

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2019年1月17日(17.01.2019)



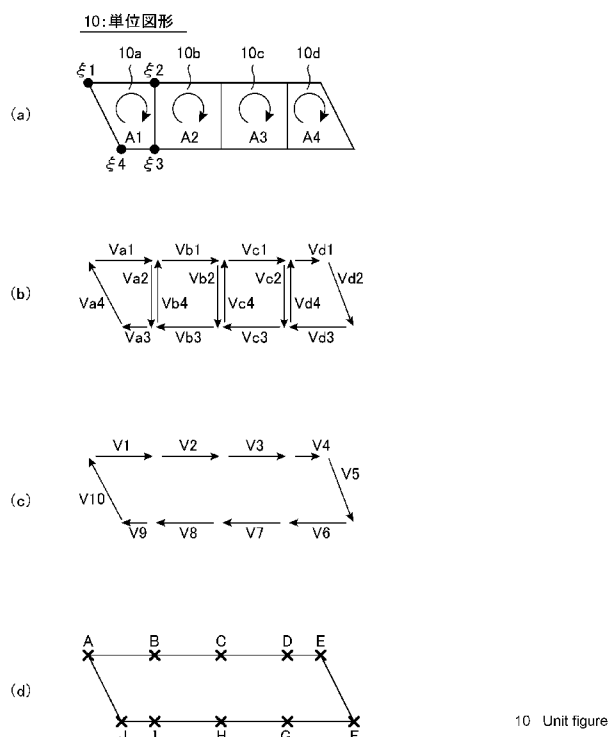
(10) 国際公開番号

WO 2019/012887 A1

- (51) 国際特許分類:
G03F 1/36 (2012.01) *G06F 17/50* (2006.01)
G03F 7/20 (2006.01) *H01L 21/027* (2006.01)
- (21) 国際出願番号: PCT/JP2018/022296
- (22) 国際出願日: 2018年6月12日(12.06.2018)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ:
特願 2017-138112 2017年7月14日(14.07.2017) JP
- (71) 出願人: 大日本印刷株式会社 (DAI NIPPON PRINTING CO., LTD.) [JP/JP]; 〒1628001 東京都新宿区市谷加賀町一丁目1番1号 Tokyo (JP).
- (72) 発明者: 大川 洋平 (OOKAWA Youhei); 〒1628001 東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内 Tokyo (JP). 下村 剛哉 (SHIMOMURA Takeya); 〒1628001 東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内 Tokyo (JP).
- (74) 代理人: 志村 浩 (SHIMURA Hiroshi); 〒1440035 東京都大田区南蒲田二丁目6番1号 ベル・シエラビル Tokyo (JP).
- (81) 指定国(表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH,

(54) Title: SHAPE CORRECTION DEVICE AND SHAPE CORRECTION METHOD FOR GRAPHICAL PATTERN

(54) 発明の名称: 図形パターンの形状補正装置および形状補正方法



(57) Abstract: Lithography based on a unit figure (10) simulates the formation of a real pattern on a real substrate. First, the unit figure (10) is divided into polygons (Fig. (a)) to define vectors (Va1 through Vd4) on individual sides of the polygons (Fig. (b)). Vectors sharing the same interval are eliminated to thereby leave vectors (V1 through V10) along the outline of the unit figure (10) (Fig. (c)). Normal end points (A through J) are defined as the start points of each vector (V1 through V10) and a contour line segment (line segment AB, etc.) is defined with mutually adjacent normal end points as

CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,
HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH,
KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY,
MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ,
NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT,
QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL,
SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA,
UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) 指定国(表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

添付公開書類:

- 一 国際調査報告 (条約第21条(3))

both ends of the contour line segment (Fig. (d)). The midpoint of each contour line segment is defined as an evaluation point to calculate the displacement of each evaluation point via simulation. The position of middle normal end points (B, C, D, G, H, I) that are not vertexes is additionally defined as a supplementary end point to secure different end points for each right and left contour line segment. The correction is performed by shifting each contour line segment in a direction perpendicular to the outline in accordance with the displacement.

(57) 要約: 単位図形 (10) に基づくリソグラフィによって実基板上に実パターンを形成するシミュレーションを行う。まず、単位図形 (10) を多角形に分割し (図(a))、多角形の個々の辺上にベクトル (V a 1 ~ V d 4) を定義する (図(b))。同一区間に重複して配置されたベクトルを除去し、単位図形 (10) の輪郭線に沿ったベクトル (V 1 ~ V 10) を残す (図(c))。各ベクトル (V 1 ~ V 10) の始点位置に通常端点 (A ~ J) を定義し、隣接する通常端点を両端とする輪郭線分 (線分 A B 等) を定義する (図(d))。各輪郭線分の中点に評価点を定義し、シミュレーションにより各評価点のずれ量を算出する。頂点ではない中間通常端点 (B, C, D, G, H, I) の位置には補充端点を追加定義して、左右の輪郭線分ごとに別個の端点を確保し、ずれ量に応じて、各輪郭線分を輪郭線に直交する方向に移動させて補正を行う。

明 細 書

発明の名称： 図形パターンの形状補正装置および形状補正方法 技術分野

[0001] 本発明は、図形パターンの形状補正装置および形状補正方法に関し、特に、リソグラフィプロセスを経て基板上に形成される図形パターンの形状を補正する技術に関する。

背景技術

[0002] 半導体デバイスの製造プロセスなど、特定の材料層に対して微細なパターンニング加工を施す必要がある分野では、光や電子線を用いた描画を伴うリソグラフィプロセスを経て、物理的な基板上に微細なパターンを形成する手法が採られている。通常、コンピュータを利用して微細なマスクパターンを設計し、得られたマスクパターンのデータに基づいて、基板上に形成されたレジスト層に対する露光を行い、これを現像した後、残存レジスト層をマスクとしたエッチングを行い、基板上に微細なパターンを形成することになる。

[0003] ただ、このようなリソグラフィプロセスを経て基板上に実際に得られる実図形パターンと、コンピュータ上で設計された元図形パターンとの間には、若干の食い違いが生じることになる。これは、上記リソグラフィプロセスには、露光、現像、エッチングという工程が含まれるため、最終的に基板上に形成される実図形パターンは、露光工程で利用された元図形パターンに正確には一致しないためである。特に、露光工程では、光や電子線によってレジスト層に対する描画を行うことになるが、その際に、近接効果（P E : Proximity Effect）によって、レジスト層に実際に描画される露光領域は、元図形パターンよりも若干広い領域になることが知られている。また、エッチング工程では、エッチングのローディング現象が生じるため、現像後のパターンとエッチング後のパターンとでは形状が異なってしまう。このエッチングのローディング現象の効果の大小は、実基板表面のレジスト層から露出した面積に応じて変化することが知られている。描画工程の近接効果や、エッチン

グ工程のローディング現象などは、いずれも元図形パターンの形状と実図形パターンの形状との間に差異を生じさせる原因となる現象であるが、このような現象の影響範囲（スケールサイズ）は、各現象で異なっている。

[0004] このような事情から、リソグラフィプロセスを含む半導体デバイスの製造工程などでは、コンピュータ上で所望の元図形パターンを設計した後、この元図形パターンを用いたリソグラフィプロセスをコンピュータ上でシミュレートし、実基板上に形成されるであろう実図形パターンの形状を推定する手順が実行される。そして、シミュレートの結果として得られる実図形パターンの形状（寸法）を踏まえ、必要に応じて、元図形パターンの形状（寸法）に対する補正を行い、この補正により得られる補正図形パターンを用いて実際のリソグラフィプロセスを実行し、実際の半導体デバイスの製造を行うことになる。

[0005] したがって、設計どおりの精密なパターンを有する最終製品を製造するためには、実際にリソグラフィプロセスを行う前に、シミュレーションによって実図形パターンの形状を正確に推定し、元図形パターンに対して適切な補正を施す必要がある。そこで、たとえば、下記の特許文献1には、元図形パターンのレイアウトを特徴づける特徴因子と、リソグラフィプロセスによって基板上に形成されるレジストパターンの寸法に影響を与える制御因子と、を入力層とするニューラルネットワークを用いて、高精度のシミュレーションを行う方法が開示されている。また、特許文献2には、2組のニューラルネットワークを利用して、シミュレーションの精度を高める方法が開示されており、特許文献3には、フォトマスクのパターンから特徴量を抽出する際に、適切な抽出パラメータを設定することにより、シミュレーションの精度を高める方法が開示されている。

先行技術文献

特許文献

[0006] 特許文献1：特開2008-122929号公報

特許文献2：特開2010-044101号公報

特許文献3：特開2010-156866号公報

発明の概要

発明が解決しようとする課題

[0007] 上述したように、元図形パターンに基づくリソグラフィプロセスによって実基板上に実図形パターンを形成する場合、事前にシミュレーションを行い、元図形パターンをそのまま用いた場合に実基板上に形成されるであろう実図形パターンの形状を推定し、当該推定結果に基づいて元図形パターンに対して必要な補正を施して補正図形パターンを得る処理が行われている。このような推定および補正を行うため、前掲の各特許文献に記載されている装置をはじめとして、様々な形状補正装置が提案されている。

[0008] 一般的な形状補正装置では、元図形パターン上の所定位置に評価点を設定し、この評価点の周囲の特徴を示す特徴量に基づいて、当該評価点についての実図形パターン上でのずれ量（プロセスバイアス）をシミュレーションによって求め、当該ずれ量に応じた補正を行うことにより、補正図形パターンが作成される。したがって、元図形パターン上の適切な位置に適切な数の評価点を設定することが非常に重要である。もちろん、評価点の密度を高めて多数の評価点を設定すれば、シミュレーションの精度が向上し、より正確な補正を行うことができる。しかしながら、評価点の数が増えれば増えるほど、シミュレーションの演算負担が増加し、シミュレーションが完了するまでの時間は増大してしまう。このため、実用上十分なシミュレーション精度を維持しつつ、できるだけ演算負担を軽減できる効率的な評価点設定を行うことが望まれている。

[0009] そこで本発明は、元図形パターンに対する補正に必要なシミュレーションを行う際に、実用上十分なシミュレーション精度を維持しつつ、できるだけ演算負担を軽減できる効率的な評価点設定を行うことが可能な図形パターンの形状補正装置を提供することを目的とする。

課題を解決するための手段

[0010] (1) 本発明の第1の態様は、元図形パターンに基づくリソグラフィプロ

セスによって実基板上に実図形パターンを形成する際に、実図形パターンが元図形パターンに一致するように、元図形パターンの形状を補正して、リソグラフィプロセスで実際に用いる補正図形パターンを作成する図形パターンの形状補正装置において、

元図形パターンを構成する個々の単位図形について、内部と外部との境界を示す輪郭線上の所定位置に評価点を設定する評価点設定ユニットと、

元図形パターンについて、評価点の周囲の特徴を示す特徴量を抽出する特徴量抽出ユニットと、

特徴量に基づいて、評価点の元図形パターン上の位置と実図形パターン上の位置とのずれ量を示すプロセスバイアスを推定するバイアス推定ユニットと、

バイアス推定ユニットから出力されるプロセスバイアスの推定値に基づいて、元図形パターンに対する補正を行うことにより、補正図形パターンを作成するパターン補正ユニットと、

を設け、

評価点設定ユニットが、

多角形からなる個々の単位図形を入力する単位図形入力部と、

単位図形を直線状の分割線で分割して複数の多角形からなる分割図形を形成する単位図形分割部と、

各分割図形について、当該分割図形を構成する多角形の輪郭線に沿って所定の順方向まわりに一周するように、多角形の個々の辺上にそれぞれ分割図形ベクトルを定義する分割図形ベクトル定義部と、

分割図形ベクトルの集合から、互いに隣接する一対の隣接多角形の重複する辺上に配置され、互いに向きが逆方向である重複ベクトル対を探索し、探索された重複ベクトル対のうちの重複区間部分を除去し、残ったベクトルにより、単位図形の輪郭線に沿って配置された複数の単位図形ベクトルを定義する単位図形ベクトル定義部と、

個々の単位図形ベクトルの始点位置に、それぞれ通常端点を定義する通常

端点定義部と、

個々の単位図形の輪郭線上において互いに隣接して配置された2つの通常端点の間に、それぞれ評価点を定義する評価点定義部と、

個々の単位図形について、多角形の頂点を構成しない通常端点である中間通常端点を探索し、探索された中間通常端点と同じ位置に補充端点を追加定義する補充端点定義部と、

を有するようにし、

パターン補正ユニットが、

個々の評価点について、単位図形の輪郭線に沿って当該評価点の一方の側に隣接する通常端点もしくは補充端点を第1の端点とし、他方の側に隣接する通常端点もしくは補充端点を第2の端点とする輪郭線分を定義し、個々の評価点についての輪郭線分を、当該評価点のプロセスバイアスの推定値に応じた補正量だけ、輪郭線分に直交する方向に移動させる移動処理を行う輪郭線分移動処理部と、

移動処理前に互いに同じ位置にあった中間通常端点および補充端点が、移動処理後に異なる2つの位置に分離した場合、それぞれの移動処理後の位置を連結する新たな輪郭線分を追加する追加処理を行う輪郭線分追加処理部と、

移動処理前に多角形の頂点を構成していた通常端点である隅部通常端点が、移動処理後に異なる2つの位置に分離した場合、分離後の2つの隅部通常端点が融合して新たな隅部通常端点となるように、分離後の隅部通常端点を含む各輪郭線分をそれぞれ伸縮する伸縮処理を行う輪郭線分伸縮処理部と、

を有するようにし、移動処理、追加処理、伸縮処理を行った後の輪郭線分の集合体を輪郭線とする単位図形によって構成される補正図形パターンを作成するようにしたものである。

[0011] (2) 本発明の第2の態様は、上述した第1の態様に係る図形パターンの形状補正装置において、

単位図形分割部が、単位図形を分割して、台形からなる分割図形を形成す

るようにしたものである。

[0012] (3) 本発明の第3の態様は、上述した第1または第2の態様に係る図形パターンの形状補正装置において、

単位図形入力部が、所定の長手方向軸に沿って伸び、互いに平行な直線状の上辺および下辺を有する細長い帯状単位図形を入力したときに、

単位図形分割部が、この帯状単位図形を、長手方向軸に沿って所定間隔で配置された複数の分割線によって分割することにより、長手方向軸に沿って一次元的に並んだ複数の分割図形を形成するようにしたものである。

[0013] (4) 本発明の第4の態様は、上述した第1または第2の態様に係る図形パターンの形状補正装置において、

単位図形入力部が、XY平面上に定義された単位図形を入力したときに、

単位図形分割部が、X軸に平行な水平分割線をY軸方向に所定間隔をあけて複数配置し、これら複数の水平分割線によって単位図形を分割することにより、X軸方向に伸びる細長い複数の帯状図形を形成し、更に、複数の帯状図形のそれぞれを、X軸に沿って所定間隔で配置されY軸に平行な複数の垂直分割線によって分割することにより、二次元的に並んだ複数の分割図形を形成するようにしたものである。

[0014] (5) 本発明の第5の態様は、上述した第1～第4の態様に係る図形パターンの形状補正装置において、

単位図形分割部が、リソグラフィプロセスで用いる描画装置のビーム最大成型サイズを設定する機能を有し、単位図形の分割に用いる分割線の配置間隔をビーム最大成型サイズ以下に設定するようにしたものである。

[0015] (6) 本発明の第6の態様は、上述した第1～第5の態様に係る図形パターンの形状補正装置において、

分割図形ベクトル定義部が、K角形からなる分割図形の各頂点に、輪郭線に沿って時計まわりもしくは反時計まわりの順に第1の頂点～第Kの頂点と番号を付し、第kの頂点を始点とし、第(k+1)の頂点(但し、k=Kの場合は第1の頂点)を終点とする第kの分割図形ベクトルを定義する処理を

、 $k = 1 \sim K$ まで繰り返し行うことにより、当該分割図形についての分割図形ベクトルを定義するようにしたものである。

[0016] (7) 本発明の第7の態様は、上述した第1～第6の態様に係る図形パターンの形状補正装置において、

評価点定義部が、単位図形の輪郭線上において互いに隣接して配置された2つの通常端点の中点位置に評価点を定義するようにしたものである。

[0017] (8) 本発明の第8の態様は、上述した第1～第7の態様に係る図形パターンの形状補正装置において、

補充端点定義部が、1つの輪郭線について定義されている合計 N 個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1) \sim$ 第 N の通常端点 $T(N)$ と呼んだときに、第 n の通常端点 $T(n)$ に着目して、第 $(n-1)$ の通常端点 $T(n-1)$ （但し、 $n=1$ の場合は第 N の通常端点 $T(N)$ ）、第 n の通常端点 $T(n)$ 、第 $(n+1)$ の通常端点 $T(n+1)$ （但し、 $n=N$ の場合は第1の通常端点 $T(1)$ ）の3点が一直線上にある場合に、第 n の通常端点 $T(n)$ を中間通常端点と判定し、この中間通常端点と同じ位置に補充端点を追加定義する処理を、 $n=1 \sim N$ まで繰り返し実行するようにしたものである。

