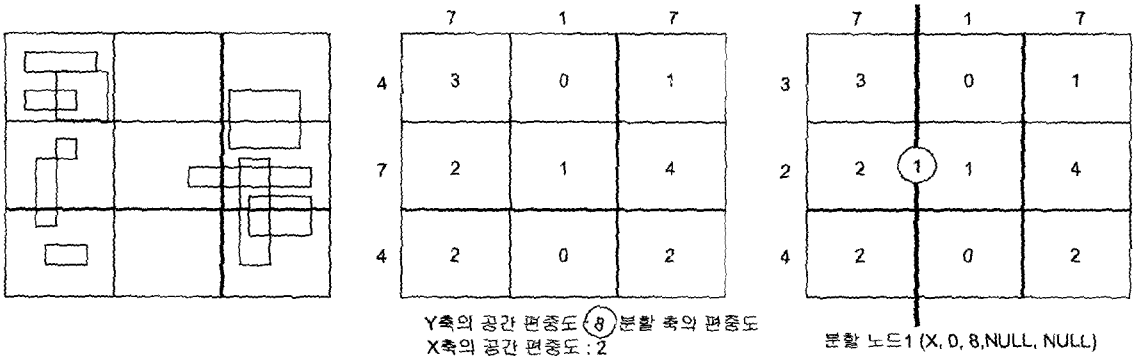
(단, $i$, $j$ : $k$번째 버켓 $B_k$의 포함되는 격자의 인덱스,
$A$: 전체 데이터분포, $0 \le k < b$)

이 분할 함수 MF는 분할된 각 버켓들의 공간 편중도를 최소가 되도록 분할한다. (그림 3)은 공간 분할 과정을 보여주고 있다. 먼저, h 함수에 의해 격자 안의 값을 구한다. 그럼 다음, 각 축에 사상된 값을 구한 다음, 각 축에 대해 g 함수를 적용하여 축의 편중도를 구한다. 마지막으로 f 함수에 의해 최대 축 편중도를 갖는 버켓과 축을 선택하여 분할될 버켓의 편중도가 최소가 되도록 분할한다. 분할 정보를 갖는 분할 노드 1을 생성하여 이진 분할 트리에 삽입한다. 이와 같은 방법으로 요구하는 버켓의 수가 생성될 때까지 이 과정을 반복한다. 이 과정은 (알고리즘 1)에 자세히 기술된다. 분할을 통해 생성된 버켓들은 이진 분할 트리에 의해 접근이 가능하다. 따라




(a) 데이터 분포 (b) 분할 단계 (c) 이진 분할 트리

(그림 2) MW 히스토그램의 구조




(그림 3) 공간 분할 과정

서, 생성과정이 모두 끝나면 버켓의 영역정보는 삭제되어 이진 분할 트리만 존재하게 된다.

***Input*** : 데이터 집합 ST, 해상도 $r$, 요구하는 버켓의 수 $b$
***Output*** : 이진 분할 트리 **BT**

Snode$i$ : $i$번째의 분할 노드, $B_i$ : $i$번째 버켓,
$B : \{B_0, B_1, B_2, \cdots, B_{b-1}\}$
전체 도메인을 해상도 $r$의 1/2인 저해상도로 분할한 다음, MF의 기초함수 $h$를 통해 각 격자 $c_{i,j}$의 값을 할당하고, 전체 도메인을 초기 버켓 $B_0$으로 설정한다.
**While** $i < b$
  **For Each** $B_i$ in $B$ **Do**
    각 버켓 $B_i$에 대해 각 축의 각 인덱스에 대한 누적 밀집도를 계산한다.
    각 축에 대한 편중도를 구한 다음, 편중도가 최대인 축과 편중도를 저장한다.
  **End For**
  현재 존재하는 버켓들 중에 가장 편중도가 큰 버켓 $B_k$을 선택하고, 그 버켓을 가르키는 상위 분할노드 snode$_k$와 방향 direction을 얻는다.
  선택된 버켓을 편중도가 최소가 되도록 선택된 축에 따라 분할하고, 이 분할 정보를 통해 새로운 분할노드 snode$_{i-1}$을 생성한다.
  Snode$_k$.direction_pointer = snode$_{i-1}$
***End*** **While**

