

根研究会学術特別賞講演

パーソナルコンピューターを利用した 画像解析による根量把握

木村和彦

(東北大・院・農学研究科)

謝辞

研究全般:

山崎慎一(東北大), 菅野均志(東北大), 八巻邦次(東北大)

一緒に研究した学生:

中丸康人(元JICA), 菊地誠次(青森農試), 奥村弥琢(東北放送)

画像解析:

Wayne Rasband(NIH), 東北大学BBSでのTurboPASCAL UG
東北大学大型計算機センターの皆さん

根の研究会:

森田茂紀(東京大学), 阿部淳(〃), 山内章(名古屋大学),

ユーザー:

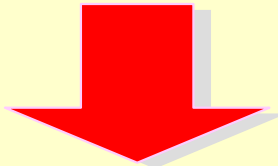
郭 洙康(IRRI), 近藤始彦(農研セ), 岡田佳奈子(山形大学)
有馬進(佐賀大学), 錦織正智(北海道立林試),

内容

- なぜこの研究が必要だったか？
- 正確に長さを測る
- 直径毎の長さを測る
- より正確に直径毎の長さを測る
- デモンストレーション

なぜこの研究が必要だったか？

- 効率的な施肥管理 ⇒ 局所施肥、緩効性肥料
根の局在化が関係ある？
- 根の測定の問題
 - ◆ 根の測定は手間がかかる(切ってランダムに置く)
 - ◆ 正確さが疑問(本当にランダムに置けたか？)
 - ◆ 直径毎の長さは不可能
- 画像解析
 - ◆ 市販の画像解析ソフトは高価
 - ◆ フリー(無料)のNIH Image(free)では不正確
 - ◆ Pan and Bolton(1991)の方法は原理的に不正確



自分で書こう！マクロで記述すれば楽！

画像解析システム

- 画像入力

- スキャナー (ScanJet 4c, Hewlett Packard)
- 透過原稿読取ユニット(Hewlett Packard)
- 染色した根を300dpiで画像入力

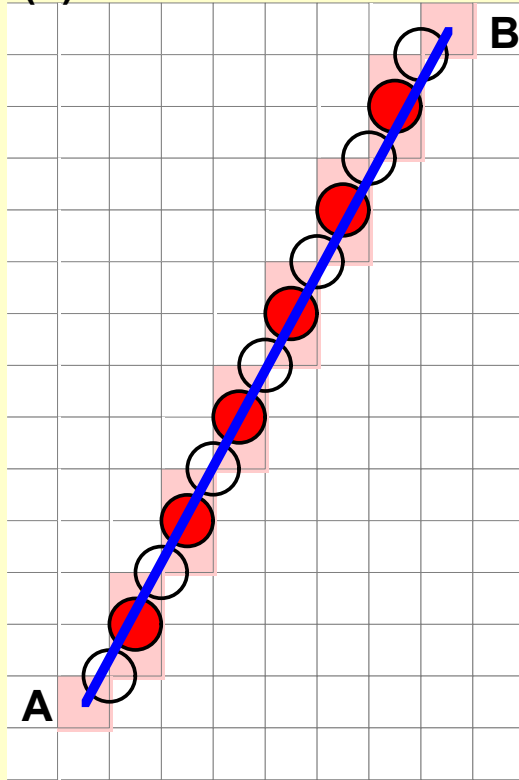
- 画像解析

- コンピューター(Power Macintosh 8500/150, Apple)
- 画像解析ソフト(NIH Image 1.60)

(現在では、windowsでも動作)