[0018] (9) 本発明の第9の態様は、上述した第1～第8の態様に係る図形パターンの形状補正装置において、

輪郭線分伸縮処理部が、ある1つの隅部通常端点が移動処理により第1の移動端点と第2の移動端点とに分離した場合に、第1の移動端点を一方の端点とする第1の移動後輪郭線分を含む直線と第2の移動端点を一方の端点とする第2の移動後輪郭線分を含む直線との交点を求め、当該交点を新たな隅部通常端点とし、第1の移動後輪郭線分および第2の移動後輪郭線分に対して一方の端点が上記新たな隅部通常端点となるような伸縮処理を行うようにしたものである。

[0019] (10) 本発明の第10の態様は、上述した第1～第9の態様に係る図形パターンの形状補正装置において、

評価点設定ユニットに、通常端点定義部によって単位図形の輪郭線上に定義された通常端点に対して、数もしくは位置を修正する再構築を行う端点再構築部を更に設け、

評価点定義部が、上記再構築によって得られた通常端点に基づいて評価点を定義するようにしたものである。

[0020] (11) 本発明の第11の態様は、上述した第10の態様に係る図形パターンの形状補正装置において、

端点再構築部が、一部もしくは全部の単位図形について、中間通常端点の一部もしくは全部を除去する端点除去処理により、通常端点の再構築を行うようにしたものである。

[0021] (12) 本発明の第12の態様は、上述した第11の態様に係る図形パターンの形状補正装置において、

端点再構築部が、1つの単位図形の1本の輪郭線について定義されている合計N個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1)$ ～第Nの通常端点 $T(N)$ と呼び、第nの通常端点 $T(n)$ に着目して、第nの通常端点 $T(n)$ と、その直前に位置する除去対象となっていない通常端点 $T(n-m)$ （但し、mは、自然数であり、 $n-m < 1$ の場合は、 $T(n-m+N)$ ）との距離を $L1$ とし、第nの通常端点 $T(n)$ と第 $(n+1)$ の通常端点 $T(n+1)$ （但し、 $n=N$ の場合は第1の通常端点 $T(1)$ ：以下同様）との間の距離を $L2$ としたときに、

$L1$ および $L2$ の少なくとも一方が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ （但し、 $n=1$ の場合は第Nの通常端点 $T(N)$ ）、 $T(n)$ 、 $T(n+1)$ の3点が一直線上にある場合に、第nの通常端点 $T(n)$ を除去対象とする判定処理を、 $n=1 \sim N$ まで繰り返し実行し、除去対象となった通常端点を除去する端点除去処理を行うようにしたものである。

[0022] (13) 本発明の第13の態様は、上述した第11の態様に係る図形パターンの形状補正装置において、

端点再構築部が、1つの単位図形の1本の輪郭線について定義されている合計N個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1)$ ～第Nの通常端点 $T(N)$ と呼び、第nの通常端点 $T(n)$ と、その直前に位置する除去対象となっていない通常端点 $T(n-m)$ （但し、mは、自然数であり、 $n-m < 1$ の場合は、 $T(n-m+N)$ ）との距離を L_1 とし、第nの通常端点 $T(n)$ と第 $(n+1)$ の通常端点 $T(n+1)$ （但し、 $n=N$ の場合は第1の通常端点 $T(1)$ ：以下同様）との間の距離を L_2 とし、第 $(n+1)$ の通常端点 $T(n+1)$ と第 $(n+2)$ の通常端点 $T(n+2)$ （但し、 $n=N-1$ の場合は第1の通常端点 $T(1)$ 、 $n=N$ の場合は第2の通常端点 $T(2)$ ：以下同様）との間の距離を L_3 としたときに、

L_2 が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ （但し、 $n=1$ の場合は第Nの通常端点 $T(N)$ ：以下同様）、 $T(n)$ 、 $T(n+1)$ 、 $T(n+2)$ の4点が一直線上にある場合には、 $L_1 < L_3$ であれば通常端点 $T(n)$ を除去対象とし、 $L_1 > L_3$ であれば通常端点 $T(n+1)$ を除去対象とし、 $L_1 = L_3$ であれば通常端点 $T(n)$ もしくは通常端点 $T(n+1)$ のいずれかを除去対象とし、

L_2 が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ 、 $T(n)$ 、 $T(n+1)$ の3点は一直線上にあるが、通常端点 $T(n+2)$ は当該一直線上にない場合には、通常端点 $T(n)$ を除去対象とし、

L_2 が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n)$ 、 $T(n+1)$ 、 $T(n+2)$ の3点は一直線上にあるが、通常端点 $T(n-1)$ は当該一直線上にない場合には、通常端点 $T(n+1)$ を除去対象とする、

という処理を、 $n \geq N$ が満たされるまで、 n を1ずつ増加させながら繰り返し実行し、除去対象となった通常端点を除去する端点除去処理を行うようにしたものである。

[0023] (14) 本発明の第14の態様は、上述した第10の態様に係る図形パターンの形状補正装置において、

端点再構築部が、一部の輪郭線について、中間通常端点を一旦除去し、当該一部の輪郭線の各辺に新たな中間通常端点を配置する端点再配置処理により、通常端点の再構築を行うようにしたものである。

[0024] (15) 本発明の第15の態様は、上述した第14の態様に係る図形パターンの形状補正装置において、

端点再構築部が、特定の輪郭線についての中間通常端点を除去した後、当該特定の輪郭線の各辺上に所定間隔で中間通常端点を再配置することにより、端点再配置処理を行うようにしたものである。

[0025] (16) 本発明の第16の態様は、上述した第10の態様に係る図形パターンの形状補正装置において、

端点再構築部が、上述した第11～第13の態様に係る図形パターンの形状補正装置による端点除去処理と、上述した第14または第15の態様に係る図形パターンの形状補正装置による端点再配置処理と、を単位図形の輪郭線ごとに選択的に実行する機能を有し、

順方向まわりの単位図形ベクトルによって囲まれた正図形上の通常端点に対しては、上記端点除去処理により端点再構築を行い、順方向とは逆の逆方向まわりの単位図形ベクトルによって囲まれた反図形上の通常端点に対しては、上記端点再配置処理により端点再構築を行うようにしたものである。

[0026] (17) 本発明の第17の態様は、上述した第1～第16の態様に係る図形パターンの形状補正装置において、

特徴量抽出ユニットが、パターン補正ユニットによって作成された補正図形パターンについて、当該補正図形パターン上の評価点の周囲の特徴を示す特徴量を抽出する機能を有し、

バイアス推定ユニットが、特徴量に基づいて、評価点の補正図形パターン上の位置と実図形パターン上の位置とのずれ量を示すプロセスバイアスを推定する機能を有し、

パターン補正ユニットが、バイアス推定ユニットから出力されるプロセスバイアスの推定値に基づいて、補正図形パターンに対する更なる補正を行う

ことにより、新たな補正図形パターンを作成する機能を有し、

補正図形パターンに対する補正が繰り返し実行されるようにしたものである。

[0027] (18) 本発明の第18の態様は、上述した第17の態様に係る図形パターンの形状補正装置において、

輪郭線分伸縮処理部が、輪郭線分を伸縮する処理を行った後、伸縮処理後の輪郭線分についての評価点の位置を修正する処理を行うようにしたものである。

[0028] (19) 本発明の第19の態様は、上述した第1～第18の態様に係る図形パターンの形状補正装置または当該図形パターンの形状補正装置に含まれる評価点設定ユニットもしくはパターン補正ユニットを、コンピュータにプログラムを組み込むことにより構成したものである。

[0029] (20) 本発明の第20の態様は、元図形パターンに基づくリソグラフィプロセスによって実基板上に実図形パターンを形成する際に、実図形パターンが元図形パターンに一致するように、元図形パターンの形状を補正して、リソグラフィプロセスで実際に用いる補正図形パターンを作成する図形パターンの形状補正方法において、

コンピュータが、元図形パターンを構成する個々の単位図形について、内部と外部との境界を示す輪郭線上の所定位置に評価点を設定する評価点設定段階と、

コンピュータが、元図形パターンについて、評価点の周囲の特徴を示す特徴量を抽出する特徴量抽出段階と、

コンピュータが、特徴量に基づいて、評価点の元図形パターン上の位置と実図形パターン上の位置とのずれ量を示すプロセスバイアスを推定するバイアス推定段階と、

コンピュータが、バイアス推定段階によって得られるプロセスバイアスの推定値に基づいて、元図形パターンに対する補正を行うことにより、補正図形パターンを作成するパターン補正段階と、

を有し、

評価点設定段階が、

多角形からなる個々の単位図形を入力する単位図形入力ステップと、

単位図形を直線状の分割線で分割して複数の多角形からなる分割図形を形成する単位図形分割ステップと、

各分割図形について、当該分割図形を構成する多角形の輪郭線に沿って所定の順方向まわりに一周するように、多角形の個々の辺上にそれぞれ分割図形ベクトルを定義する分割図形ベクトル定義ステップと、

分割図形ベクトルの集合から、互いに隣接する一対の隣接多角形の重複する辺上に配置され、互いに向きが逆方向である重複ベクトル対を探索し、探索された重複ベクトル対のうちの重複区間部分を除去し、残ったベクトルにより、単位図形の輪郭線に沿って配置された複数の単位図形ベクトルを定義する単位図形ベクトル定義ステップと、

個々の単位図形ベクトルの始点位置に、それぞれ通常端点を定義する通常端点定義ステップと、

単位図形の輪郭線上において互いに隣接して配置された2つの通常端点の間に、それぞれ評価点を定義する評価点定義ステップと、

個々の単位図形について、多角形の頂点を構成しない通常端点である中間通常端点を探索し、探索された中間通常端点と同じ位置に補充端点を追加定義する補充端点定義ステップと、

を有し、

パターン補正段階が、

個々の評価点について、単位図形の輪郭線に沿って当該評価点の一方の側に隣接する通常端点もしくは補充端点を第1の端点とし、他方の側に隣接する通常端点もしくは補充端点を第2の端点とする輪郭線分を定義し、個々の評価点についての輪郭線分を、当該評価点のプロセスバイアスの推定値に応じた補正量だけ、輪郭線分に直交する方向に移動させる移動処理を行う輪郭線分移動ステップと、

移動処理前に互いに同じ位置にあった中間通常端点および補充端点が、移動処理後に異なる2つの位置に分離した場合、それぞれの移動処理後の位置を連結する新たな輪郭線分を追加する追加処理を行う輪郭線分追加ステップと、

移動処理前に多角形の頂点を構成していた通常端点である隅部通常端点が、移動処理後に異なる2つの位置に分離した場合、分離後の2つの隅部通常端点が融合して新たな隅部通常端点となるように、分離後の隅部通常端点を含む各輪郭線分をそれぞれ伸縮する伸縮処理を行う輪郭線分伸縮ステップと、

を有し、移動処理、追加処理、伸縮処理を行った後の輪郭線分の集合体を輪郭線とする単位図形によって構成される補正図形パターンを作成するようにしたものである。

[0030] (21) 本発明の第21の態様は、上述した第20の態様に係る図形パターンの形状補正方法において、

評価点設定段階が、通常端点定義ステップによって定義された通常端点に対して、数もしくは位置を修正する再構築を行う端点再構築ステップを更に有し、

評価点定義ステップにおいて、再構築によって得られた通常端点に基づいて評価点を定義するようにしたものである。

発明の効果

[0031] 本発明に係る図形パターンの形状補正装置および形状補正方法によれば、元図形パターンを構成する単位図形が多角形に分割された上で、この多角形の頂点位置を基準として端点が定義され、隣接して配置された2つの端点間を結ぶ輪郭線分ごとに評価点が定義され、輪郭線分ごとに位置の補正を行うことができる。このため、元図形パターンに対する補正を行う際に、実用上十分なシミュレーション精度を維持しつつ、できるだけ演算負担を軽減できる効率的な評価点設定を行うことが可能になる。また、多角形の頂点位置を基準として定義した端点に対して、更に端点再構築を行う実施形態を採用す

れば、端点の数や位置が修正されるため、より効率的な評価点設定を行うことが可能になる。

図面の簡単な説明

[0032] [図1]先願発明および本発明の基本的実施形態に係る図形パターンの形状補正装置 100 の構成を示すブロック図である。

[図2]元図形パターンと実図形パターンとの間に形状の相違が生じた一例を示す平面図である。

[図3]図2に示す例において、各評価点の設定例および各評価点に生じるプロセスバイアスを示す平面図である。

[図4]図1に示す図形パターンの形状補正装置 100 を用いた製品の設計・製造工程を示す流れ図である。

[図5]図形パターンの輪郭線上に定義された評価点について、その周囲の特徴を把握する概念を示す平面図である。

[図6]図1に示す特徴量抽出ユニット 120 およびバイアス推定ユニット 130 において実行される処理の概要を示す図である。

[図7]図1に示す特徴量抽出ユニット 120 において実行される処理手順を示す流れ図である。

[図8]図1に示す図形パターンの形状補正装置 100 に与えられる元図形パターン 10 の具体例を示す平面図である。

[図9]図1に示す元画像作成部 121 において、画素の二次元配列からなるメッシュ上に元図形パターン 10 を重ね合わせる処理が行われた状態を示す平面図である。

[図10]図9に示す元図形パターン 10 に基づいて作成された面積密度マップ M1 を示す図である。

[図11]図9に示す元図形パターン 10 に基づいて作成されたエッジ長密度マップ M2 を示す図である。

[図12]ドーズ量の情報を含んだ元図形パターン 10 を示す平面図である。

[図13]図12に示すドーズ量付き元図形パターン 10 に基づいて作成された

ドーズ密度マップ M_3 を示す図である。

[図14]第 k 番目の準備画像 Q_k にガウシアンフィルタ G_{F33} を用いたフィルタ処理を施すことにより、第 k 番目の階層画像 P_k を作成する手順を示す平面図である。

[図15]図14に示すフィルタ処理により得られた第 k 番目の階層画像 P_k を示す平面図である。

[図16]図14に示すフィルタ処理に利用する画像処理フィルタの例を示す平面図である。

[図17]図14に示すフィルタ処理に利用する画像処理フィルタの別な例を示す平面図である。

[図18]第 k 番目の階層画像 P_k に対して、アベレージ・プーリング処理を施すことにより、第 $(k+1)$ 番目の準備画像 $Q(k+1)$ を作成する手順を示す平面図である。

[図19]第 k 番目の階層画像 P_k に対して、マックス・プーリング処理を施すことにより、第 $(k+1)$ 番目の準備画像 $Q(k+1)$ を作成する手順を示す平面図である。

[図20]図1に示す画像ピラミッド作成部122において、 n 通りの階層画像 $P_1 \sim P_n$ からなる画像ピラミッド P_P を作成する手順を示す平面図である。

[図21]図1に示す特徴量算出部123において、各階層画像から評価点 E についての特徴量を算出する手順を示す平面図である。

[図22]図21に示す特徴量算出手順で用いる具体的な演算方法を示す図である。

[図23]図1に示す画像ピラミッド作成部122において、 n 通りの差分画像 $D_1 \sim D_n$ からなる画像ピラミッド P_D を作成する手順を示す平面図である。

[図24]図1に示す推定演算部132として、ニューラルネットワークを利用した実施例を示すブロック図である。

[図25]図24に示すニューラルネットワークで実行される具体的な演算プロセスを示すダイアグラムである。

[図26]図25に示すダイアグラムにおける第1隠れ層の各値を求める演算式を示す図である。

[図27]図26に示す活性化関数 $f(\xi)$ の具体例を示す図である。

[図28]図25に示すダイアグラムにおける第2隠れ層～第N隠れ層の各値を求める演算式を示す図である。

[図29]図25に示すダイアグラムにおける出力層の値 y を求める演算式を示す図である。

[図30]図24に示すニューラルネットワークが用いる学習情報 L を得るための学習段階の手順を示す流れ図である。

[図31]図30に示す流れ図におけるステップS84の推定演算部学習の詳細な手順を示す流れ図である。

[図32]本発明の基本的実施形態に係る図形パターンの形状補正装置100（図1参照）内の評価点設定ユニット110の構成を示すブロック図である。

[図33]図32に示す単位図形分割部112によって行われた具体的な分割処理の例を示す平面図である。

[図34]図32に示す単位図形分割部112によって、平行四辺形型の単位図形10に対して行われた具体的な分割処理の例を示す平面図である。

[図35]図32に示す単位図形分割部112による分割処理で得られる台形状の分割図形の例を示す平面図である。

[図36]図34(c)に示す各分割図形に対して形状補正が行われた状態を示す平面図である。

[図37]図36に示す補正後の単位図形15に基づく露光工程の一例を示す平面図である。

[図38]図34(c)に示す平行四辺形型の単位図形10に対して、図32に示す分割図形ベクトル定義部113、単位図形ベクトル定義部114、通常端点定義部115による処理が行われた例を示す平面図である。

[図39] L字型の単位図形10に対して、図32に示す単位図形分割部112、分割図形ベクトル定義部113、単位図形ベクトル定義部114、通常端点定義部115による処理が行われた例を示す平面図である。