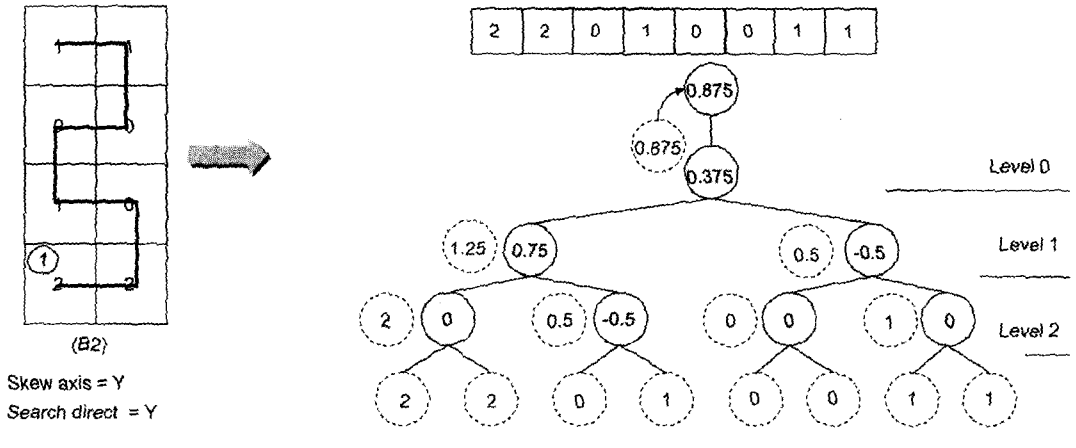
(알고리즘 1) 공간 분할

5.2.2 웨이블릿 변환

모든 분할이 끝나면, 우선 기본 해상도 $r$를 기준으로 격자를 생성하고 요약 함수 TF를 적용하여 각 격자의 값을 설정한다. 요약 함수는 공간 객체의 중심점이 격자 $c_{i,j}$에 포함된 객체들의 수를 카운트한 값을 리턴한다. 이 요약함수를 다음과 같이 정의한다.

$$TF(c_{i,j}) = count(contain(c_{i,j}, centroid(ST))) \quad (3)$$

다음으로, 생성된 각 버켓이 점유하는 영역내의 격자 값에 대한 웨이블릿 변환을 수행한다. 버켓의 각 차원의 크기가 동일하지 않기 때문에 2차원 웨이블릿 변환을 적용할 경우, 각 차원크기가 커지는 문제점을 갖는다. 이 문제점은 1차원 웨이블릿 변환에서도 발생하지만 2차원 웨이블릿 보다는 작은 확장을 가진다. 따라서, 이 논문에서는 1차원 웨이블릿을 적용한다. 먼저, 버켓의 2차원 격자를 1차원으로 변환하는 과정이 필요하다. 이 과정은 공간 탐색 정렬 기법(space-filling order) 기법을 통해 이루어진다. 웨이블릿은 인접한 영역의 값이 비슷할 경우에 값이 0에 근접한 계수들이 많이 생성된다. 그 만큼 압축효과를 더 높일 수 있다. 따라서, 이 논문에서는 분할 축과




(a) 버켓과 부모분할노드의 정보 (b) 웨이블릿의 오차 트리

(그림 4) MW 히스토그램의 웨이블릿 변환

공간의 인접성, 연결성을 고려하여 H-mirror 기법을 사용한다. (그림 4)과 같이 진행방향(search direction)은 부모 분할 노드의 분할 축과 같은 방향이다. 분할 축 방향으로 비슷한 값이 많이 분포될 확률이 크기 때문에 자동으로 0이 되는 계수가 많아 압축이 효과적이다. (그림 4)(b)는 웨이블릿 구성 과정과 이 과정을 통해 형성된 오차 트리(error tree)를 보여준다.

공간 객체의 분포와 공간 질의의 형태는 임계 모델을 선택하는데 중요한 요소이다. 공간 객체의 분포는 객체가 밀접해 있는 지역에서 희박한 지역으로 점진적으로 분포되어 있다. 따라서, 인접한 두 데이터 값이 크게 차이 나지 않는 경우가 많다. 또한, 대부분의 사용자는 어떤 윈도우 영역 안에 교차하거나(overlap), 포함하는(contain) 객체들에 대한 연산이 주로 이루어지기 때문에 각 데이터 값들의 상대 오차 보다는 그 질의 영역 전체에 대한 오차가 중요하다. 그러므로, 이 논문에서는 계산이 매우 간단하고 전체 오차를 최소화하는 결정적 임계값을 사용하여 유지될 계수들을 선택한다.

#### 5.2.3 웨이블릿 계수의 축소

한정된 저장공간에 히스토그램을 유지하기 위해서는 각 버켓에 유지할 웨이블릿의 수를 결정해야 한다. (1)에 의해 요구되는 버켓의 수가 결정되면 평균 유지되는 웨이블릿의 개수를 구할 수 있다. 그러나, 모든 버켓이 평균 개수만큼의 웨이블릿 계수를 유지하는 것은 높은 오차를 발생시킬 수 있다. 왜냐하면, 각 버켓의 공간 편중도가 모두 다르고, 점유하는 영역의 크기도 달라서 0이 아닌 계수들의 수도 상당히 다르게 되기 때문이다. 점유 영역의 크기가 크면서 공간 편중도가 높은 버켓일 경우, 제거되는 웨이블릿 계수들의 평균제곱오차가 커지게 된다. 따라서, 이 버켓 영역과 교차하는 질의가 주어질 때, 선택율의 오차는 커지게 된다. 반면, 작은 점유 영역과 0에 가까운 편중도를 갖는 버켓 일 경우, 저장공간을 낭비하는 결과를 가져온다. 따라서, 우리는 각 버켓에 유지되는 계수의 수를 차별하게 설정한다. 편중도가 0에 가까울수록 격자의 빈도값은 균일하다. 따라서, 적은 수를 할당하여도 오차가 적게 발생한다. 또한, 버켓의 영역 크기가 클수록 많은 계수들이 생성되므로 좀 더 많은 계수를 할당하여야 한다. 즉, 할당 계수의 수와 편중도, 버켓의 영역 크기는 비례적인 관계를 가진다. 따라서, 버켓 $B_i$에 유지될 계수의 수 $|W_{Ai}|$는 다음과 같이 결정한다(coeff_size는 총 유지 계수의 수).