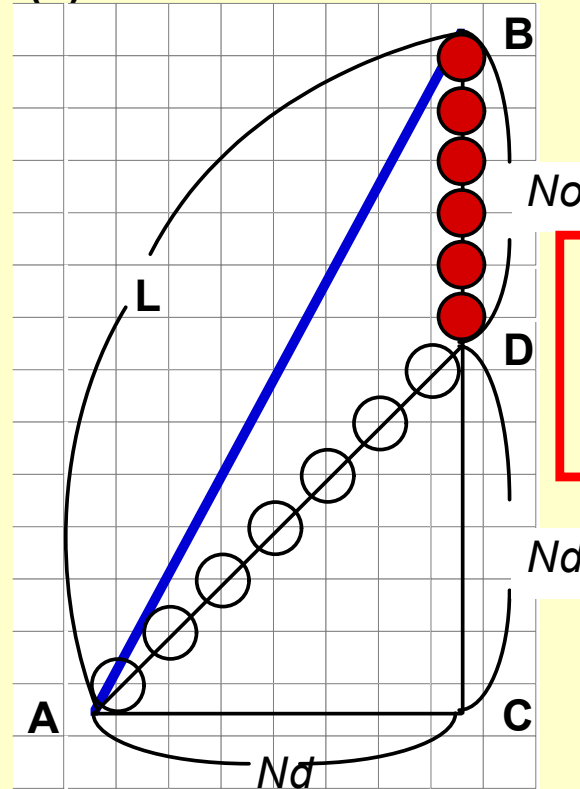
一本の直線の長さを考える

(1)



- 直線ABを表わす画素
- 画素の斜めの連結
- 画素の縦横の連結

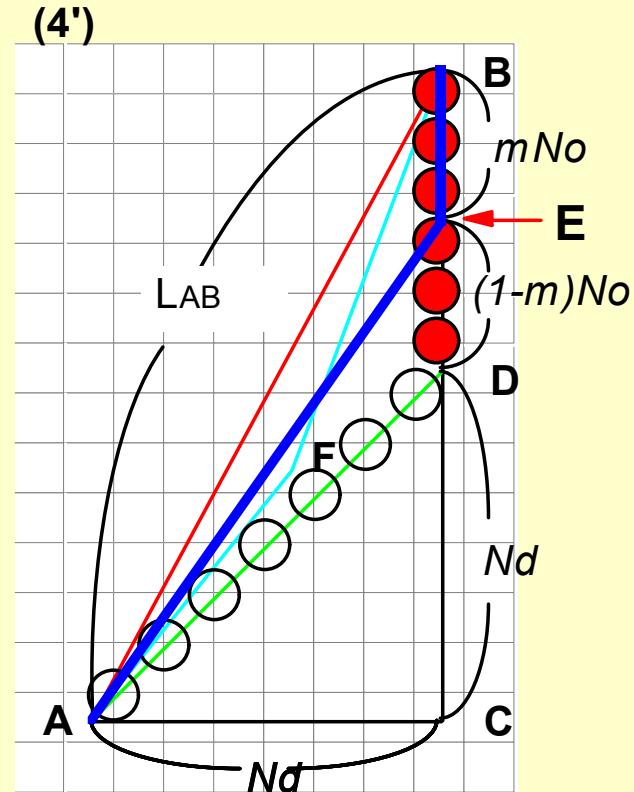
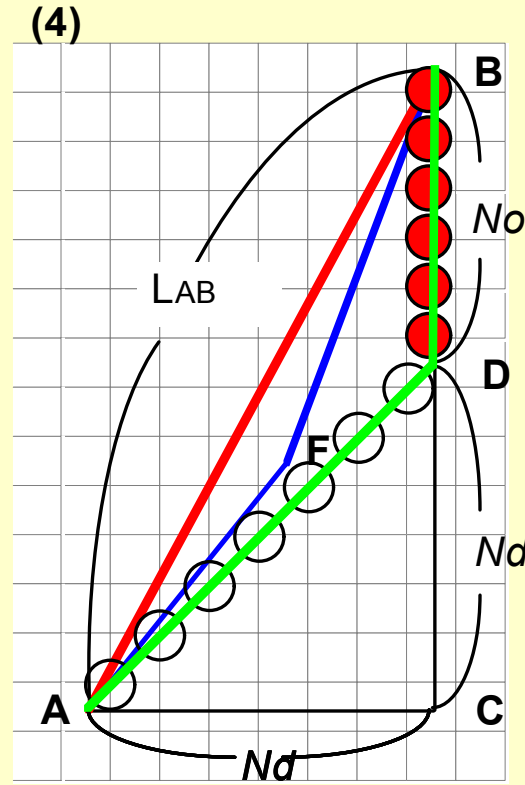
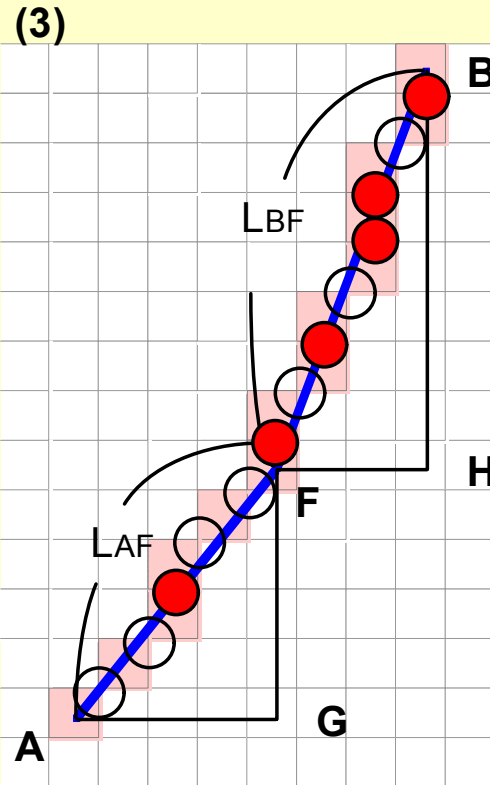
(2)