[図40] 口の字型の単位図形10に対して、図32に示す単位図形分割部112、分割図形ベクトル定義部113、単位図形ベクトル定義部114、通常端点定義部115による処理が行われた例を示す平面図である。

[図41] 図32に示す端点再構築部116による端点再構築処理として、3点式の端点除去処理を採用した場合の手順を示す模式図(図(a))および流れ図(図(b))である。

[図42] 図38(d)に示す平行四辺形型の単位図形10上の通常端点A~Jに対して、図41に示す3点式の端点除去処理を適用して端点の再構築を行った例を示す平面図である。

[図43] 図39(f)に示すL字型の単位図形10上の通常端点A~Iに対して、図41に示す3点式の端点除去処理を適用して端点の再構築を行った例を示す平面図である。

[図44] 図32に示す端点再構築部116による端点再構築処理として、4点式の端点除去処理を採用した場合の手順を示す模式図(図(a))および流れ図(図(b))である。

[図45] 図32に示す端点再構築部116による端点再構築処理として、端点再配置処理を採用した場合の手順を示す模式図である。

[図46] 図32に示す端点再構築部116による端点再構築処理として、端点再配置処理を採用した場合の手順を示す流れ図である。

[図47] 図40(f)に示す口の字型の単位図形10上の通常端点M~Tに対して、図46に示す端点再配置処理を適用して端点の再構築を行った例を示す平面図である。

[図48] 図32に示す評価点定義部117によって、図42(b)に示す端点再構築後の平行四辺形型の単位図形10上に評価点1~8を定義した例を示す平面図である。

[図49]図32に示す評価点定義部117によって、図43(b)に示す端点再構築後のL字型の単位図形10上に評価点1～8を定義した例を示す平面図である。

[図50]図32に示す評価点定義部117によって、図47(b)に示す端点再構築後の口の字型の単位図形10上に評価点1～18を定義した例を示す平面図である。

[図51]図32に示す補充端点定義部118によって、図48(b)，図49(b)，図50(b)に示す各単位図形上に補充端点を定義した例を示す平面図である。

[図52]図32に示す補充端点定義部118によって行われる補充端点定義処理の具体的な手順を示す模式図(図(a)，(b))および流れ図(図(c))である。

[図53]本発明の基本的実施形態に係る図形パターンの形状補正装置100(図1参照)内のパターン補正ユニット140およびこれに関連する構成要素を示すブロック図である。

[図54]図51(b)に示すL字型の単位図形上に定義された各端点および各評価点を示す拡大平面図である。

[図55]図54に示す各端点および各評価点の相互関係を表すデータフォーマットの一例を示す図である。

[図56]図53に示す輪郭線分移動処理部141によって、図54に示すL字型の単位図形10に対して輪郭線分の移動処理が行われた状態を示す平面図である。

[図57]図53に示す輪郭線分追加処理部142によって、図56に示す移動処理後のL字型の単位図形に対して輪郭線分の追加処理が行われた状態を示す平面図である。

[図58]図53に示す輪郭線分伸縮処理部143によって、図57に示す追加処理後のL字型の単位図形に対して輪郭線分の伸縮処理が行われた状態を示す平面図である。

[図59]図53に示す輪郭線分伸縮処理部143によって、図58に示す伸縮処理後のL字型の単位図形の輪郭線分について、評価点の位置を修正する処理を行った状態を示す平面図である。

[図60]単位図形の個々の辺ごとに所定間隔で端点を定義するという従来手法の手順を示す平面図である。

[図61]本発明において、単位図形を多角形に分割し、この多角形の頂点位置を基準として端点を定義するメリットを示す平面図である。

[図62]補充端点を定義するメリットを示す平面図である。

[図63]本発明において、補充端点を輪郭線に直交する方向に移動させるメリットを示す平面図である。

[図64]中間通常端点および補充端点を任意方向に移動させた場合のデメリットを示す平面図である。

[図65]図58に示す例において、端点再構築を行わなかった場合のデメリットを示す平面図である。

発明を実施するための形態

[0033] 以下、本発明を図示する実施形態に基づいて説明する。なお、本発明に係る技術は、特許協力条約に基づく国際出願第PCT/JP2018/022100号（以下、先願と呼ぶ）に記載された発明（以下、先願発明と呼ぶ）に適用するために開発されたものである。そこで、ここでは説明の便宜上、以下の§1～§4において、まず、先願発明の説明を行い、§5以降において、本発明の特徴を説明することにする。したがって、以下の§1～§4で述べる内容（図1～図31に示す内容）は、実質的に先願に記載された基本的実施形態と同じものである。

[0034] <<< §1. 先願発明に係る形状補正装置の基本構成 >>>

ここでは、先願発明の基本的実施形態に係る図形パターンの形状補正装置100の構成を、図1のブロック図を参照しながら説明する。図示のとおり、この図形パターンの形状補正装置100は、評価点設定ユニット110、特徴量抽出ユニット120、バイアス推定ユニット130、パターン補正ユ

ニット140を有している。ここで、評価点設定ユニット110、特徴量抽出ユニット120、バイアス推定ユニット130の3つのユニットによって、先願発明に係る図形パターンの形状推定装置100'が構成されており、先願発明に係る図形パターンの形状補正装置100は、この図形パターンの形状推定装置100'に、更に第4番目のユニットとしてパターン補正ユニット140を付加することにより構成される。

[0035] <1. 1 図形パターンの形状推定装置>

はじめに、図形パターンの形状推定装置100'の構成および機能について説明する。この図形パターンの形状推定装置100'は、元図形パターン10を用いたリソグラフィプロセスをシミュレートすることにより、実基板S上に形成される実図形パターン20の形状を推定する役割を果たす。図1の上方には、元図形パターン10から右方向に向かう一点鎖線の矢印が示され、この矢印の先には、実図形パターン20を有する実基板Sが描かれている。この一点鎖線の矢印は、物理的なリソグラフィプロセスを示している。

[0036] 図示の元図形パターン10は、コンピュータを用いた設計作業によって作成された図形パターンのデータであり、一点鎖線の矢印は、このデータに基づいて、物理的な露光、現像、エッチング等のリソグラフィプロセスを実施することにより、実基板Sが作製されることを示している。この実基板S上には、元図形パターン10に応じた実図形パターン20が形成されることになる。ただ、このようなリソグラフィプロセスを実施すると、元図形パターン10と実図形パターン20との間には、若干の食い違いが生じる。これは、前述したとおり、リソグラフィプロセスに含まれる、露光、現像、エッチングという工程の諸条件により、元図形パターン10どおりの正確な図形を実基板S上に形成することが困難なためである。

[0037] 図2は、元図形パターン10と実図形パターン20との間に形状の相違が生じた具体例を示す平面図である。半導体デバイスなどでは、実際には、非常に微細かつ複雑な図形パターンをシリコン等の実基板Sの表面に形成する必要があるが、ここでは説明の便宜上、図2(a)に示すような単純な図形が

元図形パターン１０として与えられた場合について説明する。図示の元図形パターン１０は、１つの長方形からなるパターンであり、たとえば、実基板Ｓ上に、ハッチングを施した長方形内部領域に相当する材料層を形成することを示す元の図形データということになる。

[0038] 実際のリソグラフィ工程では、実基板Ｓ上の材料層の上面にレジスト層を形成し、光や電子線による露光を行って、このレジスト層に対する描画を行うことになる。たとえば、レジスト層に対して、図２(a)に示す元図形パターン１０の内部領域（ハッチングを施した部分）について露光を行った後、レジスト層を現像して非露光部を除去すれば、露光部が残存レジスト層（ハッチングを施した部分）として残ることになる。更に、この残存レジスト層をマスクとして、材料層に対してエッチングを施せば、理論的には、材料層についても、元図形パターン１０の内部領域（ハッチングを施した部分）を残すことができ、実基板Ｓ上に、図２(a)に示す元図形パターン１０と同一の実図形パターンを得ることができる。

[0039] しかしながら、実際には、実基板Ｓ上に得られる実図形パターン２０は、元図形パターン１０に正確には一致しない。その原因は、リソグラフィプロセスに含まれる、露光、現像、エッチングという工程の諸条件が、最終的に得られる実図形パターン２０の形状に影響を及ぼすためである。たとえば、露光工程では、光や電子線によってレジスト層に対する描画を行うことになるが、その際に、近接効果（ＰＥ：Proximity Effect）によって、レジスト層に実際に描画される露光領域は、元図形パターン１０よりも若干広い領域になることが知られている。このような近接効果の影響を受けると、実基板Ｓ上に得られる実図形パターン２０は、図２(b)に示す例のように、元図形パターン１０（破線で示す）よりも広がった領域になる。

[0040] この他、現像工程で用いる現像液の特性、エッチング工程で用いるエッチング液やプラズマの特性、現像工程やエッチング工程の時間などの条件も、実図形パターン２０の形状に影響を及ぼす要因になる。したがって、実際に半導体デバイスなどを製造する際には、コンピュータ上で所望の元図形パタ

ーン10を設計した後、この元図形パターン10を用いたリソグラフィプロセスをコンピュータ上でシミュレートし、実基板S上に形成されるであろう実図形パターン20の形状を推定する手順が実行される。図1に示す図形パターンの形状推定装置100'は、このような推定を行う機能をもった装置であり、実基板Sを作成するリソグラフィプロセス（露光、現像、エッチング工程）を実際に行うことなしに、シミュレーションによって、実基板S上に形成されるであろう実図形パターン20の形状を推定する機能を有している。

[0041] 図1に示す図形パターンの形状推定装置100'では、評価点設定ユニット110によって、元図形パターン10上に評価点Eが設定される。具体的には、形状推定装置100'には、元図形パターン10として、図形の内部と外部との境界を示す輪郭線の情報を含む図形データが与えられ、評価点設定ユニット110は、そのような元図形パターン10に基づいて、輪郭線上の所定位置に評価点を設定する処理を行う。

[0042] 図3は、図2(a)に示す元図形パターン10について、いくつかの評価点を設定した例および各評価点に生じるプロセスバイアス（寸法誤差）を示す平面図である。まず図3(a)は、図2(a)に示す元図形パターン10（長方形の図形）の輪郭線上に、評価点E11、E12、E13を設定した例を示す平面図である。図3では、説明の便宜上、3つの評価点E11、E12、E13が設定された簡素な例が示されているが、実際には、長方形の各辺上に、より多数の評価点が設定される。たとえば、輪郭線に沿って所定ピッチで連続的に評価点を設定するように定めておけば、自動的に多数の評価点を設定することができる。なお、本発明は、この評価点設定に関する新たな手法を提案するものであり、その具体的内容は、§5以降で説明する。

[0043] ここでは、図2(a)に示す元図形パターン10に基づいて、図2(b)に示すような実図形パターン20が得られた場合を考えてみよう。図3(b)は、図2(b)に示す実図形パターン20の輪郭（実線）を、図2(a)に示す元図形パターン10の輪郭（破線）と対比して示した平面図であり、実線で示す

実図形パターン 20 の輪郭線が、破線で示す元図形パターン 10 の輪郭線に比べて、寸法 y だけ外側に広がっている状態が示されている。このため、元図形パターン 10 の横幅 a は、実図形パターン 20 では横幅 b に広がっている。縦幅についても同様に若干の広がりが生じている。

[0044] 図 3 (b) において、実図形パターン 20 上の各評価点 E_{21} , E_{22} , E_{23} は、元図形パターン 10 上の各評価点 E_{11} , E_{12} , E_{13} に対応する点として定められた評価点である。ここで、評価点 E_{21} は、評価点 E_{11} を破線で示す輪郭線の法線方向外側に向かって所定寸法 y_{11} だけずらした点として定義されている。同様に、評価点 E_{22} は、評価点 E_{12} を破線で示す輪郭線の法線方向外側に向かって所定寸法 y_{12} だけずらした点として定義され、評価点 E_{23} は、評価点 E_{13} を破線で示す輪郭線の法線方向外側に向かって所定寸法 y_{13} だけずらした点として定義されている。

[0045] 図 3 に示す例のように、ここでは、元図形パターン 10 に対する実図形パターン 20 の形状変化を定量的に示すために、各評価点 E について生じる輪郭線の法線方向についてのずれ量 y を用いることにする。また、このずれ量 y を、リソグラフィプロセスに起因して生じるバイアス量であることから「プロセスバイアス y 」と呼ぶことにする。プロセスバイアス y は正負の符号をもった値であり、以下に示す実施例では、図形の露光部（描画部）が太る方向を正の値、細る方向を負の値と定義することにする。図示の例では、輪郭線で囲まれた図形の内部が露光部（描画部）になるので、輪郭線の外側方向にずれた場合には正の値、輪郭線の内側方向にずれた場合には負の値となる。図 3 に示す例の場合、各評価点 E はいずれも外側方向にずれを生じているため、プロセスバイアス y_{11} , y_{12} , y_{13} は正の値をとる。

[0046] なお、図 3 には、説明の便宜上、3 つの評価点 E_{11} , E_{12} , E_{13} しか示されていないが、実際には、元図形パターン 10 の輪郭線上には、より多数の評価点 E が定義される。したがって、各評価点 E のそれぞれについてプロセスバイアス y を推定することができれば、各評価点 E を輪郭線の法線方向にプロセスバイアス y だけずらすことにより、リソグラフィプロセス後

の各評価点Eの位置を推定することができ、実図形パターン20の輪郭線位置を推定することができる。

[0047] もっとも、各評価点Eのプロセスバイアス y の値は、個々の評価点ごとに異なる。たとえば、図3(b)に示す例の場合、プロセスバイアス y_{11} 、 y_{12} 、 y_{13} の値はそれぞれ個別の値になる。これは、各評価点E₁₁、E₁₂、E₁₃の元図形パターン10に対する相対位置がそれぞれ異なっているため、リソグラフィプロセスによる影響も異なり、それぞれに生じるずれ量にも差が生じるためである。したがって、元図形パターン10に基づいて実図形パターン20の形状をシミュレーションによって推定する際の推定精度を高めるためには、個々の評価点ごとにリソグラフィプロセスによる影響を適切に予測し、適切なプロセスバイアス y を求めることが重要である。

[0048] そこで、図1に示す図形パターンの形状推定装置100'では、まず、評価点設定ユニット110により、元図形パターン10上に評価点が設定される。具体的には、元図形パターン10に含まれている図形の内部と外部との境界を示す輪郭線の情報に基づいて、この輪郭線上の所定位置に各評価点Eを設定すればよい。たとえば、輪郭線に沿って所定間隔で連続的に評価点を設定することができる（本発明に固有の評価点設定方法については、§5以降で説明する。）。

[0049] 続いて、特徴量抽出ユニット120により、元図形パターン10について、各評価点Eの周囲の特徴を示す特徴量を抽出する。ある1つの評価点Eについての特徴量 x は、当該評価点Eの周囲の特徴を示す値ということになる。このような特徴量 x を抽出する処理を行うため、特徴量抽出ユニット120は、図1に示すとおり、元画像作成部121、画像ピラミッド作成部122、特徴量算出部123を有している。

[0050] 元画像作成部121は、与えられた元図形パターン10に基づいて、それぞれ所定の画素値を有する画素の集合体からなる元画像を作成する。たとえば、図2(a)に示すような元図形パターン10が与えられた場合、長方形の内部の画素（図のハッチング部分の画素）については画素値1、長方形の外

部の画素については画素値 0 を付与すれば、二値画像からなる元画像が作成される。

[0051] 画像ピラミッド作成部 1 2 2 は、この元画像を縮小して縮小画像を作成する縮小処理を含む画像ピラミッド作成処理を行い、複数 n 枚の階層画像からなる画像ピラミッドを作成する。画像ピラミッドの各階層を構成する n 枚の階層画像は、いずれも元画像作成部 1 2 1 で作成された元画像に対して、所定の画像処理を施すことにより得られる画像であり、それぞれ異なるサイズをもっている。このような複数の階層画像の集合体を「画像ピラミッド」と呼ぶのは、各階層画像をサイズの大きい方から小さい方へと順に積み上げて階層構造を形成すると、あたかもピラミッドが構成されるように見えるためである。

[0052] 特徴量算出部 1 2 3 は、この画像ピラミッドの階層を構成する n 枚の各階層画像について、それぞれ評価点 E の近傍の画素の画素値に基づいて特徴量を算出する。具体的には、1 枚目の階層画像における評価点 E の近傍の画素の画素値に基づいて特徴量 x_1 が算出され、2 枚目の階層画像における評価点 E の近傍の画素の画素値に基づいて特徴量 x_2 が算出され、同様に、 n 枚目の階層画像における評価点 E の近傍の画素の画素値に基づいて特徴量 x_n が算出されることになり、1 つの評価点 E について、 n 個の特徴量 $x_1 \sim x_n$ が抽出される。たとえば、図 3 (a) に示す例の場合、評価点 E_{11} について、 n 通りの特徴量 $x_1 (E_{11}) \sim x_n (E_{11})$ が抽出され、評価点 E_{12} について、 n 通りの特徴量 $x_1 (E_{12}) \sim x_n (E_{12})$ が抽出され、評価点 E_{13} について、 n 通りの特徴量 $x_1 (E_{13}) \sim x_n (E_{13})$ が抽出される。