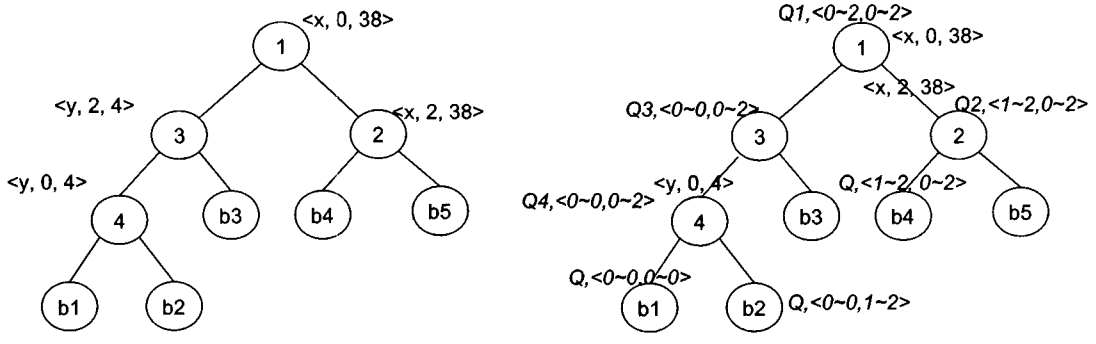
$$|W_{Ai}| = \frac{B_i.skew \times |B_i|}{\sum B_i.skew \times |B_i|} \times coeff\_size \qquad (4)$$

#### 5.3 선택율 추정

임의의 질의 Q $<q_{xl}, q_{yl}, q_{xu}, q_{yu}>$가 주어질 경우, 요약함수의 정의에 따라 질의의 크기를 변경하여야 한다. MW 히스토그램의 요약함수는 객체의 중심점을 포함하는 격자에 카운트하기 때문에, 질의를 처리하기 전에 평균 객체의 높이 $H_{avg}$와 너비 $W_{avg}$만큼 질의의 확장하여야 한다. 확장된 질의 Q' $<q'_{xl}, q'_{yl}, q'_{xu}, q'_{yu}>$는 다음과 같다.

$$q'_{xl} = q_{xl} - W_{avg},\ q'_{yl} = q_{yl} - H_{avg},$$
$$q'_{xu} = q_{xu} + W_{avg},\ q'_{yu} = q_{yu} + H_{avg}$$

확장된 질의 Q'는 이진 분할 트리의 루트 노드에서 단말 노드까지 진행하면서 노드의 분할 위치를 기준으로 분할한다. 단말노드에 의해 분할된 질의는 계산될 버켓과 만나게 되고, 그 버켓의 웨이블릿 요약정보를 통해 선택율을 계산하게 된다. 예를 들어, (그림 5)는 질의에 교차하는 버켓을 찾는 과정을 보여주고 있다. 만일, 확장된 윈도우 질의 Q'가 <0~2, 0~2>에 걸쳐 존재한다면, 이 질의는 이진 트리를 거치면서 (그림 5)(b)에서 보듯이 질의를 분해하여 버켓들을 찾는다. 먼저 이진 분할 트리의 루트에서 x축으로 인덱스 1에 의해 질의를 분해한다. 즉, 질의는 Q2, Q3로 분할된다. 이 과정을 이진 분할 트리의 단말노드까지 진행 후 최종 분할된 윈도우 $Q_i$와 질의가 수행될 버켓들을 선택한다. 분해된 각 질의는 다시 공간 탐색 기법을 통해 질의 정렬 시퀀스로 변화한 다음, 각 버켓의 웨이블릿 요약정보를




(a) 이진분할 트리 (b) 질의 Q에 대한 버켓 검색

(그림 5) 임의의 질의 Q에 대한 선택율 추정

통해 질의 정렬 시퀀스에 포함된 원본데이터들을 복원하여 합한다. 각 분할된 질의에 대한 복원값들을 모두 합하여 임의의 질의 Q에 대한 선택율을 얻는다.