ピタゴラスの定理より
$$L = (Nd^2 + (Nd + No)^2)^{1/2}$$

Nd : 画素の斜めの連結数
 No : 画素の縦横の連結数

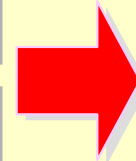
二本の直線の長さを考える



$L = Nd + No$
 は直線BCの長さ
 ⇒過小評価
 ⇒補正する
 ↓ ↓ ↓
 ライン交差法

$L = (Nd^2 + (Nd + No)^2)^{1/2}$
 は直線ABの長さ
 ⇒過小評価

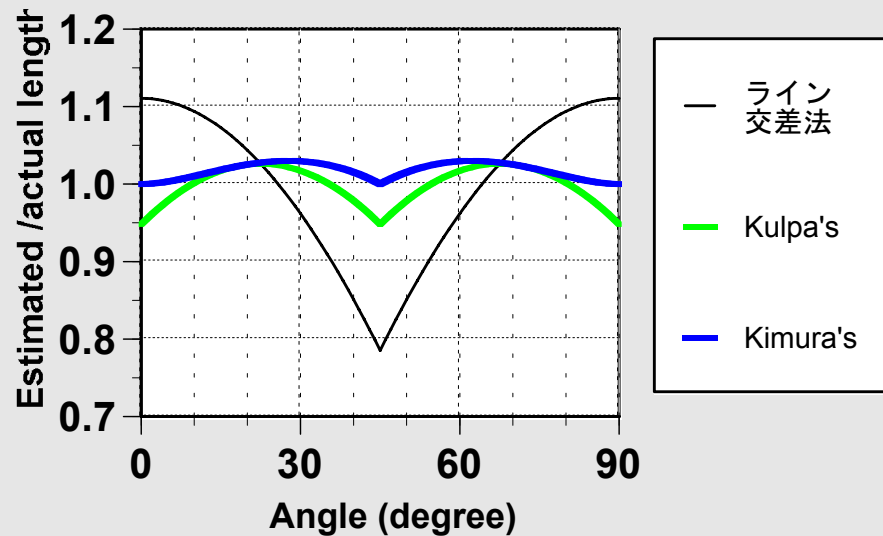
$L = \sqrt{2}Nd + No$
 は直線AD+DBの長さ
 ⇒過大評価