[0053] 一方、バイアス推定ユニット 1 3 0 は、特徴量抽出ユニット 1 2 0 によって抽出された特徴量 x に基づいて、評価点 E の元図形パターン 1 0 上の位置と実図形パターン 2 0 上の位置とのずれ量を示すプロセスバイアス y を推定する処理を行う。このような推定処理を行うため、バイアス推定ユニット 1 3 0 は、特徴量入力部 1 3 1 と推定演算部 1 3 2 を有している。特徴量入力

部 1 3 1 は、特徴量算出部 1 2 3 によって評価点 E について算出された特徴量 $x_1 \sim x_n$ を入力する構成要素であり、推定演算部 1 3 2 は、予め実施された学習段階によって得られた学習情報 L に基づいて、特徴量 $x_1 \sim x_n$ に応じた推定値を求め、求めた推定値を評価点 E についてのプロセスバイアスの推定値 y として出力する処理を行う。

[0054] より具体的に言えば、推定演算部 1 3 2 は、元図形パターン 1 0 を構成する図形の輪郭線上に位置する各評価点 E について、当該輪郭線の法線方向についてのずれ量としてプロセスバイアスの推定値 y を出力することになる。たとえば、図 3 (a) に示す元図形パターン 1 0 については、図 3 (b) に示すように、評価点 E 1 1 についてプロセスバイアス $y_1 1$ 、評価点 E 1 2 についてプロセスバイアス $y_1 2$ 、評価点 E 1 3 についてプロセスバイアス $y_1 3$ が、それぞれ推定値として推定演算部 1 3 2 から出力される。こうして、各評価点 E についてのプロセスバイアスの推定値 y が得られれば、各評価点 E の新たな位置（輪郭線の法線方向に、プロセスバイアス y だけずらした位置）を決定することができるので、図 3 (b) に示すように、実図形パターン 2 0 の形状を推定することができる。

[0055] 以上が、先願発明に係る図形パターンの形状推定装置 1 0 0' の基本構成および基本動作である。なお、特徴量抽出ユニット 1 2 0 の具体的な動作は § 2 で詳述し、バイアス推定ユニット 1 3 0 の具体的な動作は § 3 で詳述する。

[0056] < 1. 2 図形パターンの形状補正装置 >

続いて、先願発明に係る図形パターンの形状補正装置 1 0 0 の構成および機能について説明する。図形パターンの形状補正装置 1 0 0 は、上述した図形パターンの形状推定装置 1 0 0' を用いて、元図形パターン 1 0 の形状を補正する装置であり、図 1 に示すとおり、図形パターンの形状推定装置 1 0 0' の構成要素となる評価点設定ユニット 1 1 0、特徴量抽出ユニット 1 2 0、バイアス推定ユニット 1 3 0 に加えて、更に、パターン補正ユニット 1 4 0 を備えている。パターン補正ユニット 1 4 0 は、バイアス推定ユニット

130から出力されるプロセスバイアスの推定値 y に基づいて、元図形パターン10に対する補正を行う構成要素であり、このパターン補正ユニット140による補正によって得られる補正図形パターン15が、この図形パターンの形状補正装置100の最終的な出力となる。

[0057] パターン補正ユニット140による補正は、元図形パターン10上の各評価点Eをプロセスバイアス y を相殺する方向に移動させ、各図形の境界線を移動後の各評価点Eの位置に修正することによって行うことができる。たとえば、図3(a)に示す元図形パターン10について、図3(b)に示すような実図形パターン20が推定された場合を考えてみよう。この場合、元図形パターン10に含まれる長方形の横幅 a は、実図形パターン20上では横幅 b に増加しており、 $(b - a) / 2 = y$ とすれば、長方形の左右の辺の位置をいずれも内側へ y だけ移動させれば、元図形パターン10と同じ横幅 a を有する長方形が得られることになる。したがって、基本的には、元図形パターン10の横幅 a を $2y$ だけ減じる補正を行い、この補正後の図形パターンに基づいてリソグラフィプロセスを実行すれば、実基板S上には、実図形パターン20として、当初の設計どおりの横幅 a を有する長方形を得ることができる。

[0058] 図3(a)に示す元図形パターン10の場合、評価点E11を左方（長方形の内側）にプロセスバイアス $y11$ だけ移動させ、評価点E12を左方（長方形の内側）にプロセスバイアス $y12$ だけ移動させ、評価点E13を上方（長方形の内側）にプロセスバイアス $y13$ だけ移動させる補正を行えばよい。もちろん、実際には、より多数の評価点が定義されているので、これらすべての評価点について、プロセスバイアスに相当する寸法だけ長方形の内側に移動させる補正を行い、補正後の評価点を結ぶ新たな輪郭線を定義すれば、当該輪郭線で画定される図形を含む補正図形パターン15が得られることになる。このような補正処理自体は公知の技術なので、ここでは詳しい説明は省略する。

[0059] もっとも、実際には、こうして得られた補正図形パターン15を用いてリ

ソグラフィプロセスを実行し、実基板S上に実図形パターン25（図示省略）を形成しても、得られる実図形パターン25は、設計当初の元図形パターン10には正確には一致しない（たとえば、実基板S上に形成される長方形の横幅は、正確にはaにならない）。なぜなら、元図形パターン10に含まれる図形と補正図形パターン15に含まれる図形とでは、サイズや形状が異なるため、リソグラフィプロセスを実行した場合の近接効果などの影響に相違が生じるためである。

[0060] 別言すれば、特定の評価点Eについてシミュレーションを行った結果、プロセスバイアス y が生じることが判明したとしても、単に当該評価点Eの位置をプロセスバイアス y だけ逆方向に移動させただけでは、正確な補正を行うことはできないのである。

[0061] もちろん、元図形パターン10を用いてリソグラフィプロセスを実行した結果として得られる実図形パターン20に比べれば、補正図形パターン15を用いてリソグラフィプロセスを実行した結果として得られる実図形パターン25の方が、より元図形パターン10に近いパターンになるので、元図形パターン10をそのまま用いて実際のリソグラフィプロセスを実行するよりは、パターン補正ユニット140による補正によって得られる補正図形パターン15を用いて実際のリソグラフィプロセスを実行した方が、実基板S上には、より正確な図形パターンが得られることになる。すなわち、パターン補正ユニット140によって補正を行えば、誤差が縮小することは確かである。

[0062] そこで、実用上は、図1に示すとおり、パターン補正ユニット140から出力された補正図形パターン15を、再び、図形パターンの形状補正装置100に与える処理を繰り返し行うようにする。すなわち、補正図形パターン15は、図形パターンの形状推定装置100'に新たな元図形パターンとして与えられることになり、この新たな元図形パターン（補正図形パターン15）について、§1.1で述べた処理が実行される。具体的には、評価点設定ユニット110によって補正図形パターン15上に各評価点Eの設定が行

われ、特徴量抽出ユニット 120 によって各評価点 E についての特徴量が抽出され、バイアス推定ユニット 130 によって各評価点 E についてのプロセスバイアス推定値 y が算出される。そして、この算出されたプロセスバイアス推定値 y を用いて、パターン補正ユニット 140 において再度の補正処理が行われる。

[0063] 図 1 に示す図形パターンの形状補正装置 100 は、このように、図形パターンに対する補正を繰り返し実行する機能を有している。すなわち、元図形パターン 10 に基づいて第 1 の補正図形パターン 15 が得られ、この第 1 の補正図形パターン 15 に基づいて第 2 の補正図形パターンが得られ、この第 2 の補正図形パターンに基づいて第 3 の補正図形パターンが得られ、... という補正処理が繰り返されることになる。補正処理を行うたびに、元図形パターンとシミュレーションにより得られる図形パターンとの形状誤差は縮小してゆく。

[0064] したがって、たとえば、元図形パターンとシミュレーションにより得られる図形パターンとの形状誤差が所定の許容範囲内に収束した時点で補正完了とし、最後に得られた補正図形パターンを用いて実際のリソグラフィプロセスを実行すれば、実基板 S 上には、設計当初の元図形パターン 10 に近い実図形パターンを形成することができる。かくして、先願発明に係る図形パターンの形状補正装置 100 を用いれば、元図形パターンの形状を的確に補正することが可能になる。

[0065] なお、図 1 に示す評価点設定ユニット 110、特徴量抽出ユニット 120、バイアス推定ユニット 130、パターン補正ユニット 140 は、いずれもコンピュータに所定のプログラムを組み込むことによって構成されている。したがって、先願発明に係る図形パターンの形状推定装置 100' や図形パターンの形状補正装置 100 は、実際には、汎用のコンピュータに専用のプログラムを組み込むことによって実現される。

[0066] 図 4 は、図 1 に示す図形パターンの形状補正装置 100 を用いた製品の設計・製造工程を示す流れ図である。まず、ステップ S1 において、製品設計

段階が行われる。この製品設計段階は、半導体デバイスなどを構成するための図形パターンを作成する工程であり、図 1 に示す元図形パターン 10 は、この製品設計段階で作成されることになる。なお、半導体デバイスなどの製品設計を行い、図形パターンを作成する装置自体は既に公知の装置であるので、ここでは詳しい説明は省略する。

[0067] 次のステップ S 2 の評価点設定段階は、図 1 に示す評価点設定ユニット 110 において実行される段階である。たとえば、図 2 (a) に示すような元図形パターン 10 が与えられた場合、図 3 (a) に示すような評価点 E 11, E 12, E 13 などが設定される。続くステップ S 3 の特徴量抽出段階は、図 1 に示す特徴量抽出ユニット 120 において実行される段階であり、前述したとおり、1 つの評価点 E について、 n 通りの特徴量 $x_1 \sim x_n$ が抽出される（詳細な抽出手順は、§ 2 で述べる）。そして、ステップ S 4 のプロセスバイアス推定段階は、図 1 に示すバイアス推定ユニット 130 において実行される段階であり、前述したとおり、1 つの評価点 E について、 n 通りの特徴量 $x_1 \sim x_n$ を用いてプロセスバイアス推定値 y が求められる（詳細な算出手順は、§ 3 で述べる）。

[0068] そして、ステップ S 5 のパターン形状補正段階は、図 1 に示すパターン補正ユニット 140 において実行される段階であり、前述したとおり、各評価点 E について求められたプロセスバイアス推定値 y を用いて、元図形パターン 10 に対する補正を行うことにより補正図形パターン 15 が得られる。このような補正は、1 回だけでは不十分であるため、ステップ S 6 において「補正完了」と判断されるまで、ステップ S 2 へ戻る処理が繰り返される。すなわち、ステップ S 5 で得られた補正図形パターン 15 を、新たな元図形パターン 10 として取り扱うことにより、ステップ S 2 ～ S 5 の処理が繰り返し実行されることになる。

[0069] このような繰り返し処理の結果、ステップ S 6 において「補正完了」と判断されるに至った場合には、ステップ S 7 へと進み、リソグラフィプロセスが実行される。「補正完了」との判断は、たとえば「一定割合の評価点 E に

ついて、元図形パターン上の位置とシミュレーションにより得られる図形パターン上の位置との誤差が、所定の基準値以下になる」というような特定の条件が満たされたことにより行うことができる。ステップS 7のリソグラフィプロセスでは、最終的に得られた補正図形パターンに基づいて、露光、現像、エッチングという実工程が行われ、実基板Sの製造が行われる。

[0070] 図4に示す流れ図において、ステップS 1～S 6の段階は、計算機（コンピュータ）上で実行するプロセスであり、ステップS 7の段階は、実基板S上で実行するプロセスである。

[0071] <1. 3 先願発明における特徴量抽出の基本概念>

これまで、図1に示す図形パターンの形状推定装置100'および図形パターンの形状補正装置100の基本構成とその基本動作を述べた。これらの各装置において、先願発明として最も特徴的な構成要素は、特徴量抽出ユニット120である。先願発明は、元図形パターン10からの的確な特徴量を抽出して的確なシミュレーションを行い、実基板S上に形成される実図形パターン20の形状を正確に推定するという作用効果を奏するものであるが、このような作用効果を得るために最も重要な役割を果たす構成要素が特徴量抽出ユニット120である。別言すれば、先願発明の重要な特徴は、元図形パターン10から非常にユニークな方法で特徴量の抽出を行う点にある。そこで、ここでは、先願発明における特徴量抽出の基本概念を説明する。

[0072] 図5は、長方形からなる図形パターン10の輪郭線上に定義された各評価点E 11, E 12, E 13について、その周囲の特徴を把握する概念を示す平面図である。図5(a)は、長方形の右辺中央に設定された評価点E 11について、参考円C 1の内部と参考円C 2の内部の特徴を抽出した状態を示している。参考円C 1, C 2は、いずれも評価点E 11を中心とする円であるが、参考円C 1に比べて参考円C 2はより大きな円になっている。同様に、図5(b)は、長方形の右辺下方に設定された評価点E 12について、図5(c)は、長方形の下辺中央に設定された評価点E 13について、それぞれ2つの参考円C 1, C 2の内部の特徴を抽出した状態を示している。

[0073] ここで、各評価点について、参考円C 1の内部の特徴を比較すると、評価点E 1 1と評価点E 1 2については、左半分が図形内部（ハッチング領域）、右半分が図形外部（空白領域）になっており、参考円C 1の内部の特徴に関しては差がない。一方、評価点E 1 3についての参考円C 1の内部の特徴は、上半分が図形内部（ハッチング領域）、下半分が図形外部（空白領域）になっており、評価点E 1 1、E 1 2の参考円C 1内の特徴を90°回転させたものになっているが、ハッチング領域の占有率について差はない。一方、参考円C 2の内部の特徴を、各評価点E 1 1、E 1 2、E 1 3について比較すると、ハッチング領域の分布がそれぞれ異なっており、互いに異なる特徴を有していることがわかる。

[0074] このように、図形パターン10上の各評価点E 1 1、E 1 2、E 1 3について、その近傍の特徴を把握するにしても、参考円C 1のような狭い近傍領域の特徴を抽出するか、参考円C 2のように若干広い近傍領域の特徴を抽出するか、によって、抽出される特徴は異なる。したがって、ある1つの評価点Eについて、その近傍領域の特徴を何らかの特徴量xとして定量的に抽出する場合、近傍領域の範囲を段階的に変化させれば、より多様な方法で特徴量の抽出が行えることがわかる。

[0075] また、前述したとおり、元図形パターン10に基づいてリソグラフィプロセスを実行した場合、基板上に得られる実図形パターン20は、露光工程における近接効果により、元図形パターン10に対して寸法誤差（プロセスバイアス）を生じることになる。特に、電子線露光における近接効果には、影響範囲が狭い前方散乱に起因する効果や、影響範囲が広い後方散乱に起因する効果など、多様な効果が含まれている。たとえば、前方散乱は、レジスト層などから構成される被成形層に電子ビームを照射したときに、質量の小さい電子が、レジスト内で分子に散乱されながら広がっていく現象として説明され、後方散乱は、レジスト層の下にある金属基板などの表面付近で散乱されて跳ね返ってきた電子がレジスト層内で拡散してゆく現象として説明されている。また、エッチング工程によってもプロセスバイアスが生じ、その大

きさは、エッチング時のローディング現象に応じて変動する。このローディング現象も、上述した露光工程における近接効果と同様に、狭い影響範囲をもつ成分や広い影響範囲をもつ成分など、多様な成分が合わさって生じることになる。

[0076] 結局、ある1つの評価点Eについてのプロセスバイアス y の値は、様々なスケール感をもった現象が融合して決まる値になる。したがって、周囲の狭い範囲に関する特徴量から広い範囲に関する特徴量に至るまで、多様な特徴量を抽出することは、影響範囲がそれぞれ異なる、プロセスバイアスに影響を与える各工程中の様々な現象に対して、正確なシミュレーションを行う上で重要である。そこで、先願発明では、1つの評価点Eについて、近傍から遠方に至るまで、その周囲の様々な領域についての特徴量を抽出するようにしている。このように、1つの評価点Eについて複数の特徴量を抽出するため、先願発明では、それぞれ異なるサイズをもった複数の階層画像からなる「画像ピラミッド」を作成する方法を採用している。この画像ピラミッドには、影響範囲がそれぞれ異なる様々な現象を多重化した情報が含まれていることになる。

[0077] 図6は、図1に示す特徴量抽出ユニット120およびバイアス推定ユニット130において実行される処理の概要を示す図である。図の上方に示された元画像Q1は、図1に示す元画像作成部121によって作成された画像であり、与えられた元図形パターン10に相当する画像になる。前述したように、元図形パターン10は、半導体デバイスの設計装置などによって作成されるデータであり、図2(a)に示すような図形を示すデータになるが、通常、図形の輪郭線を示すベクトルデータ（各頂点の座標値と各頂点の連結関係を示すデータ）として与えられる。

[0078] 元画像作成部121は、与えられた元図形パターン10のデータに基づいて、それぞれ所定の画素値を有する画素の集合体からなる元画像Q1を作成する処理を実行する。たとえば、元図形パターン10を構成する図形の内部に画素値1をもった画素を配置し、外部に画素値0をもった画素を配置すれ