### 5.4 MW 히스토그램의 동적 갱신

분할 히스토그램의 최대 단점은 동적 갱신에 의해 어떤 버켓의 공간 편중도가 변하였을 경우 전체 공간을 다시 분할해야 하는 오버헤드가 있다. 이것은 온라인 질의 시에 느린 응답시간을 갖게 되는 원인이 된다. 그러나, 동적 갱신에 대해 제안된 MW 히스토그램은 계층적 구조를 통해서 빠르게 대처할 수 있다. 제안된 히스토그램의 구조는 공간을 분할 전의 버켓의 공간 편중도가 저장된 내부 노드들로 구성되는 이진 분할 트리 구조를 갖고 있다. MW 히스토그램은 이진 분할 트리의 단말노드에 의해 분할된 버켓들을 통합하고 재분할하여 갱신에 대해 적절하게 조치를 취한다. 다음은 이웃 버켓간의 통합되는 기본 조건이다.

① 어떤 버켓 $B_i$의 공간 편중도(skew)가 갱신에 의해 그 상위 노드인 분할 노드 $snode_k$의 공간 편중도보다 크다.
② 한 분할노드에 의해 분할된 두 버켓의 공간 편중도 합이 그 분할 노드의 공간 편중도보다 크다.

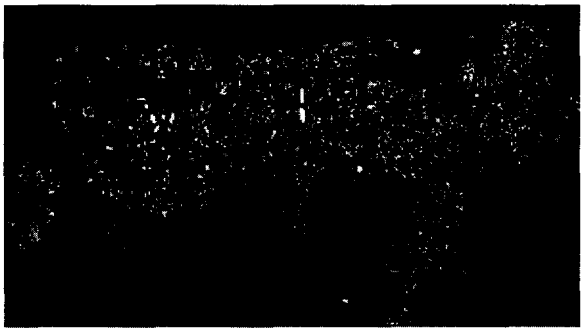
위 조건 만족할 경우, 상위 분할 노드에 의해 생성된 버켓들을 통합하여, 분할 알고리즘을 다시 수행한다. 그런 후, 각 버켓에 웨이블릿 변환을 수행하고 유지될 계수들을 선택한다. 단, 통합 전과 후의 저장공간은 같아야 한다. 만일, 재 분할에 의해서도 위 통합 조건을 만족하면, 그 상위 노드의 상위 노드에 의해 분할 된 버켓들과 통합하여 2번의 재분할을 수행한다. 이 때도 마찬가지로 분할 전후의 저장공간은 같아야 한다.

## 6. 실험 평가

이 실험의 목적은 저장공간에 비해 상대적으로 큰 도메인을 갖는 데이터베이스나 저장공간이 매우 제약된 디바이스에서 제안된 기법의 우수성을 평가하는 것이다.

### 6.1 실험 환경과 오차 측정

이 논문에서 제안된 방법의 정확도를 측정하기 위해서 실제 데이터를 사용하여 여러 요인을 변화시키면서 평가한다. 실험 환경으로는 384M의 주 메모리, 60G 하드디스크를 가진 WindowXP 운영체제하의 Intel Pentium Ⅲ 1.0GHz PC에서 실험하였다. 실험 데이터로 11,000개의 서울시 중구의 빌딩 데이터를 사용하였다. 이 빌딩 객체들의 분포는 (그림 6)과 같다. 이 실험데이터 집합의 공간 도메인은 <(196500, 449500), (202300, 452100)>이며, 실험환경과 유사하게 조성하기 위해 해상도 64을 설정하여 저장공간에 비해 큰 도메인을 갖게 하였다. 실험 파라미터로써 저장공간은 60~720 단위 공간만큼 변화시켰고, 질의의 크기는 전체 공간 영역의 5%~20%까지 변화시켰다. 또한, 요구되는 버켓의 수와 저장공간 크기를 다르게 설정한 MW 히스토그램을 생성하여 실험하였다.



(그림 6) 실험데이터의 공간 분포

MW0~MW2는 유지되는 버켓의 수가 {0.3, 0.5, 0.7}×M/6 되게 변환시킨 히스토그램이다. 이것은 버켓 수와 웨이블릿 수의 상관 관계를 평가하는 것이며, 최선의 비율을 추정할 수 있다. 우리는 실험 결과로서 10개의 같은 크기의 질의의 평균을 취하였고 추정된 결과와 비교하였다.

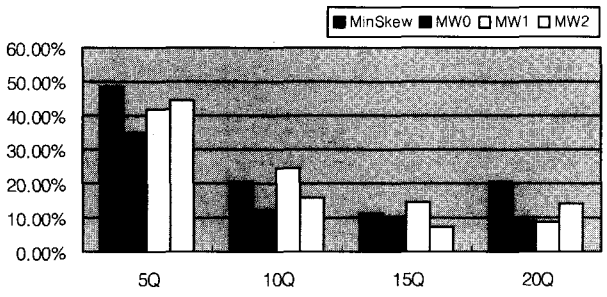
추정치의 정확도를 측정하기 위해서 다음과 같이 정의된 상대 오차( $e_r$)를 사용하였다.

$$e_r = \frac{|d - \hat{d}|}{d} \times 100\% \quad (5)$$

( $d$ : 실제 결과 크기, $\hat{d}$ : 추정된 결과 크기)

### 6.2 질의에 따른 평가

이 논문의 목적은 공간 분포를 적은 저장공간에서 유지함과 동시에 타당한 선택율을 얻는 것이다. 논문의 타당성을 실험하기 위해 저장공간이 360인 MinSkew 히스토그램과 180인 MW0~MW2인 MW 히스토그램을 비교하였다. (그림 7)에서 MW0~MW2은 절반의 저장공간에도 불구하고 MinSkew 히스토그램보다 비슷하거나 낮은 상대 오차를 보여주고 있다.