$LAF + LFB = LAE + LEB$
 となる点Eが必ずある。
 $(0 \leq m \leq 1)$

ランダムで二本より多い時は
 m を固定(0.5)して推定

長さを求める精度

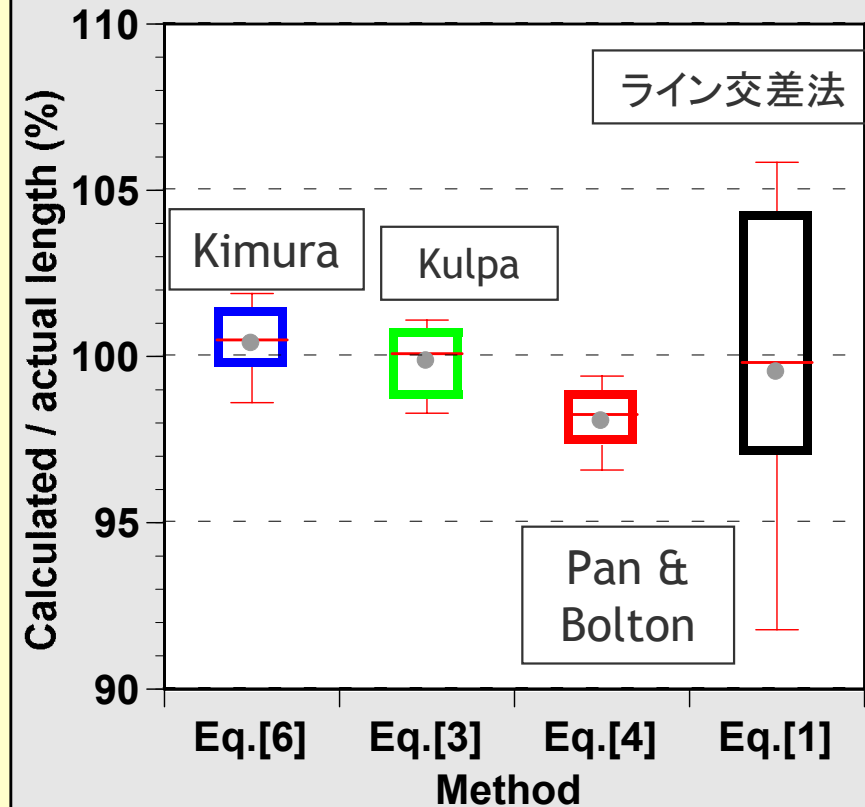


1本の直線の場合の誤差

ライン交差法 :-21 ~ +11%

Kulpa :-5.2 ~ +2.6%

Kimuraの方法: 0 ~ +3.0%



5本のランダムな直線の場合の誤差

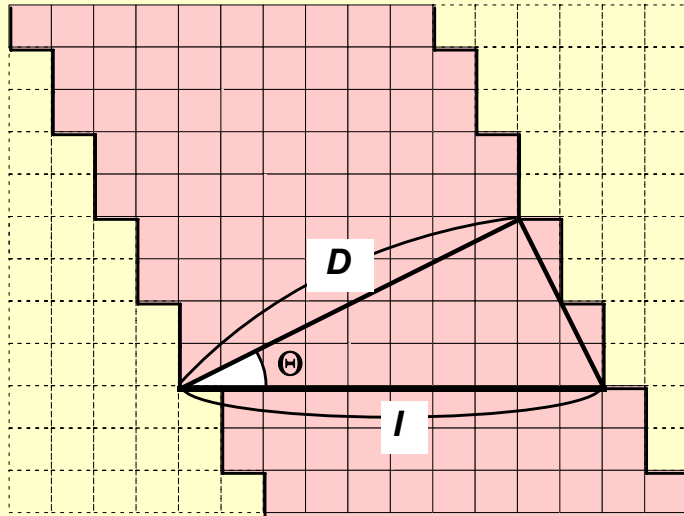
一本の時は最大3%の過大評価(それでも他より誤差は少ない!)