ば、多数の画素Uの集合体からなる元画像Q 1を作成することができる。図6に示す元画像Q 1は、このような画素Uの集合体からなる画像であり、破線で示すように、元図形パターン1 0に含まれる長方形の図形を画像情報として有している。また、評価点設定ユニット1 1 0によって、この図形の輪郭線上に評価点Eが設定されている。なお、図6では、便宜上、1つの評価点Eのみが描かれているが、実際には、図形の輪郭線に沿って、多数の評価点が設定されている。

[0079] 図1に示す画像ピラミッド作成部1 2 2は、この元画像Q 1に基づいて、画像ピラミッドP Pを作成する。画像ピラミッドP Pは、それぞれ異なるサイズをもった複数の階層画像によって構成される。図には、複数 n 通り ($n \geq 2$) の階層画像P 1 ~ P n によって構成された画像ピラミッドP Pが示されている。元画像Q 1から階層画像P 1 ~ P n を作成する具体的な手順は§ 2で説明するが、基本的には、階層画像P 1 ~ P n は、画素数を小さくする縮小処理によって作成される。たとえば、図示の例の場合、階層画像P 1は、元画像Q 1と同じサイズの画像（縦横の画素数が同じ画像）であるのに対して、階層画像P 2は、縮小された小さなサイズの画像になっており、階層画像P 3は、階層画像P 2を更に縮小した、より小さなサイズの画像になっている。

[0080] このように、先願発明では、元画像Q 1に基づいて、画像サイズが徐々に小さくなってゆくような階層画像P 1 ~ P n が作成される。このように、サイズの異なる複数の階層画像を上下に並べた様子は、図示のようなピラミッドの形態になるため、先願では、これら複数の階層画像P 1 ~ P n の集合体を画像ピラミッドP Pと呼んでいる。階層画像P 1 ~ P n は、いずれも元画像Q 1に基づいて作成されているため、元図形パターン1 0の情報を有しており、それぞれについて評価点Eの位置を定義することができる。図では、各階層画像P 1 ~ P n にそれぞれ長方形の図形が描かれており、その輪郭線上に評価点Eが配置されている。

[0081] 図1に示す特徴量算出部1 2 3は、画像ピラミッドを構成する各階層画像

について、評価点の近傍の画素の画素値に基づいて特徴量を算出する処理を行う。図6には、画像ピラミッドPPを構成するn枚の階層画像P1～Pnから、それぞれ評価点Eの特徴量 $x_1 \sim x_n$ が抽出された状態が示されている。図示されている特徴量 $x_1 \sim x_n$ は、いずれも同じ評価点Eの周囲の特徴を示す値であるが、特徴量 x_1 は、第1の階層画像P1上の評価点Eの近傍画素の画素値に基づいて算出された値であり、特徴量 x_2 は、第2の階層画像P2上の評価点Eの近傍画素の画素値に基づいて算出された値であり、特徴量 x_3 は、第3の階層画像P3上の評価点Eの近傍画素の画素値に基づいて算出された値である。各特徴量 $x_1 \sim x_n$ の具体的な算出手順は§2で説明する。

[0082] 図6には、便宜上、1つの評価点Eのみが描かれているが、実際には、多数の評価点のそれぞれについて、n個の特徴量 $x_1 \sim x_n$ が算出されることになる。個々の特徴量 $x_1 \sim x_n$ は、いずれも所定のスカラー値であるが、個々の評価点Eごとにそれぞれn個の特徴量 $x_1 \sim x_n$ が得られることになる。したがって、このn個の特徴量 $x_1 \sim x_n$ をn次元ベクトルとして把握すれば、特徴量抽出ユニット120は、1つの評価点Eについて、n次元ベクトルからなる特徴量を抽出する処理を行うことになる。

[0083] こうして抽出された各評価点についての特徴量（n次元ベクトル）は、バイアス推定ユニット130内の特徴量入力部131によって入力され、推定演算部132に与えられる。推定演算部132は、ここに示す実施例の場合、ニューラルネットワークによって構成され、予め実施された学習段階によって得られた学習情報Lに基づいて、n次元ベクトルとして与えられた特徴量 $x_1 \sim x_n$ に応じて、評価点Eについてのプロセスバイアスの推定値y（スカラー値）を算出する演算を行う。具体的な演算手順は§3で説明する。

[0084] このように、先願発明によれば、実際のリソグラフィプロセスに対して、物理的・実験的なシミュレーションモデルが構築されていなくても、あらゆる元図形パターン10に対して正確な特徴量を抽出することができる。また、そもそも物理的・実験的なシミュレーションモデルを構築する必要がない

ため、後に § 3. 2 で述べるニューラルネットワークの学習段階において、材料物性や工程条件の種々の設定値を考慮する必要もない。

[0085] 以上、先願発明に係る図形パターンの形状推定装置 100' および形状補正装置 100 を、半導体デバイスを製造するためのリソグラフィプロセスに利用する例について説明したが、先願発明や本願発明は、半導体デバイスの製造工程への利用に限定されるものではなく、リソグラフィプロセスを含む様々な製品の製造工程に利用することができる。たとえば、NIL (Nano Imprint Lithography) や EUV (Extreme UltraViolet Lithography) などのリソグラフィプロセスを含む様々な製品の製造工程においても先願発明や本願発明を利用することが可能である。特に、NIL においては、元図形パターンから露光リソグラフィを通して作製したマスターテンプレート上の実図形パターンが、元図形パターンと一致するように、元図形パターンに対して補正を行ってもよい。あるいは、マスターテンプレートからインプリントを通して作製したレプリカテンプレート上の実図形パターンが、元図形パターンと一致するように、元図形パターンに対する補正を行ってもよい。その他にも、先願発明や本願発明は、MEMS (Micro Electro Mechanical Systems)、LSPM (Large-size Photomask)、リードフレーム、メタルマスク、メタルメッシュセンサー、カラーフィルタなど、リソグラフィプロセスを含むすべての製品分野に適用可能である。

[0086] <<< § 2. 特徴量抽出ユニットの詳細 >>>

続いて、特徴量抽出ユニット 120 の詳細な処理動作を説明する。図 1 に示すように、特徴量抽出ユニット 120 は、元画像作成部 121、画像ピラミッド作成部 122、特徴量算出部 123 を有しており、図 4 の流れ図におけるステップ S3 の特徴量抽出処理を実行する機能を有している。この特徴量抽出処理は、実際には、図 7 に示す各手順により実行される。ここで、ステップ S31、S32 は、元画像作成部 121 によって実行される手順であり、ステップ S33～S36 は、画像ピラミッド作成部 122 によって実行される手順であり、ステップ S37 は、特徴量算出部 123 によって実行さ

れる手順である。以下、各部で実行される手順を、具体例を挙げて詳細に説明する。

[0087] <2. 1 元画像作成部 121 による処理手順>

元画像作成部 121 は、与えられた元図形パターン 10 に基づいて、それぞれ所定の画素値を有する画素の集合体からなる元画像を作成する機能を果たし、図 7 の流れ図のステップ S 31, S 32 を実行する。まず、ステップ S 31 において、元図形パターン 10 を入力する処理が行われ、続くステップ S 32 において、元画像作成処理が行われる。

[0088] § 1 では、説明の便宜上、ステップ S 31 で入力される元図形パターン 10 の一例として、図 2 (a) のような 1 つの長方形の図形のみからなる単純なパターンを示した。ここでは、より詳細な説明を行うため、図 8 に示すような 5 つの図形 F 1 ~ F 5 (矩形) を含む元図形パターン 10 が与えられた場合を考えてみよう。前述したように、図形パターンの形状補正装置 100 に与えられる元図形パターン 10 は、通常、図形の輪郭線を示すベクトルデータになる。したがって、図 8 では、各図形 F 1 ~ F 5 の内部にハッチングを施して示しているが、実際には、この元図形パターン 10 は、5 つの長方形 F 1 ~ F 5 の各頂点の座標値とこれら各頂点の連結関係を示すベクトルデータとして与えられる。

[0089] ステップ S 32 の元画像作成処理は、このようなベクトルデータとして与えられた元図形パターン 10 に基づいて、画素の集合体からなる元画像 Q 1 のデータ (ラスタデータ) を作成する処理ということになる。具体的には、元画像作成部 121 は、画素 U の二次元配列からなるメッシュを定義し、このメッシュ上に元図形パターン 10 を重ね合わせ、個々の画素 U の位置と元図形パターン 10 を構成する図形 F 1 ~ F 5 の輪郭線の位置との関係に基づいて、個々の画素 U の画素値を決定する処理を行う。

[0090] 図 9 は、元画像作成部 121 において、画素 U の二次元配列からなるメッシュ上に元図形パターン 10 を重ね合わせる処理が行われた状態を示す平面図である。この例では、縦横ともに画素寸法 u をもった画素 U を二次元配列

させたメッシュが定義されており、多数の画素 U が、縦横ともにピッチ u で並べられている。画素寸法 u は、各図形 $F1 \sim F5$ の形状を十分な解像度で表現することができる適切な値に設定される。画素寸法 u を小さく設定すればするほど、形状表現の解像度は向上するが、後の処理負担は重くなる。一般に、半導体デバイスを製造するために用いられる元図形パターン 10 は、極めて微細なパターンになるため、画素寸法 u としては、たとえば、 $u = 5 \sim 20 \text{ nm}$ 程度の値を設定するのが好ましい。

[0091] こうして、画素 U の二次元配列が定義されたら、図形 $F1 \sim F5$ の輪郭線の位置との関係に基づいて、個々の画素 U に画素値を定義する。画素値の定義には、いくつかの方法がある。

[0092] 最も基本的な定義方法は、元図形パターン 10 に基づいて、各図形 $F1 \sim F5$ の内部領域と外部領域とを認識し、各画素 U 内における内部領域の占有率を当該画素の画素値とする方法である。図8において、ハッチングが施された領域が各図形 $F1 \sim F5$ の内部領域であり、白い領域が外部領域である。したがって、この方法を採用する場合、図9のように重ね合わせた状態において、各画素の画素値は、当該画素内における内部領域（ハッチング領域）の占有率（ $0 \sim 1$ ）として定義される。このような方法で画素値が定義された画像は、一般に「面積密度マップ」と呼ばれている。

[0093] 図10は、図9に示す「元図形パターン 10 + 画素の二次元配列」に基づいて作成された面積密度マップ $M1$ を示す図である。ここで、各セルは、図9で定義された各画素であり、セル内の数字は各画素に定義された画素値である。なお、空白のセルは、画素値 0 をもつ画素である（画素値 0 の図示は省略されている）。この面積密度マップ $M1$ において、たとえば、画素値 1.0 をもつ画素は、図9におけるハッチング領域の占有率が 100% の画素であり、画素値 0.5 をもつ画素は、図9におけるハッチング領域の占有率が 50% の画素である。この面積密度マップ $M1$ は、基本的に、図形内部を画素値 1 、図形外部を画素値 0 で表現した二値画像ということになるが、図形の輪郭線が位置する画素については、内部領域の割合を示す値が画素値と

して与えられることになるので、全体としてはモノクロの階調画像ということになる。

[0094] 画素値の別な定義方法として、元図形パターン10に基づいて、各図形F1～F5の輪郭線を認識し、各画素U内に存在する輪郭線の長さを当該画素の画素値とする方法を採用することもできる。この方法を採用する場合、図9のように重ね合わせた状態において、各画素の画素値は、当該画素内に存在する輪郭線の長さの総和として定義される。このような方法で画素値が定義された画像は、一般に「エッジ長密度マップ」と呼ばれている。輪郭線の長さの単位としては、たとえば、画素寸法 u を1とした単位を用いればよい。

[0095] 図11は、図9に示す「元図形パターン10+画素の二次元配列」に基づいて作成されたエッジ長密度マップM2を示す図である。ここでも、各セルは、図9で定義された各画素であり、セル内の数字は各画素内に存在する輪郭線の長さの総和（画素寸法 u を1とした長さ）として定義された画素値である。なお、空白のセルは、画素値0をもつ画素である（画素値0の図示は省略されている）。このエッジ長密度マップM2において、たとえば、画素値1.0をもつ画素は、図9において、画素内に存在する輪郭線の長さが u となる画素である。このエッジ長密度マップM2は、基本的に、輪郭線の密度分布を示すモノクロの階調画像ということになるので、前述した面積密度マップM1とは性質がかなり異なる画像になる。ただ、輪郭線上に定義された評価点Eについての特徴量を抽出する上では、非常に有用な画像になる。

[0096] 以上、図8に示す元図形パターン10に基づいて、個々の画素の画素値を定義する方法を述べたが、リソグラフィプロセスに用いる元図形パターン10には、図形の内部と外部との境界を示す輪郭線の幾何学的な情報に加えて、更に、各図形に関するドーズ量の情報が付加されることがある。図12は、このようなドーズ量の情報を含んだ元図形パターン10を示す平面図である。図12に示すドーズ量付きの元図形パターン10は、図8に示す元図形パターン10と同様に、各図形F1～F5の輪郭線の情報を含んでいるが、それに加えて、各図形F1～F5のそれぞれについて、ドーズ量を定義する

情報を含んでいる。

[0097] 具体的には、図示の例の場合、図形 F 1 ～ F 3 についてはドーズ量 100 % が定義され、図形 F 4 についてはドーズ量 50 % が定義され、図形 F 5 についてはドーズ量 10 % が定義されている。これらのドーズ量は、リソグラフィプロセスの露光工程において、照射する光や電子線の強度（露光の回数によって総エネルギー量を制御する場合も含む）を示すものである。図示の例の場合、図形 F 1 ～ F 3 の内部領域を露光する際には 100 % の強度で光や電子線を照射するが、図形 F 4 の内部領域を露光する際には 50 %、図形 F 5 の内部領域を露光する際には 10 % の強度で光や電子線を照射することになる。このように、露光工程時に、個々の図形ごとにドーズ量を制御するようにすれば、実基板 S 上に形成される実図形パターン 20 の寸法を更に細かく調整することができる。

[0098] リソグラフィプロセスの露光工程において、このようなドーズ量の制御を行うケースでは、シミュレーションにおいても、ドーズ量を考慮する必要がある。このようなケースでは、元画像の画素値を決定する際に、ドーズ量を考慮した方法を採用する必要がある。すなわち、図形の内部と外部との境界を示す輪郭線の情報と、リソグラフィプロセスにおける各図形に関するドーズ量の情報と、を含む元図形パターン 10 が与えられた場合には、当該元図形パターン 10 に基づいて、各図形の内部領域と外部領域とを認識し、更に、各図形に関するドーズ量を認識し、各画素内に存在する各図形について「内部領域の占有率と当該図形のドーズ量との積」を求め、当該積の総和を当該画素の画素値とする方法を採用すればよい。

[0099] この方法を採用する場合、図 9 のように重ね合わせた状態において、各画素の画素値は、当該画素内における特定の図形の内部領域（ハッチング領域）の占有率（0 ～ 1）と当該特定の図形に関するドーズ量の積の総和として定義される。このような方法で画素値が定義された画像は、一般に「ドーズ密度マップ」と呼ばれている。このドーズ密度マップも、全体としてはモノクロの階調画像になる。

[0100] 図13は、図12に示すドーズ量付きの元図形パターン10に基づいて作成されたドーズ密度マップM3を示す図である。ここで、各セルは、図9で定義された各画素であり、セル内の数字は各画素に定義された画素値である。なお、空白のセルは、画素値0をもつ画素である（画素値0の図示は省略されている）。このドーズ密度マップM3を、図10に示す面積密度マップM1と比較すると、ドーズ量100%が与えられた図形F1～F3が配置された画素についての画素値に変わりはないが、ドーズ量50%が与えられた図形F4やドーズ量10%が与えられた図形F5が配置された画素についての画素値は、当該ドーズ量に応じた量だけ減じられていることがわかる。これは、実際の露光工程において、図形F4、F5の内部に対して、強度が減じられた光や電子線が照射される現象を示すものである。

[0101] 以上、元図形パターン10に基づいて、面積密度マップM1、エッジ長密度マップM2、ドーズ密度マップM3という3通りの画像（所定の画素値を有する画素Uの集合体）を作成する例を示した。図7の流れ図におけるステップS32の元画像作成処理では、これら3通りの画像のいずれかを作成して元画像とすればよい。図7では、一例として、面積密度マップM1を元画像Q1とした例が示されている。もちろん、上述した3通り以外の方法で元画像Q1を作成することも可能である。

[0102] なお、このステップS32で作成された元画像Q1は、ステップS33のフィルタ処理において、第1番目の準備画像Q1として利用される。ステップS33のフィルタ処理は、後述するように、第k番目の準備画像Qkに画像処理フィルタを作用させて、第k番目の階層画像Pkを作成する処理である。そこで、ステップS32の手順は、パラメータk（ $k = 1, 2, 3, \dots$ ）を初期値 $k = 1$ に設定して、第1番目の準備画像Q1を作成する処理とすることができる。この第1番目の準備画像Q1は、元図形パターン10に基づいて最初に作成された元画像であり、後述する画像ピラミッド作成工程において最初に用いられる基準の画像ということになる。