(그림 7) 선택율의 상대 오차 비교

이 그림에서 질의에 대한 MW0~MW2의 상대 오차 변화를 살펴보면, MW0의 경우에 크기가 큰 질의에 대한 상대 오차의 크기가 비슷하다. 또한, MW1은 질의 크기가 커

질수록 상대 오차는 선형적으로 감소된다. 반면, MW2는 20% 질의에서 MW0과 MW1보다 높은 오차를 가진다. 크기가 작은 질의는 상대적으로 큰 버켓 영역에 포함되기 쉽기 때문에 잘못된 카운팅(false counting)할 확률이 크다. 따라서, 높은 오차를 발생시킨다. 그러나, 유지되는 웨이블릿 계수의 수가 큰 MW0는 버켓내의 분포 정보를 웨이블릿 요약정보로 함축되어 저장하고 있기 때문에 잘못된 카운팅의 확률을 줄여준다. 그리고, 크기가 큰 질의는 하나의 버켓을 완전히 포함하거나, 대부분을 포함하기 때문에 좀 더 정확한 선택율을 얻을 수 있다.

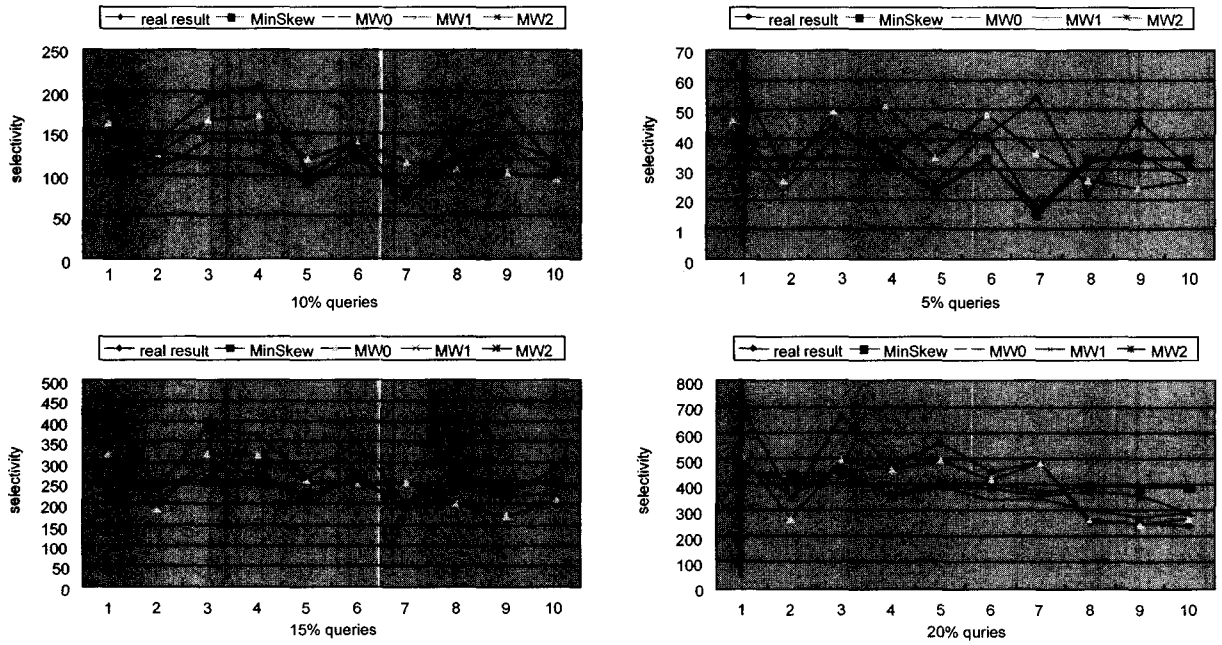
그러나, MW2는 MinSkew보다 작은 수의 버켓과 MW0보다 유지되는 웨이블릿 계수의 수가 적기 때문에 큰 질의에 포함되는 버켓의 수가 적을 뿐만 아니라, 버켓 내의 분포 정보를 정확하게 반영하기 어렵다. 그렇기 때문에, (그림 7)의 20% 질의에 대한 MW2의 상대 오차가 다른 MW히스토그램보다 조금 높은 오차를 갖게 된다. 그러나, MinSkew 히스토그램보다는 낮은 오차를 갖는다. 이 결과로 제안된 MW 히스토그램이 공간 도메인이 크거나, 저장공간의 제약이 매우 심한 시스템에서 상대적으로 큰 버켓 영역내의 편중된 분포를 해결할 수 있음을 보인다.