しかも、本数が多くなると系統誤差は無視できる!

⇒ ランダムに配置する必要がない。

直径の考え方

(1)

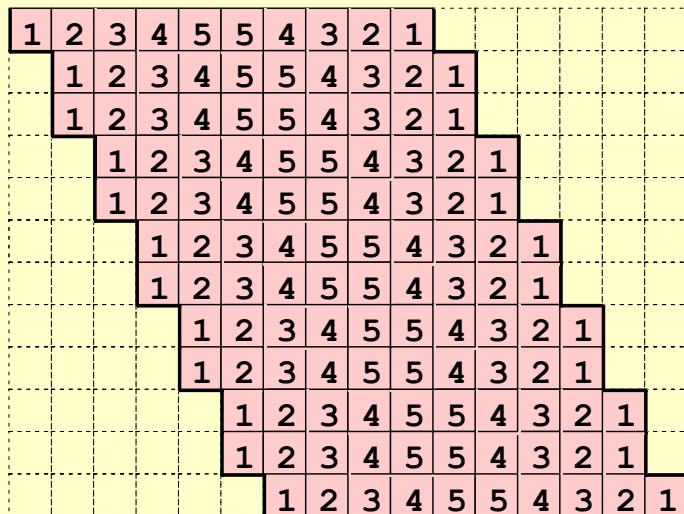


$$D = I \times \cos \Theta$$

Θ はランダムな配置を前提
 $\therefore D = I \times 0.891$



(2)



I をエッジ消去の回数(n)から推定すると

$$D = 2 \times n \times 0.891$$



本質的には長さのライン交差法と同じで、
 角度による誤差がある。

foreground pixel background pixel

より正確な直径の考え方

A. 前の方法(4近傍距離)

C	1	2	3	4	5	6	7	8	9	10	11
1	2	3	4	5	6	7	8	9	10	11	12
2	3	4	5	6	7	8	9	10	11	12	13
3	4	5	6	7	8	9	10	11	12	13	14
4	5	6	7	8	9	10	11	12	13	14	15
5	6	7	8	9	10	11	12	13	14	15	16
6	7	8	9	10	11	12	13	14	15	16	17
7	8	9	10	11	12	13	14	15	16	17	18
8	9	10	11	12	13	14	15	16	17	18	19
9	10	11	12	13	14	15	16	17	18	19	20
10	11	12	13	14	15	16	17	18	19	20	21
11	12	13	14	15	16	17	18	19	20	21	22