[0103] <2. 2 画像ピラミッド作成部122による処理手順>

続いて、画像ピラミッド作成部 122 による画像ピラミッドの作成処理の手順を説明する。画像ピラミッド作成部 122 は、図 7 のステップ S32 で作成された元画像 Q1（たとえば、図 10 に示す面積密度マップ M1）に基づいて、画素数を小さくする縮小処理を行う機能を有し、それぞれ異なるサイズをもった複数の階層画像からなる画像ピラミッドを作成する画像ピラミッド作成処理を行う。ここで述べる実施例の場合、この画像ピラミッド作成処理は、図 7 の流れ図のステップ S33～S36 に示す手順によって実行される。

[0104] ここでは、ステップ S31 において、図 8 に示すような元図形パターン 10 が入力され、ステップ S32 において、図 10 に示す面積密度マップ M1 が元画像 Q1 として作成された場合を例にとって、具体的な画像ピラミッド作成処理の手順を説明する。

[0105] まず、ステップ S33 では、第 k 番目の準備画像 Qk に画像処理フィルタを作用させて、第 k 番目の階層画像 Pk を作成するフィルタ処理が実行される。このフィルタ処理は、具体的には、たとえば、画像処理フィルタとしてガウシアンフィルタを用いた畳込演算として実行される。図 14 は、準備画像 Qk にガウシアンフィルタ GF33 を用いたフィルタ処理を施すことにより、第 k 番目の階層画像 Pk を作成する手順を示す平面図である。この図 14 に示されている第 k 番目の準備画像 Qk は、実際には、図 10 に示す面積密度マップ M1 と同じものである。

[0106] すなわち、図 10 に示す面積密度マップ M1 は、便宜上、8×8 の画素配列として示され、画素値 0 の記載が省略されているのに対して、図 14 に示す準備画像 Qk は、10×10 の画素配列として示され、画素値 0 も記載されているが、両者は実質的に同一の画像である。要するに、図 14 では、フィルタ処理を実行する都合上、図 10 に示す 8×8 の画素配列からなる面積密度マップ M1 の周囲に、画素値 0 を有する画素を配置して 10×10 の画素配列としているだけである。この段階では、パラメータ k は初期値 1 であり、図 14 に示す準備画像 Qk は、第 1 番目の準備画像 Q1 ということにな

る。上述したとおり、この第1番目の準備画像Q1は、ステップS32における元画像作成処理で作成された元画像に他ならない。

[0107] 図14に示すフィルタ処理では、ガウシアンフィルタGF33を用いた畳込演算が実行される。ガウシアンフィルタGF33は、図示のような3×3の画素配列であり、このガウシアンフィルタGF33を、第k番目の準備画像Qkの所定位置に重ね合わせて積和演算処理を行うことにより、第k番目の階層画像Pk（フィルタ処理画像）が得られる。図15は、図14に示すフィルタ処理により得られた第k番目の階層画像Pkを示す平面図である。この第k番目の階層画像Pkは、第k番目の準備画像Qkと同様に10×10の画素配列からなり、個々の画素の画素値は、ガウシアンフィルタGF33を用いた積和演算によって得られた値になる。

[0108] たとえば、図15に示す第k番目の階層画像Pkにおいて、太枠で囲った画素（第4行第3列の画素）に着目すると、この着目画素には画素値0.375が与えられている。当該画素値は、図14に太枠で囲った3×3の画素配列（第4行第3列の画素を中心とした9画素）上に、図示のガウシアンフィルタGF33を重ね、9画素のそれぞれについて、同位置に重ねられた画素同士の画素値の積を求め、9個の積の総和として求められたものである。具体的には、着目画素の画素値0.375は、「 $(1/16 \times 0) + (2/16 \times 0.25) + (1/16 \times 0.5) + (2/16 \times 0) + (4/16 \times 0.5) + (2/16 \times 1.0) + (1/16 \times 0) + (2/16 \times 0.25) + (1/16 \times 0.5)$ 」なる積和演算値として求められる。このような積和演算によるフィルタ処理は、画像についての畳込演算処理として一般的に知られた処理であるため、ここでは詳しい説明は省略する。

[0109] なお、図14に示すフィルタ処理では、画像処理フィルタとして、図16(a)に示すような3×3の画素配列からなるガウシアンフィルタGF33を用いた畳込演算を行っているが、図16(b)に示すような3×3の画素配列からなるラプラシアンフィルタLF33を用いた畳込演算を行うようにしてもよい。一般に、ガウシアンフィルタを用いたフィルタ処理は、画像の輪郭を

ボケさせる効果を与え、ラプラシアンフィルタを用いたフィルタ処理は、画像の輪郭を強調させる効果を与えることが知られている。いずれのフィルタ処理を採用しても、第 k 番目の準備画像 Q_k に対して若干特徴が異なる第 k 番目の階層画像 P_k を得ることができるので、それぞれ異なる特徴をもった複数の階層画像からなる画像ピラミッドを作成する上で効果的である。

[0110] もちろん、ステップS33のフィルタ処理に用いる画像処理フィルタは、図16(a)に示すガウシアンフィルタ GF_{33} や図16(b)に示すラプラシアンフィルタ LF_{33} に限定されるものではなく、この他にも種々の画像処理フィルタを利用することが可能である。また、用いる画像処理フィルタのサイズも、 3×3 の画素配列に限定されることなく、任意のサイズの画像処理フィルタを利用することが可能である。たとえば、図17(a)に示すような 5×5 の画素配列からなるガウシアンフィルタ GF_{55} や、図17(b)に示すような 5×5 の画素配列からなるラプラシアンフィルタ LF_{55} を利用してもよい。

[0111] こうして、図7の手順におけるステップS33のフィルタ処理が完了すると、ステップS34において、パラメータ k が所定の設定値 n に到達したか否かが判断され、 $k < n$ の場合には、ステップS35の縮小処理が実行される。この縮小処理は、所定の対象画像を元に、当該対象画像よりも画素数の小さな画像を作成する処理である。ここに示す実施例の場合、ステップS33のフィルタ処理で作成された第 k 番目の階層画像 P_k に対して縮小処理を実行することにより、第 $(k+1)$ 番目の準備画像 $Q_{(k+1)}$ を作成する処理が実行される。したがって、準備画像 $Q_{(k+1)}$ は、階層画像 P_k よりもサイズが小さな画像（画素配列における縦横の画素数が小さな画像）になる。

[0112] このような画像についての縮小処理は、「プーリング処理」とも呼ばれ、ステップS35で行う縮小処理としては、たとえば、「アベレージ・プーリング処理」を採用することができる。図18は、第 k 番目の階層画像 P_k に対して、アベレージ・プーリング処理を施すことにより、縮小画像として、

第 $(k + 1)$ 番目の準備画像 $Q(k + 1)$ を作成する手順を示す平面図である。具体的には、図 18(a) に示す 4×4 の画素配列からなる階層画像 P_k にアベレージ・プーリング処理（縮小処理）を施すことにより、図 18(b) に示すような 2×2 の画素配列からなる準備画像 $Q(k + 1)$ が縮小画像として作成されている。

[0113] 図 18 に示すアベレージ・プーリング処理は、 2×2 の画素配列からなる 4 個の画素を 1 つの画素に変換（縮小）する処理であり、元の 4 個の画素の画素値の平均値を、変換後の 1 つの画素の画素値とすることにより、縮小画像が作成されることになる。たとえば、図 18(a) に示す階層画像 P_k の左上に配置されている 2×2 の画素配列からなる 4 個の画素（太枠内の画素）は、図 18(b) の準備画像 $Q(k + 1)$ 上では太枠で示す 1 画素に変換（縮小）されている。この変換（縮小）後の太枠画素の画素値 0.5 は、元の 4 個の画素の画素値の平均値になっている。

[0114] 一方、図 19 は、第 k 番目の階層画像 P_k に対して、マックス・プーリング処理を施すことにより、縮小画像として、第 $(k + 1)$ 番目の準備画像 $Q(k + 1)$ を作成する手順を示す平面図である。具体的には、図 19(a) に示す 4×4 の画素配列からなる階層画像 P_k にマックス・プーリング処理（縮小処理）を施すことにより、図 19(b) に示すような 2×2 の画素配列からなる準備画像 $Q(k + 1)$ が縮小画像として作成されている。

[0115] 図 19 に示すマックス・プーリング処理は、図 18 に示すアベレージ・プーリング処理と同様に、 2×2 の画素配列からなる 4 個の画素を 1 つの画素に変換（縮小）する処理であるが、元の 4 個の画素の画素値の最大値を、変換後の 1 つの画素の画素値とすることにより、縮小画像が作成されることになる。たとえば、図 19(a) に示す階層画像 P_k の左上に配置されている 2×2 の画素配列からなる 4 個の画素（太枠内の画素）は、図 19(b) の準備画像 $Q(k + 1)$ 上では太枠で示す 1 画素に変換（縮小）されている。この変換（縮小）後の太枠画素の画素値 1.0 は、元の 4 個の画素の画素値の最大値になっている。

[0116] なお、図18および図19に示す各プーリング処理は、 2×2 の画素配列からなる4個の画素を単一の画素に変換する縮小処理であるが、もちろん、 3×3 の画素配列からなる9個の画素を単一の画素に変換する縮小処理を行うことも可能であるし、 3×2 の画素配列からなる6個の画素を単一の画素に変換する縮小処理を行うことも可能である。

[0117] 結局、画像ピラミッド作成部122は、ステップS35の縮小処理として、複数 m 個の隣接画素を、これら複数 m 個の隣接画素の画素値の平均値を画素値とする単一の画素に置き換えるアベレージ・プーリング処理を実行することにより縮小画像を作成することもできるし、複数 m 個の隣接画素を、これら複数 m 個の隣接画素の画素値の最大値を画素値とする単一の画素に置き換えるマックス・プーリング処理を実行することにより縮小画像を作成することもできる。もちろん、ステップS35の縮小処理としては、その他の縮小処理を行うことも可能である。要するに、階層画像 P_k に対して画素数を小さくするような変換を施すことにより、サイズの小さな準備画像 $Q(k+1)$ を作成することができる処理、別言すれば、「画素数が小さくなった縮小画像」を作成する処理であれば、ステップS35において、どのような縮小処理を実行してもかまわない。

[0118] こうして、ステップS35の縮小処理が完了したら、ステップS36において、パラメータ k が1だけ増加され、再びステップS33のフィルタ処理が実行される。結局、上述のように、 $k=1$ として第1番目の準備画像 Q_1 （元画像）に対してステップS33でフィルタ処理を行うことにより第1番目の階層画像 P_1 が作成され、続いて、ステップS35で、この階層画像 P_1 に対して縮小処理を行うことにより第2番目の準備画像 Q_2 が作成され、ステップS36で $k=2$ に更新され、再びステップS33において、第2番目の準備画像 Q_2 に対するフィルタ処理を行うことにより第2番目の階層画像 P_2 が作成されることになる。

[0119] このような繰り返し手順が、ステップS34において、 $k=n$ と判断されるまで繰り返し実行される。ここで、 n の値としては、画像ピラミッドの階

層数（すなわち、画像ピラミッドを構成する階層画像の総数）として適切な値を予め設定しておけばよい。 n の値を大きく設定すればするほど、1つの評価点Eについて抽出される特徴量の数 n が多くなるので、より正確なシミュレーションが可能になるが、演算負担は増大する。また、ステップS35の縮小処理を繰り返すほど、画像のサイズはどんどん小さくなってゆくのので、 n の値を大きく設定しすぎると、ステップS35の縮小処理を行うことができなくなる。したがって、実用上は、元画像Q1のサイズや演算負担を考慮して、 n の値を適切に設定すればよい。

[0120] こうして、ステップS34において、 $k = n$ と判断されると、画像ピラミッド作成部122による処理は完了である。この時点で、図6に示すように、複数 n 通りの階層画像P1～P n からなる画像ピラミッドPPが作成されたことになる。そこで、ステップS34からステップS37へと進み、特徴量算出処理が実行される。

[0121] 図20は、画像ピラミッド作成部122において、 n 通りの階層画像P1～P n からなる画像ピラミッドPPを作成する手順（図7のステップS33～S36の手順）を示す平面図である。図の上段左に示す第1の準備画像Q1は、ステップS32の元画像作成処理において、 $k = 1$ として作成された元画像であり、個々の画素には、たとえば図10に示す面積密度マップM1のような画素値が定義されている。上述したとおり、ステップS33では、この第1の準備画像Q1に対してフィルタ処理が行われる。具体的には、たとえば、 3×3 の画素配列からなるガウシアンフィルタGF33を用いた畳込演算により、図20の上段右に示すような第1の階層画像P1が作成される。この第1の階層画像P1のサイズは、第1の準備画像Q1のサイズと同じである。

[0122] 続いて、ステップS35の縮小処理において、第1の階層画像P1に対する縮小処理（たとえば、アベレージ・プーリング処理）が行われ、図20の中段左に示す第2の準備画像Q2が作成される。この第2の準備画像Q2のサイズは、第1の階層画像P1のサイズよりも小さなものになる。続いて、

ステップS 3 6において、パラメータ k の値が2に更新され、再びステップS 3 3のフィルタ処理が実行される。すなわち、 3×3 の画素配列からなるガウシアンフィルタ $G F 3 3$ を用いた畳込演算により、図20の中段右に示すような第2の階層画像 $P 2$ が作成される。この第2の階層画像 $P 2$ のサイズは、第2の準備画像 $Q 2$ のサイズと同じである。

[0123] そして再びステップS 3 5の縮小処理が実行される。すなわち、第2の階層画像 $P 2$ に対する縮小処理（たとえば、アベレージ・プーリング処理）が行われ、図20の下段左に示す第3の準備画像 $Q 3$ が作成される。この第3の準備画像 $Q 3$ のサイズは、第2の階層画像 $P 2$ のサイズよりも小さなものになる。続いて、ステップS 3 6において、パラメータ k の値が3に更新され、再びステップS 3 3のフィルタ処理が実行される。すなわち、 3×3 の画素配列からなるガウシアンフィルタ $G F 3 3$ を用いた畳込演算により、図20の下段右に示すような第3の階層画像 $P 3$ が作成される。この第3の階層画像 $P 3$ のサイズは、第3の準備画像 $Q 3$ のサイズと同じである。

[0124] このような処理が、パラメータ $k = n$ になるまで繰り返し実行され、最終的に、第 n の準備画像 $Q n$ と第 n の階層画像が得られることになる。こうして、第1の階層画像 $P 1 \sim$ 第 n の階層画像 $P n$ までのサイズが異なる n 通りの階層画像によって、画像ピラミッド $P P$ が構成されることになる。

[0125] 結局、図7の手順に示す実施例の場合、画像ピラミッド作成部122は、元画像 $Q 1$ もしくは縮小画像 $Q (k + 1)$ に対して、所定の画像処理フィルタを用いたフィルタ処理を行う機能を有しており、このフィルタ処理と縮小処理とを交互に実行することにより、複数の階層画像 $P 1 \sim P n$ からなる画像ピラミッド $P P$ を作成することになる。

[0126] より具体的には、画像ピラミッド作成部122は、元画像作成部121によって作成された元画像を第1の準備画像 $Q 1$ とし、第 k の準備画像 $Q k$ （但し、 k は自然数）に対するフィルタ処理によって得られる画像を第 k の階層画像 $P k$ とし、第 k の階層画像 $P k$ に対する縮小処理によって得られる画像を第 $(k + 1)$ の準備画像 $Q (k + 1)$ として、第 n の階層画像 $P n$ が得

られるまでフィルタ処理と縮小処理とを交互に実行することにより、第1の階層画像P1～第nの階層画像Pnを含む複数n通りの階層画像からなる画像ピラミッドPPを作成することになる。

[0127] 先願発明における画像ピラミッドPPは、それぞれ異なるサイズをもった複数の階層画像によって構成されるものであればよいので、図7の流れ図に示す手順において、ステップS35の縮小処理は必須の処理になるが、ステップS33のフィルタ処理は必ずしも必要な処理ではない。ただ、フィルタ処理を行うと、個々の画素の画素値に周囲の画素の画素値の影響を作用させることができる。このため、フィルタ処理を加えることにより、バリエーションに富んだ複数の階層画像を作成することができ、より多様な情報を含んだ特徴量を抽出することができるようになり、結果的に、より正確なシミュレーションが可能になる。したがって、実用上は、図7の流れ図に示す手順のように、縮小処理とフィルタ処理とを交互に実行するようにするのが好ましい。

[0128] <2.3 特徴量算出部123による処理手順>

次に、特徴量算出部123による特徴量算出処理の手順を説明する。特徴量算出部123は、図7のステップS37に示すとおり、画像ピラミッドPPを構成する各階層画像P1～Pnに基づいて、各評価点Eについての特徴量 $x_1 \sim x_n$ を算出する処理を行う。ここでは、この特徴量 $x_1 \sim x_n$ の算出処理の手順を具体的に説明する。