(그림 8)은 저장공간의 크기가 60 단위인 비교적 작은 저장공간에서 각 히스토그램의 선택율 곡선을 보여 주고 있다. MinSkew 히스토그램은 유지되는 버켓의 수가 10개이다. 적은 수의 버켓 정보를 통해 선택율을 추정하기 때문에 선택율은 같은 크기의 질의에 대해 거의 같은 값을 갖는다. 실제 선택율 곡선과 비교할 때 같은 크기의 질의에서 MinSkew 히스토그램의 곡선 변화는 거의 없다. 반면, MW0~MW2의 선택율 곡선은 매우 민감한 변화를 갖는다. 특히, 버켓당 유지되는 웨이블릿 계수의 수가 큰 MW0의 경우에 다른 히스토그램보다 더 민감한 반응을 보인다. 이것은 유지 웨이블릿 계수가 많은 히스토그램이 다른 히스토그램보다 더 많은 정보를 유지하고 있다는 사실을 증명한다.

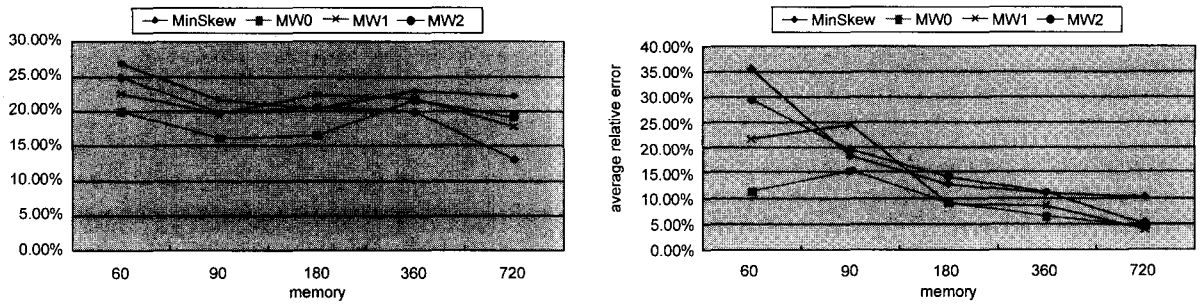
그러나, 웨이블릿 계수의 수가 많다고 좋은 선택율을 얻는 것은 아니다. MW 히스토그램은 공간 편중도에 따라 유지되는 버켓의 수와 웨이블릿 계수의 수를 적절히 조절해야 추정 결과의 상대 오차를 줄일 수 있다.

### 6.3 저장공간에 따른 평가

(그림 9)는 각각 질의 5%와 20%에서의 저장공간에 따른 MW0~MW2의 상대 오차을 보여주고 있다. 대체로, 히스토그램은 저장공간이 증가할수록 낮은 오차를 보이고 있다. 5%의 질의일 경우 교차하는 버켓의 수가 적기 때문에 저장공간 크기에 많은 영향을 받지 않는다. 그러나, 20% 질의에서의 상대 오차 곡선은 저장공간 크기가 증가할수록 변화가 크다. 특히, 저장공간이 60일 때의 MW 히스토그램은 MinSkew 히스토그램의 상대 오차보다 낮은 오차를 갖는다. 이 결과는 작은 저장공간에서도 MW 히스토그램이 더 많은 정보를 유지하고 있음을 보인다. MW 히스토그램 중에 버켓당 유지되는 웨이블릿 계수의 수가 큰 MW0는 다른 히스토그램보다 매우 낮은 오차를 가진다. 즉, 유지되는 웨이블릿 계수의 수가 클수록 작은 저장공간에서 좋은 선택율을 얻을 수 있다. 그러나, MW 히스토그램이 유지되는 웨이블릿 계수의 수가 클수록 좋은 선택률을 얻고 있지만, 선택률을 추정하기 위해서는 질의를 만족하는 버켓내의 웨이블릿 변환과정을 필요로 하므로, 상대적으로 높은 런타임 오버헤드를 초래하게 된다. 따라서, 적당한 버켓수와 웨




(그림 8) 저장공간이 60일 때 질의에 따른 선택율 곡선




(그림 9) 5% 질의와 20% 질의에 대한 평균 상대 오차

이블릿 계수를 결정하는 기법이 필요하며, 이러한 기법은 향후 연구에서 다룰 것이다.