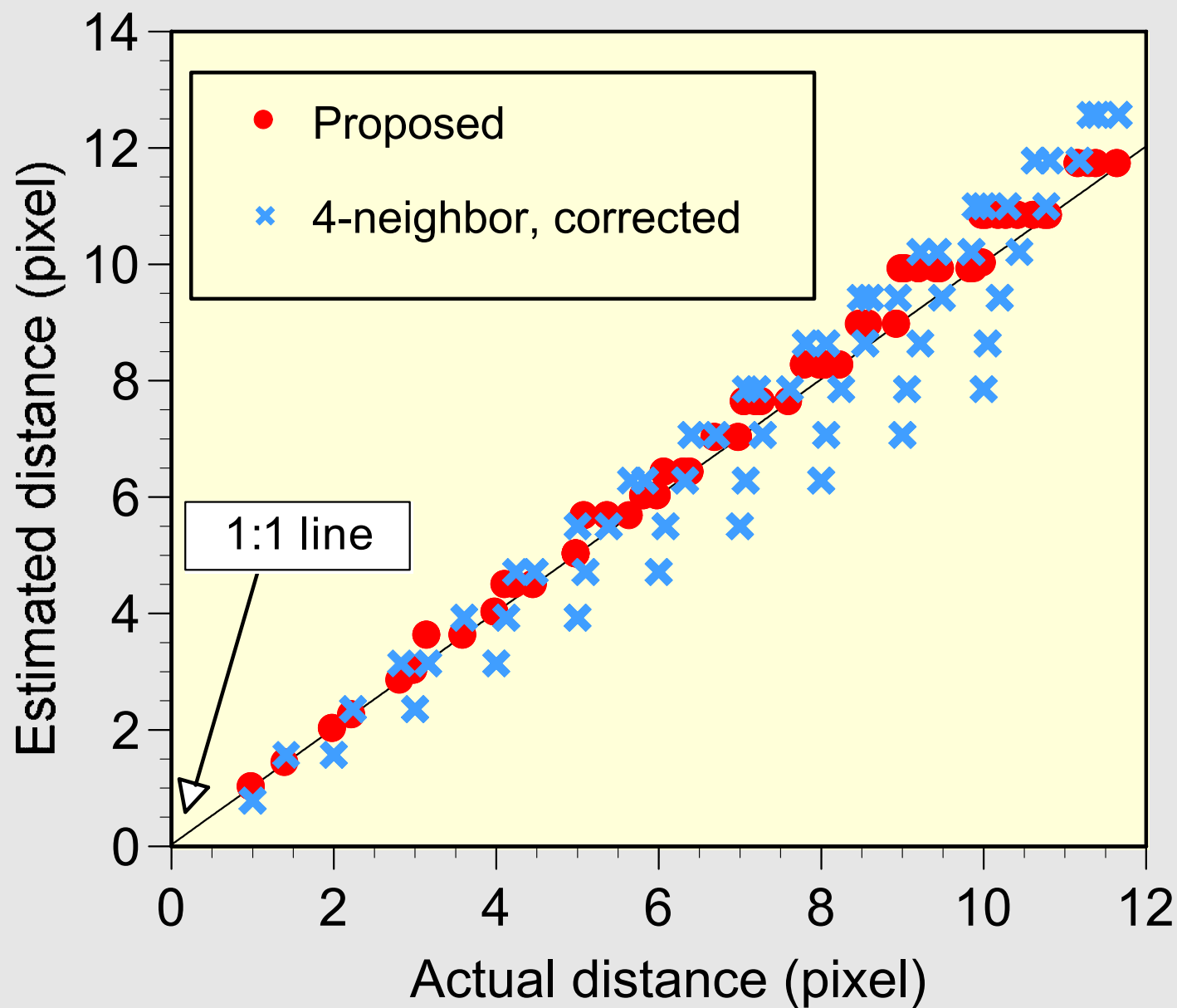
四近傍距離: 縦or横の連結数
⇒ 真の距離ではなく、誤差

B. 新しい方法(ピタゴラス距離)

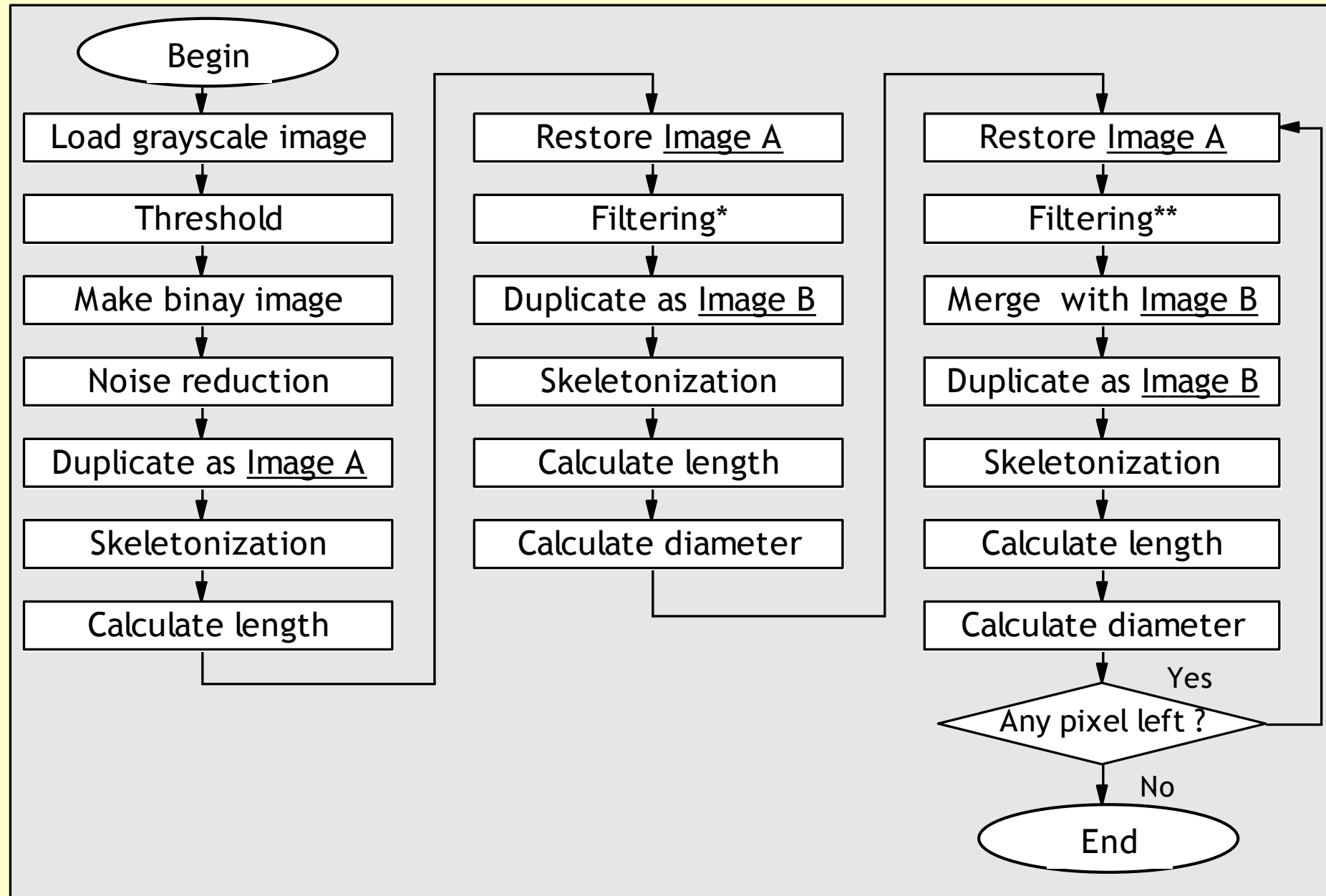
C	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.2	9.9	10.8	11.7
1.0	1.4	2.2	3.6	4.5	5.7	6.4	7.6	8.2	9.9	10.8	11.7
2.0	2.2	2.8	3.6	4.5	5.7	6.4	7.6	8.2	9.9	10.8	11.7
3.0	3.6	3.6	4.5	5.0	6.0	7.0	7.6	8.9	9.9	10.8	11.7
4.0	4.5	4.5	5.0	5.7	6.4	7.6	8.2	8.9	9.9	10.8	11.7
5.0	5.7	5.7	6.0	6.4	7.6	8.2	8.9	9.9	10.8	11.7	-
6.0	6.4	6.4	7.0	7.6	8.2	8.9	9.9	10.8	10.8	11.7	-
7.0	7.6	7.6	7.6	8.2	8.9	9.9	9.9	10.8	11.7	-	-
8.2	8.2	8.2	8.9	8.9	9.9	10.8	10.8	11.7	-	-	-
9.9	9.9	9.9	9.9	9.9	10.8	10.8	11.7	-	-	-	-
10.8	10.8	10.8	10.8	10.8	11.7	11.7	-	-	-	-	-
11.7	11.7	11.7	11.7	11.7	-	-	-	-	-	-	-