[0129] 図21は、特徴量算出部123において、各階層画像P1～Pnから特定の評価点Eについての特徴量 $x_1 \sim x_n$ を算出する手順を示す平面図である。具体的には、図21(a)～(c)の左側には、それぞれ第1の階層画像P1、第2の階層画像P2、第3の階層画像P3の各画素配列に、元図形パターン10を構成する長方形（太枠で示す）を重ねた状態が示されており、図21(a)～(c)の右側には、各階層画像P1、P2、P3に基づいて、特定の評価点Eについての特徴量 x_1 、 x_2 、 x_3 を算出する原理が示されている。

[0130] 図21に示す各階層画像 P_1 , P_2 , P_3 は、画像ピラミッド PP の各階層を構成する画像の一部である。実際には、 $P_1 \sim P_n$ までの n 通りの階層画像が用意され、 n 組の特徴量 $x_1 \sim x_n$ が抽出されることになるが、図21では、説明の便宜上、3枚の階層画像 P_1 , P_2 , P_3 から3組の特徴量 x_1 , x_2 , x_3 を抽出する様子が示されている。

[0131] ここで、第1の階層画像 P_1 は、元画像 Q_1 （第1の準備画像）に対してフィルタ処理を施すことによって得られた画像であり、図示の例では、 16×16 の画素配列を有している。これに対して、第2の階層画像 P_2 は、第1の階層画像 P_1 に対して縮小処理およびフィルタ処理を施すことによって得られた画像であり、図示の例では、 8×8 の画素配列を有している。また、第3の階層画像 P_3 は、第2の階層画像 P_2 に対して縮小処理およびフィルタ処理を施すことによって得られた画像であり、図示の例では、 4×4 の画素配列を有している。

[0132] 図21では、各階層画像 P_1 , P_2 , P_3 を、その輪郭が同じ大きさの正方形になるように描いているため、いずれも同じ大きさの画像になっているが、画素配列としては、 16×16 , 8×8 , 4×4 と徐々に縮小しており、画像のサイズは徐々に減少したものになっている。ただ、図21では、各階層画像 P_1 , P_2 , P_3 の外枠を、同じ大きさの正方形として描いているため、画素の大きさが徐々に大きくなっている。別言すれば、画像の解像度は、階層画像 P_1 , P_2 , P_3 の順に低下してゆき、徐々に粗い画像になってゆく。

[0133] 上述したとおり、図には、元図形パターン10を構成する長方形が太枠で描かれている。各階層画像 P_1 , P_2 , P_3 は、いずれも画素の集合体からなるラスタ画像であるので、図に太枠で描かれた長方形の輪郭線は、実際には、輪郭線そのものの情報として含まれているわけではなく、個々の画素の画素値の情報として含まれていることになる。ただ、図21では、説明の便宜上、各階層画像 P_1 , P_2 , P_3 上の長方形の位置を太線で示してある。ここでは、この長方形の輪郭線上に定義された特定の評価点Eについて、

特徴量 $x_1 \sim x_n$ を抽出する処理を説明する。

[0134] 図 2 1 (a) ～(c) を見ればわかるとおり、各階層画像 P_1 , P_2 , P_3 に対して、太枠で示す長方形は同じ相対位置に配置されており、特定の評価点 E も同じ相対位置に配置されている。ここに示す実施例の場合、1つの評価点 E についての特徴量は、その近傍の画素の画素値に基づいて算出される。

[0135] まず、図 2 1 (a) に示すように、第 1 の階層画像 P_1 に基づいて、評価点 E についての特徴量 x_1 が抽出される。具体的には、特徴量算出部 1 2 3 は、図 2 1 (a) の右側に示すように、第 1 の階層画像 P_1 を構成する画素から、評価点 E の近傍に位置する 4 個の画素（図にハッチングを施した画素）を着目画素として抽出し、これら 4 個の着目画素の画素値を用いた演算により特徴量 x_1 を算出する。同様に、図 2 1 (b) の右側に示すように、第 2 の階層画像 P_2 を構成する画素から、評価点 E の近傍に位置する 4 個の画素（図にハッチングを施した画素）を着目画素として抽出し、これら 4 個の着目画素の画素値を用いた演算により特徴量 x_2 を算出する。また、図 2 1 (c) の右側に示すように、第 3 の階層画像 P_3 を構成する画素から、評価点 E の近傍に位置する 4 個の画素（図にハッチングを施した画素）を着目画素として抽出し、これら 4 個の着目画素の画素値を用いた演算により特徴量 x_3 を算出する。

[0136] このような処理を、第 4 の階層画像 P_4 ～第 n の階層画像 P_n についても行えば、特定の評価点 E について、 n 組の特徴量 $x_1 \sim x_n$ を抽出することができる。これら n 組の特徴量 $x_1 \sim x_n$ は、いずれも元図形パターン 1 0 上の同じ評価点 E について、その周囲の特徴を示すパラメータになるが、元図形パターン 1 0 から影響を受ける範囲が互いに異なっている。たとえば、第 1 の階層画像 P_1 から抽出された特徴量 x_1 は、図 2 1 (a) の右側の図にハッチングが施された狭い領域内の特徴を示す値になるが、第 2 の階層画像 P_2 から抽出された特徴量 x_2 は、図 2 1 (b) の右側の図にハッチングが施されたより広い領域内の特徴を示す値になり、第 3 の階層画像 P_3 から抽出された特徴量 x_3 は、図 2 1 (c) の右側の図にハッチングが施された更に広

い領域内の特徴を示す値になる。

[0137] 前述したように、ある1つの評価点Eについてのプロセスバイアス y の値は、前方散乱、後方散乱など、様々なスケール感をもった現象が融合して決まる値になる。したがって、同一の評価点Eについての特徴量として、その周囲のごく狭い範囲に関する特徴量 x_1 から、より広い範囲に関する特徴量 x_n に至るまで、多様な特徴量 $x_1 \sim x_n$ を抽出すれば、影響範囲がそれぞれ異なる様々な現象を考慮した正確なシミュレーションを行うことができる。図21には、1つの評価点Eについて、 n 組の特徴量 $x_1 \sim x_n$ を抽出する処理が示されているが、実際には、元図形パターン10上に定義された多数の評価点のそれぞれについて、 n 組の特徴量 $x_1 \sim x_n$ が同様の手順によって抽出されることになる。

[0138] 評価点Eの近傍にある着目画素の画素値に基づいて特徴量 x を算出する方法としては、着目画素の画素値の単純平均を特徴量 x とする単純な方法を採用することができる。たとえば、図21(a)に示すように、第1の階層画像P1から評価点Eについての特徴量 x_1 を抽出するには、図にハッチングを施して示した着目画素（評価点Eの近傍にある4個の画素）の画素値の単純平均を特徴量 x_1 とすればよい。ただ、より正確な特徴量を算出するためには、評価点Eと各着目画素との距離に応じた重みを考慮した加重平均を求め、この加重平均の値を特徴量 x_1 とするのが好ましい。

[0139] 図22は、図21に示す特徴量算出手順で用いる具体的な演算方法（加重平均の値を特徴量とする演算方法）を示す図である。ここでは、特定の評価点Eの近傍に位置する着目画素として、4個の画素A、B、C、Dが選択された例が示されている。具体的には、処理対象となる階層画像P上において、評価点Eに近い順に合計4個の画素を選択する処理を行えば、着目画素A、B、C、Dを決定することができる。そこで、この4個の着目画素A、B、C、Dの画素値について、評価点Eと各画素との距離に応じた重みを考慮した加重平均を特徴量 x とする演算を行えばよい。

[0140] 図22(a)には、各着目画素A、B、C、Dの中心点に x 印が表示され、

これら×印を連結する破線が描かれている。各着目画素A, B, C, Dの画素寸法は縦横ともにuであり、上記破線は、この画素寸法uをもった画素を半分に分割する分割線になっている。ここに示す実施例の場合、評価点Eと各着目画素A, B, C, Dとの距離として、評価点Eと各着目画素A, B, C, Dの中心点との横方向距離および縦方向距離を採用している。具体的には、図22(a)に示す例の場合、着目画素Aについては、横方向距離a, 縦方向距離cになり、着目画素Bについては、横方向距離b, 縦方向距離cになり、着目画素Cについては、横方向距離a, 縦方向距離dになり、着目画素Dについては、横方向距離b, 縦方向距離dになる。

[0141] この場合、評価点Eの特徴量xは、各着目画素A, B, C, Dの画素値を同じ符号A, B, C, Dで表し、画素寸法u（画素ピッチ）を1とすれば、図22(b)に示すとおり、

$$G = (A \cdot b + B \cdot a) / 2$$

$$H = (C \cdot b + D \cdot a) / 2$$

$$x = (G \cdot d + H \cdot c) / 2$$

なる演算によって求めることができる。

[0142] もちろん、4個の着目画素A, B, C, Dの画素値から特徴量xを算出する方法は、この図22に例示する方法に限定されるものではなく、評価点Eの近傍の画素の画素値を反映した特徴量xを算出することができるのであれば、この他にも様々な算出方法を採ることが可能である。また、図21および図22に例示する算出方法では、着目画素として、評価点Eの近傍に位置する4個の画素を選択しているが、特徴量xの算出に利用する着目画素の数は4個に限定されるものではない。たとえば、評価点Eの近傍に位置する3×3の画素配列を構成する9個の画素を着目画素として選択し、これら9個の着目画素の画素値について、評価点Eとの距離に応じた重みを考慮した加重平均を求め、これを評価点Eについての特徴量xとすることも可能である。

[0143] 一般論として、特徴量算出部123は、特定の階層画像P上の特定の評価

点Eについての特徴量 x を算出する際に、当該特定の階層画像 P を構成する画素から、当該特定の評価点Eに近い順に合計 j 個の画素を着目画素として抽出し、抽出した j 個の着目画素の画素値について、当該特定の評価点Eと各着目画素との距離に応じた重みを考慮した加重平均を求める演算を行い、得られた加重平均の値を特徴量 x とすることができる。

[0144] <2. 4 特徴量抽出処理の変形例>

ここでは、これまで述べてきた特徴量抽出処理の手順についての変形例をいくつか述べておく。

[0145] (1) 差分画像 D_k を階層画像とする変形例

§ 2. 2では、画像ピラミッド作成部122による画像ピラミッド PP の作成処理手順として、図7の流れ図に示すステップS33～S36の処理を例示した。この処理は、元画像 Q_1 （第1の準備画像）を出発点として、フィルタ処理と縮小処理を交互に実行し、図20に示すような手順により、フィルタ処理によって作成される n 枚の画像（以下、フィルタ処理画像と呼ぶ）を、そのまま階層画像 $P_1 \sim P_n$ として採用し、画像ピラミッド PP を作成する処理である。

[0146] これに対して、ここで述べる変形例は、図7の流れ図に示すステップS33～S36の処理を基本としつつ、更に、ステップS33のフィルタ処理が完了した時点で、フィルタ処理により得られた第 k のフィルタ処理画像 P_k （§ 2. 2では、第 k の階層画像 P_k と呼んでいた画像）から第 k の準備画像 Q_k を減じる差分演算「 $P_k - Q_k$ 」を行って第 k の差分画像 D_k を求める処理が付加される。別言すれば、ここで述べる変形例では、図7の流れ図に示すステップS33～S36の処理がそのまま実行されることになるが、更に、第 k のフィルタ処理画像 P_k から第 k の準備画像 Q_k を減じる差分演算「 $P_k - Q_k$ 」が余分に行われることになる。

[0147] ここで、差分演算「 $P_k - Q_k$ 」は、第 k のフィルタ処理画像 P_k と第 k の準備画像 Q_k とについて、画素配列上で同じ位置に配置された画素を対応画素と定義し、画像 P_k 上の各画素の画素値から画像 Q_k 上の対応画素の画

素値を減算して差分をとり、得られた差分を画素値とする新たな画素の集合体からなる差分画像 D_k を求める処理である。

[0148] 図23は、このような差分演算「 $P_k - Q_k$ 」によって、 n 通りの差分画像 $D_1 \sim D_n$ からなる画像ピラミッド PD を作成する手順を示す平面図である。ここで、上段右側に示す第1の階層画像 D_1 は、差分演算「 $P_1 - Q_1$ 」によって得られる差分画像であり、具体的には、図20の上段右側に示す第1のフィルタ処理画像 P_1 （図20では、第1の階層画像 P_1 と呼ばれている）から上段左側に示す第1の準備画像 Q_1 を減じる差分演算（対応位置にある画素同士の画素値の引き算）によって算出される。

[0149] 同様に、図23の中段右側に示す第2の階層画像 D_2 は、差分演算「 $P_2 - Q_2$ 」によって得られる差分画像であり、具体的には、図20の中段右側に示す第2のフィルタ処理画像 P_2 （図20では、第2の階層画像 P_2 と呼ばれている）から中段左側に示す第2の準備画像 Q_2 を減じる差分演算によって算出される。また、図23の下段右側に示す第3の階層画像 D_3 は、差分演算「 $P_3 - Q_3$ 」によって得られる差分画像であり、具体的には、図20の下段右側に示す第3のフィルタ処理画像 P_3 （図20では、第3の階層画像 P_3 と呼ばれている）から下段左側に示す第3の準備画像 Q_3 を減じる差分演算によって算出される。以下、同様の差分演算が行われ、最終的に、差分演算「 $P_n - Q_n$ 」によって得られる差分画像が、第 n の階層画像 D_n ということになる。

[0150] §2.2で述べた実施例では、図20に示すとおり、第1の階層画像（第1のフィルタ処理画像） $P_1 \sim$ 第 n の階層画像（第 n のフィルタ処理画像） P_n によって画像ピラミッド PP が構成されていたが、ここで述べる変形例では、図23に示すとおり、第1の階層画像（第1の差分画像） $D_1 \sim$ 第 n の階層画像（第 n の差分画像） D_n によって画像ピラミッド PD が構成されることになる。

[0151] 上述したように、ここで述べる変形例を実施するには、§2.2で述べた実施例の手順に、更に、差分演算の手順を付加すればよい。具体的には、画

像ピラミッド作成部 1 2 2 は、元画像作成部 1 2 1 によって作成された元画像を第 1 の準備画像 Q_1 とし、第 k の準備画像 Q_k (但し、 k は自然数) に対するフィルタ処理によって得られるフィルタ処理画像 P_k と第 k の準備画像 Q_k との差分画像 D_k を求め、当該差分画像 D_k を第 k の階層画像 D_k とし、第 k のフィルタ処理画像 P_k に対する縮小処理によって得られる画像を第 $(k+1)$ の準備画像 $Q_{(k+1)}$ として、第 n の階層画像 D_n が得られるまでフィルタ処理と縮小処理とを交互に実行することにより、第 1 の階層画像 D_1 ~ 第 n の階層画像 D_n を含む複数 n 通りの階層画像からなる画像ピラミッド PD を作成すればよい。

[0152] 結局、この変形例において画像ピラミッド PD の第 k 番目の階層を構成する第 k の階層画像 D_k は、フィルタ処理後の画像 (フィルタ処理画像 P_k) とフィルタ処理前の画像 (準備画像 Q_k) との差分画像ということになり、各画素の画素値は、フィルタ処理前後の画素値の差に相当する。すなわち、§ 2. 2 で述べた実施例における第 k の階層画像 P_k が、フィルタ処理後の画像自体を示しているのに対して、ここで述べる変形例における第 k の階層画像 D_k は、フィルタ処理によって生じた差を示すことになる。このように、§ 2. 2 で述べた実施例で作成される画像ピラミッド PP と、ここで述べた変形例で作成される画像ピラミッド PD とは、その構成要素となる階層画像の意味合いが大きく異なることになるが、いずれも評価点 E についての何らかの特徴を示す画像である点に変わりはない。したがって、ここで述べた変形例で作成される各階層画像 D_1 ~ D_n から、特徴量の抽出を行うことが可能であり、本変形例における特徴量算出部 1 2 3 は、各階層画像 D_1 ~ D_n から特徴量 x_1 ~ x_n を抽出する処理を行うことになる。

[0153] (2) 複数通りのアルゴリズムにより複数通りの画像ピラミッドを作成する変形例

§ 2. 2 で述べた実施例では、図 20 に示すとおり、元画像 (第 1 の準備画像 Q_1) に対してフィルタ処理および縮小処理を交互に実行することにより各フィルタ処理画像 P_1 ~ P_n を求め、これら各フィルタ処理画像 P_1 ~

P_n をそのまま n 通りの階層画像 $P_1 \sim P_n$ とするアルゴリズムにより、画像ピラミッド PP が作成されている。これに対して、§ 2. 4 (1) で述べた差分画像 D_k を階層画像とする変形例では、図 23 に示すとおり、フィルタ処理画像 P_k と準備画像 Q_k との差分演算を行うことにより各差分画像 $D_1 \sim D_n$ を求め、これら各差分画像 $D_1 \sim D_n$ を n 通りの階層画像 $D_1 \sim D_n$ とするアルゴリズムにより、画像ピラミッド PD が作成されている。