## 7. 결 론

이 논문에서는 큰 공간 도메인을 갖는 데이터베이스에서 공간 객체들의 분포를 고려하여 적은 저장공간에 요약정보를 형성하여 타당한 선택율을 얻기 위한 추정 기법에 관하여 다루었다. 큰 공간 도메인을 갖거나, 저장공간 크기가 매우 제약적인 모바일 환경의 소형 데이터베이스에서의 선택율 추정은 기존 기법의 효과에 의한 상대 오차가 낮은 선택율을 얻을 수 없다. 따라서, 이 논문에서는 이런 문제점을 해결하기 위해 여러 연구에서 타당성이 인정된 웨이블릿 변환기법과 공간 분포를 고려한 분할 히스토그램 기법을 결합하여 작은 공간에 압축 요약정보를 유지하고 타당한 선택율을 얻을 수 있는 MW 히스토그램을 제안하였고 이를 이용한 선택율 추정 기법을 제안하였다. 실험을 통해, 단위공간 60인 작은 저장공간에서 MW 히스토그램이 MinSkew 히스토그램보다 좋은 선택율을 얻음을 보였다. 또한, 적은 저장공간일수록 버켓당 유지되는 웨이블릿 계수의 수가 크면 좋은 선택율을 얻을 수 있다. 실험 결과를 통해 다음과 같은 잇점을 입증해 주고 있다. ① 저장공간이 매우 작은 시스템에서 MW 히스토그램은 높은 압축과 타당한 선택율을 얻는다. ② 동적 갱신에 매우 유연하게 대처할 수 있다.

실험 결과를 통해 우리는 향후에 논의될 몇 가지 문제점을 살펴보면, 먼저 공간 편중도와 유지되는 버켓의 수와의 상관관계 분석이 요구된다. 즉, 공간 분포의 편중정도에 따른 적절한 버켓의 수를 결정하기 위한 분석이 필요하다. 또한, 개개의 오차까지 고려하는 확률적 모델 적용하여 압축하는 연구가 필요하다.

## 참 고 문 헌

[1] 조문증, "데이터베이스 시스템에서 웨이블릿 변환에 기반한 통합 요약정보의 관리", 전자전산학과 전산학전공, 한국과학기술원 박사논문, 2001.

[2] 임정옥, 조숙경, 배해영, "시간적 제약을 갖는 공간 질의 처리를 위한 실시간 연산 후배치 기법", 정보과학회논문지 : 컴퓨팅의 실재, 제7권 제3호, pp.193-210, June, 2001.

[3] 문현수, 황환규, "공간 영역 질의의 선택율 추정을 위한 향상된 면적 균등 분할 방법", Journal of Telecommunications and Information, Vol.4, 2000.

[4] 정지훈, 홍석진, 배진욱, 안성준, 송병호, 이석호, "다차원 히스토그램에서 범위 질의의 선택도에 대한 오차 추정", 정보과학회 2001년 추계학술대회, Vol.28, No.2, pp.211-213.

[5] 김홍연, 배해영, 다차원 히스토그램을 이용한 공간 위상 술어의 선택도 추정 기법, 정보처리논문지, 제6권 제4호, pp. 841-850, April, 1999.

[6] Poosala et al., "*Improved Histograms for Selectivity Estimation of Range Predicates*," In Proc. ACM SIGMOD Int. Conf. on Management of Data, pp.294-305, 1996.

[7] Yossi Matias, Jeffrey Scott Vitter, Min Wang, "*Wavelet-Based Histograms for Selectivity Estimation*," In Proc. ACM SIGMOD Int. Conf. on Management of Data, pp.448-459, 1998.

[8] Swarup Acharya, Viswanath Poosala, Sridhar Ramaswamy, "*Selectivity estimation in spatial databases*," In Proc. ACM SIGMOD Int. Conf. on Management of Data, pp.13-24, 1999.

[9] Vitter, Wang, "*Approximate Computation of Multidimensional Aggregates of Sparse Data using Wavelets*," In Proc. ACM SIGMOD Int. Conf. on Management of Data, pp. 193-204, 1999.

[10] A. Aboulnaga, J. Naughton, "*Accurate estimation of the cost of spatial selections*," In Proceedings of the IEEE International Conference on Data Engineering (ICDE), pp.123-134, 2000.

[11] Yossi Matias, Jeffrey Scott Vitter, Min Wang, "*Dynamic Maintenance of Wavelet-Based Histograms*," The VLDB Journal, pp.101-110, 2000.

[12] L. Getoor, B. Taskar, D. Koller, "*Selectivity estimation using probabilistic models*," In Proc. ACM SIGMOD Int. Conf. on Management of Data, 2001.

[13] Nikos Mamoulis, Dimitris Papadias, "*Selectivity estimation of complex spatial queries*," In Proc. Int. Symp. on Spatial and Temporal Databases, pp.156-174, 2001.

[14] Min Wang, Jeffrey Scott Vitter, Lipyeow Lim, Sriram

Padmanabhan, "*Wavelet-based cost Estimation for Spatial Queries*," In Proc. Int. Symp. on Spatial and Temporal Databases, pp.175-196, 2001.

[15] Ning An, Zhen-Yu Yang, Sivasubramaniam, A., "*Selectivity estimation for spatial joins*," In Proceedings of the IEEE International Conference on Data Engineering (ICDE), pp.368-375, 2001.