ピタゴラス距離は正確 but 計算は面倒
⇒ フィルター処理により
限定された距離(半径)のものを求める。

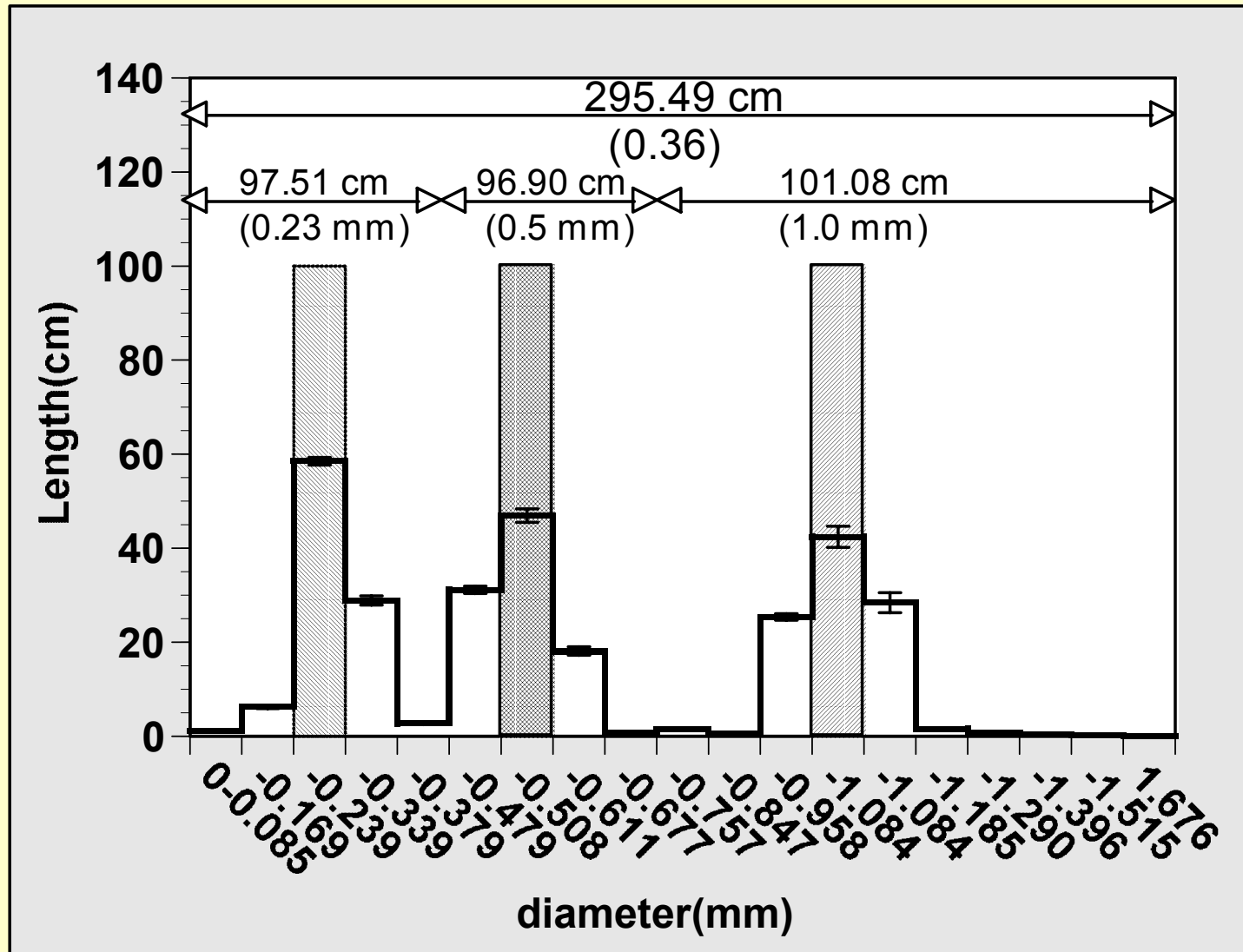
直径推定精度の比較



画像解析のアルゴリズム



針金での検証



デモンストレーション