[16] C. Sun, D. Agrawal, A. El Abbadi, "*Selectivity for spatial joins with geometric selections*," Proc. of EDBT, pp.609-626, 2002.

[17] Yong-Jin Choi, Chin-Wan Chung, "*Selectivity estimation for spatio-temporal queries to moving objects*," In Proc. ACM SIGMOD Int. Conf. on Management of Data, pp. 440-451, 2002.

[18] Tao, Y., Sun, J., Papadias, D., "*Selectivity Estimation for Predictive Spatio-Temporal Queries*," In Proceedings of the IEEE International Conference on Data Engineering (ICDE), pp.417-428. 2003.

[19] Sun, C., Agrawal, D., El Abbadi, A., "*Exploring spatial datasets with histograms (full version)*," Technical Report, Computer Science Department, University of California, santa Barbara, 2001.

[20] Antonios Deligiannakis, Nick Roussopoulos., "*Extended Wavelets for Multiple Measures*," ACM SIGMOD 2003, pp. 229-240, June, 2003.

[21] Kaushik C., Minos G., Rajeev R., Kyuseok S., "*Approximate query processing using wavelets*," The VLDB Journal, pp. 199-223, 2001.

[22] Minos G., Phillip B.G ., "*Wavelet Synopses with Error Guarantees*," ACM SIGMOD, Jine 4-5, Madison, Wisconsin, USA, 2002.

[23] Yannis E. Ioannidis, "*Query Optimization*," ACM survey, 1996.

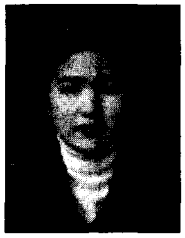
[24] E. Clementini and P. Di Felice, "*A Comparison of Methods for Representing Topological Relationships*," Information Sciences 3, pp.149-178, 1995.

[25] Jin, N. An, A. Sivasubramaniam, "*Analyzing Range Queries on Spatial Data*," In Proceedings of the IEEE International Conference on Data Engineering (ICDE), pp.525-534, 2000.

[26] S. Muthukrishnan, Viswanath Poosala, Torsten Suel, "*On Rectangular Partitionings in Two Dimensions : Algorithms, Complexity, and Applications*," 7th International Conference on Database Theory, ICDT'99, 1999.

[27] Jin Yul Lee, Jeong Hee Chi, Keun Ho Ryu, "*Spatial Selectivity Estimation Using Wavelet*," Proceddings of the 4th International Symposium on Advanced Intelligent Systems, ISSN 1738-0073, ISIS2003, pp.459-462, Sepmtember, 2003.

[28] Jeong Hee Chi, Jin Yul Lee and Keun Ho Ryu, "*Selectivity Estimation for Spatial Databases*," Asian Conference on Remote Sensing & International Symposium on Remote Sensing (ISRS), November, 2003.

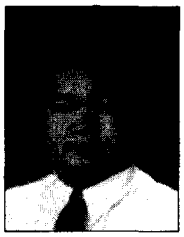


**지 정 희**

e-mail : jhchi@dblab.chungbuk.ac.kr
1999년 충주대학교 전자계산학과
2001년 충주대학교 대학원 전자계산학 석사
2003년 충북대학교 대학원 전자계산학 박사수료
관심분야 : 시공간 데이터베이스, Temporal GIS, 시공간 질의 최적화, 시공간 색인기법, 이동객체 관리 기법

**이 진 열**

e-mail : jinylee@pointi.com
2002년 충북대학교 토목공학과
2004년 충북대학교 대학원 정보산업공학 석사
2004년~현재 포인트 아이 근무
관심분야 : 시공간 데이터베이스, 질의 최적화, 시공간 색인기법



**김 상 호**

e-mail : kimsh@dblab.chungbuk.ac.kr
1997년 충북대학교 컴퓨터과학과
1999년 충북대학교 대학원 전자계산학 석사
2004년 충북대학교 대학원 전자계산학 박사
관심분야 : 시공간 데이터베이스, Web Visualization, Component GIS, 이동객체 관리기법

**류 근 호**

e-mail : khryu@dblab.chungbuk.ac.kr
1976년 숭실대학교 전자계산학과
1980년 연세대학교 공학대학원 전자계산학 석사
1988년 연세대학교 대학원 전자계산학 박사
1976년~1986년 육군군수지원사전산실(ROTC 장교), 한국전자통신연구소(연구원), 한국방송통신대 전산학과(조교수) 근무
1989년~1991년 Univ. of Arizona 연구원(TempIS Project)
1986년~현재 충북대학교 전기전자 및 컴퓨터공학부 교수
관심분야 : 시간 데이터베이스, 시공간 데이터베이스, Temporal GIS, 객체 및 지식베이스 시스템, 지식기반 정보검색시스템, 데이터마이닝, 데이터베이스 보안 및 Bio-Informatics