[0154] また、フィルタ処理に用いる画像フィルタには、図 16 や図 17 に示すように様々な種類があり、縮小処理（プーリング処理）の方法にも、図 18 や図 19 に示すように様々な種類がある。このように、出発点が同じ元画像（第 1 の準備画像 Q_1 ）であったとしても、採用するアルゴリズムによって、得られる画像ピラミッドを構成する各階層画像の内容は異なってくる。しかも、先願発明を実行する上で、利用する画像ピラミッドは必ずしも 1 つである必要はなく、複数通りのアルゴリズムにより複数の画像ピラミッドを作成し、個々の画像ピラミッドからそれぞれ特徴量を抽出することも可能である。

[0155] すなわち、画像ピラミッド作成部 122 に、1 つの元画像（第 1 の準備画像 Q_1 ）について、互いに異なる複数通りのアルゴリズムに基づく画像ピラミッド作成処理を行う機能をもたせておき、複数通りの画像ピラミッドが作成されるようにしてもよい。この場合、特徴量算出部 123 は、この複数通りの画像ピラミッドのそれぞれを構成する各階層画像について、評価点の位置に応じた画素（評価点の周囲に位置する画素）の画素値に基づいて特徴量を算出する処理を行うようにすればよい。

[0156] たとえば、画像ピラミッド作成部 122 が画像ピラミッド作成処理を行う際に、主アルゴリズムとして、§ 2. 2 で述べた実施例のアルゴリズムを採用すれば、図 20 に示すように、 n 通りの主階層画像 $P_1 \sim P_n$ （フィルタ処理画像）によって構成される主画像ピラミッド PP を作成することができ、副アルゴリズムとして、§ 2. 4 (1) で述べた差分画像 D_k を階層画像とする変形例のアルゴリズムを採用すれば、図 23 に示すように、 n 通りの副

階層画像 $D_1 \sim D_n$ （差分画像）によって構成される副画像ピラミッド P_D を作成することができる。したがって、画像ピラミッド作成部 122 が、上記 2 通りのアルゴリズムを用いて画像ピラミッド作成処理を行うようにすれば、主画像ピラミッド P_P と副画像ピラミッド P_D との 2 通りの画像ピラミッドを作成することができる。

[0157] ここで、図 14 に例示するようなガウシアンフィルタを用いたフィルタ処理により作成された主画像ピラミッド P_P は、ガウシアンピラミッドと呼ぶことができる。また、差分画像を用いて構成された副画像ピラミッド P_D は、ラプラシアンピラミッドと呼ぶことができる。ガウシアンピラミッドとラプラシアンピラミッドは、互いに性質が大きく異なる画像ピラミッドになるので、これらを主画像ピラミッド P_P および副画像ピラミッド P_D として採用し、2 通りの画像ピラミッドを利用して特徴量の抽出を行うようにすれば、より多様性をもった特徴量の抽出が可能になる。

[0158] 一方、特徴量算出部 123 が、主画像ピラミッド P_P を構成する主階層画像 $P_1 \sim P_n$ および副画像ピラミッド P_D を構成する副階層画像 $D_1 \sim D_n$ について、それぞれ評価点 E の近傍の画素の画素値に基づいて特徴量を算出する処理を行うようにすれば、主階層画像 $P_1 \sim P_n$ から算出された特徴量 $x_{p1} \sim x_{pn}$ と副階層画像 $D_1 \sim D_n$ から算出された特徴量 $x_{d1} \sim x_{dn}$ とが得られる。すなわち、1 つの評価点 E について、合計 $2n$ 個の特徴量が抽出されることになる。この場合、推定演算部 132 に対しては、1 つの評価点 E についての特徴量が、 $2n$ 次元ベクトルとして与えられることになるので、より正確な推定演算を行うことが可能になる。

[0159] 結局、上述した変形例を実施するには、画像ピラミッド作成部 122 が、元画像作成部 121 によって作成された元画像を第 1 の準備画像 Q_1 とし、第 k の準備画像 Q_k （但し、 k は自然数）に対するフィルタ処理によって得られる画像を第 k の主階層画像 P_k とし、第 k の主階層画像 P_k に対する縮小処理によって得られる画像を第 $(k+1)$ の準備画像 $Q_{(k+1)}$ として、第 n の主階層画像 P_n が得られるまでフィルタ処理と縮小処理とを交互に

実行することにより、第1の主階層画像 P_1 ～第 n の主階層画像 P_n を含む複数 n 通りの階層画像からなる主画像ピラミッドを作成し、更に、第 k の主階層画像 P_k と第 k の準備画像 Q_k との差分画像 D_k を求め、当該差分画像 D_k を第 k の副階層画像 D_k とすることにより、第1の副階層画像 D_1 ～第 n の副階層画像 D_n を含む複数 n 通りの階層画像からなる副画像ピラミッドを作成するようにすればよい。

[0160] また、特徴量算出部123については、主画像ピラミッド PP および副画像ピラミッド PD を構成する各階層画像について、評価点 E の近傍の画素の画素値に基づいて特徴量を算出するようにすればよい。そうすれば、1つの評価点 E についての特徴量として、 $2n$ 次元ベクトルを抽出することが可能になり、より正確な推定演算を行うことが可能になる。

[0161] (3) 複数通りの元画像を作成して複数通りの画像ピラミッドを作成する変形例

上述した§2.4(2)では、同一の元画像に対して、複数通りのアルゴリズムを適用することにより、複数通りの画像ピラミッドを作成する変形例を述べた。ここでは、複数通りの元画像を作成して複数通りの画像ピラミッドを作成する変形例を述べる。

[0162] 先願発明に係る図形パターンの形状推定装置100'では、図1に示すように、元画像作成部121が、与えられた元図形パターン10に基づいて元画像を作成する処理を行う。ここで作成される元画像としては、§2.1で述べたとおり、面積密度マップ M_1 (図10)や、エッジ長密度マップ M_2 (図11)や、ドーズ密度マップ M_3 (図13)など、様々な形態の画像を採用することができる。

[0163] 別言すれば、元画像作成部121は、元図形パターン10に基づいて元画像を作成する際に、様々な作成アルゴリズムを採用することができ、いずれのアルゴリズムを採用したかによって、内容の異なる様々な元画像を作成することができる。たとえば、図10に示す面積密度マップ M_1 、図11に示すエッジ長密度マップ M_2 、図13に示すドーズ密度マップ M_3 は、いずれ

も同一の元図形パターン10に基づいて作成された画像であるが、個々の画素のもつ画素値は相互に異なっており、それぞれ異なる画像になっている。あるいは、同一の元図形パターン10に基づいて、画素サイズ（1画素の寸法）およびマップサイズ（縦横に並んだ画素の数）が異なる複数の密度マップ（要するに、解像度が異なる複数のマップ）を準備し、これら複数の密度マップのそれぞれを元画像として複数の画像ピラミッドを作成してもよい。もちろん、理論的には、画素サイズが小さく、マップサイズが大きな密度マップ（高解像度の密度マップ）を元画像として用いるのが理想的であるが、計算機のもつメモリは有限であるため、実用上は、画素サイズおよびマップサイズが異なる複数の密度マップを元画像として用いるのが好ましい。

[0164] そこで、元画像作成部121に、互いに異なる複数通りのアルゴリズムに基づく元画像作成処理を行い、複数通りの元画像を作成する機能をもたせておき、画像ピラミッド作成部122に、この複数通りの元画像に基づいてそれぞれ別個独立した画像ピラミッドを作成する処理を行う機能をもたせておけば、複数通りの画像ピラミッドを作成することができる。そして、特徴量算出部123に、この複数通りの画像ピラミッドのそれぞれを構成する各階層画像について、評価点Eの近傍の画素の画素値に基づいて特徴量を算出する機能をもたせておけば、より高次元のベクトルからなる特徴量を抽出することが可能になり、より正確な推定演算を行うことが可能になる。

[0165] たとえば、元画像作成部121によって、図10に示す面積密度マップM1からなる第1の元画像と、図11に示すエッジ長密度マップM2からなる第2の元画像と、図13に示すドーズ密度マップM3からなる第3の元画像と、を作成する機能をもたせておけば、画像ピラミッド作成部122は、この3通りの元画像に基づいて、3組の別個独立した画像ピラミッドを作成することができる。いずれの画像ピラミッドも、同一の元図形パターン10に基づいて作成された画像ピラミッドである。特徴量算出部123は、この3通りの画像ピラミッドのそれぞれを構成する各階層画像について、評価点Eの近傍の画素の画素値に基づいて特徴量を算出する処理を行うことができる。

。1つの画像ピラミッドから、 n 個の特徴量 $x_1 \sim x_n$ を抽出することになれば、同一の評価点 E について、合計 $3n$ 個の特徴量を抽出することが可能になる。すなわち、1つの評価点 E についての特徴量として、 $3n$ 次元ベクトルを与えることができるので、より正確な推定演算を行うことが可能になる。

[0166] もちろん、ここで述べた変形例に、上述した § 2. 4 (2) の変形例を組み合わせることも可能である。§ 2. 4 (2) で述べたとおり、画像ピラミッド作成部 122 が、画像ピラミッドを作成する際に、異なる2種類のアルゴリズムを採用すれば、主画像ピラミッド PP と副画像ピラミッド PD との2通りの画像ピラミッドを作成することができる。そこで、上記3通りの元画像のそれぞれに基づいて、主画像ピラミッド PP と副画像ピラミッド PD との2通りの画像ピラミッドを作成すれば、同一の元図形パターン 10 に基づいて合計6組の画像ピラミッドを作成することができるので、1つの評価点 E についての特徴量として、 $6n$ 次元ベクトルを与えることができる。

[0167] <<< § 3. バイアス推定ユニットの詳細 >>>

ここでは、バイアス推定ユニット 130 の詳細な処理動作を説明する。図1に示すように、バイアス推定ユニット 130 は、特徴量入力部 131 と推定演算部 132 を有しており、図4の流れ図におけるステップ S4 のプロセスバイアス推定処理を実行する機能を有している。図6に示す実施例の場合、1つの評価点 E について、 $x_1 \sim x_n$ なる特徴量 (n 次元ベクトル) が特徴量入力部 131 に入力され、推定演算部 132 による推定演算が実行され、評価点 E についてのプロセスバイアスの推定値 y が求められている。図6に示す実施例では、推定演算部 132 として、ニューラルネットワークが用いられている。そこで、以下、このニューラルネットワークの詳細な構成および動作を説明する。

[0168] <3. 1 ニューラルネットワークによる推定演算>

ニューラルネットワークは、近年、人工知能の根幹をなす技術として注目されており、画像処理をはじめとする様々な分野で利用されている。このニ

ューラルネットワークは、生物の脳の構造を模したコンピュータ上での構築物であり、ニューロンとそれを繋ぐエッジによって構成される。

[0169] 図24は、図1に示す推定演算部132として、ニューラルネットワークを利用した実施例を示すブロック図である。図示のとおり、ニューラルネットワークには、入力層、中間層（隠れ層）、出力層が定義され、入力層に与えられた情報に対して、中間層（隠れ層）において所定の情報処理がなされ、出力層にその結果が出力される。先願発明の場合、図示のとおり、ある1つの評価点Eについての特徴量 $x_1 \sim x_n$ が、 n 次元ベクトルとして入力層に与えられ、出力層には、当該評価点Eについてのプロセスバイアスの推定値 y が出力される。ここで、プロセスバイアスの推定値 y は、§1で述べたように、所定の図形の輪郭線上に位置する評価点Eについて、当該輪郭線の法線方向についてのずれ量を示す推定値である。

[0170] 図24に示す実施例の場合、推定演算部132は、特徴量入力部131が入力した特徴量 $x_1 \sim x_n$ を入力層とし、プロセスバイアスの推定値 y を出力層とするニューラルネットワークを有しており、このニューラルネットワークの中間層は、第1隠れ層、第2隠れ層、..., 第N隠れ層なるN層の隠れ層によって構成されている。これら隠れ層は、多数のニューロン（ノード）を有し、これら各ニューロンを繋ぐエッジが定義されている。

[0171] 入力層に与えられた特徴量 $x_1 \sim x_n$ は、エッジを介して各ニューロンに信号として伝達されてゆく。そして、最終的に、出力層からプロセスバイアスの推定値 y に相当する信号が出力されることになる。ニューラルネットワーク内の信号は、1つの隠れ層のニューロンから次の隠れ層のニューロンへと、エッジを介した演算を経て伝達されてゆく。エッジを介した演算は、学習段階で得られた学習情報L（具体的には、後述するパラメータ W , b ）を用いて行われる。

[0172] 図25は、図24に示すニューラルネットワークで実行される具体的な演算プロセスを示すダイアグラムである。図に太線で示す部分は、第1隠れ層, 第2隠れ層, ..., 第N隠れ層を示しており、各隠れ層内の個々の円はニュー

ーロン（ノード）、各円を連結する線はエッジを示している。前述したとおり、入力層には、ある1つの評価点Eについての特徴量 $x_1 \sim x_n$ が、 n 次元ベクトルとして与えられ、出力層には、当該評価点Eについてのプロセスバイアスの推定値 y がスカラー値（輪郭線の法線方向についてのずれ量を示す寸法値）として出力される。

[0173] 図示の例の場合、第1隠れ層は $M(1)$ 次元の層であり、合計 $M(1)$ 個のニューロン $h(1, 1) \sim h(1, M(1))$ によって構成され、第2隠れ層は $M(2)$ 次元の層であり、合計 $M(2)$ 個のニューロン $h(2, 1) \sim h(2, M(2))$ によって構成され、第 N 隠れ層は $M(N)$ 次元の層であり、合計 $M(N)$ 個のニューロン $h(N, 1) \sim h(N, M(N))$ によって構成されている。

[0174] ここで、第1隠れ層のニューロン $h(1, 1) \sim h(1, M(1))$ に伝達される信号の演算値を、同じ符号を用いて、それぞれ演算値 $h(1, 1) \sim h(1, M(1))$ と表すことにすると、これら演算値 $h(1, 1) \sim h(1, M(1))$ の値は、図26の上段に示す行列の式で与えられる。この式の右辺の関数 $f(\xi)$ としては、図27(a)に示すシグモイド関数、図27(b)に示す正規化線形関数ReLU、図27(c)に示す正規化線形関数LeakyReLUなどの活性化関数を用いることができる。

[0175] また、関数 $f(\xi)$ の引数として記載されている ξ は、図26の中段に示すように、行列 $[W]$ と行列 $[x_1 \sim x_n]$ （入力層に n 次元ベクトルとして与えられた特徴量）との積に、行列 $[b]$ を加えた値になる。ここで、行列 $[W]$ および行列 $[b]$ の内容は、図26の下段に示すとおりであり、行列の個々の成分（重みパラメータ $W(u, v)$ とバイアスパラメータ $b(u, v)$ ）は、後述する学習段階によって得られた学習情報 L である。すなわち、行列 $[W]$ および行列 $[b]$ を構成する個々の成分（パラメータ $W(u, v)$ 、 $b(u, v)$ ）の値は、学習情報 L として与えられており、図26に示された演算式を用いれば、入力層に与えられた特徴量 $x_1 \sim x_n$ に基づいて、第1隠れ層の演算値 $h(1, 1) \sim h(1, M(1))$ を算出するこ

とができる。

[0176] 一方、図28は、図25に示すダイアグラムにおける第2隠れ層～第N隠れ層の各値を求める演算式を示す図である。具体的には、第 $(i+1)$ 隠れ層 $(1 \leq i \leq N)$ のニューロン $h(i+1, 1) \sim h(i+1, M(i+1))$ に伝達される信号の演算値を、同じ符号を用いて、それぞれ演算値 $h(i+1, 1) \sim h(i+1, M(i+1))$ と表すことにすると、これら演算値 $h(i+1, 1) \sim h(i+1, M(i+1))$ の値は、図28の上段に示す行列の式で与えられる。この式の右辺の関数 $f(\xi)$ としては、前述したように、図27に示す各関数などを用いることができる。

[0177] また、関数 $f(\xi)$ の引数として記載されている ξ は、図28の中段に示すように、行列 $[W]$ と行列 $[h(i, 1) \sim h(i, M(i))]$ （1つ前の第 i 隠れ層のニューロン $h(i, 1) \sim h(i, M(i))$ の演算値）との積に、行列 $[b]$ を加えた値になる。ここで、行列 $[W]$ および行列 $[b]$ の内容は、図28の下段に示すとおりであり、行列の個々の成分（パラメータ $W(u, v)$ 、 $b(u, v)$ ）は、やはり、後述する学習段階によって得られた学習情報 L である。

[0178] ここでも、行列 $[W]$ および行列 $[b]$ を構成する個々の成分（パラメータ $W(u, v)$ 、 $b(u, v)$ ）の値は、学習情報 L として与えられており、図28に示された演算式を用いれば、第 i 隠れ層で求められた演算値 $[h(i, 1) \sim h(i, M(i))]$ に基づいて、第 $(i+1)$ 隠れ層の演算値 $h(i+1, 1) \sim h(i+1, M(i+1))$ を算出することができる。よって、図25に示すダイアグラムにおける第2隠れ層～第N隠れ層の各値は、図28に示す演算式に基づいて順次求めることができる。

[0179] 図29は、図25に示すダイアグラムにおける出力層の値 y を求める演算式を示す図である。具体的には、出力値 y （評価点 E についてのプロセスバイアスの推定値：スカラー値）は、図29の上段に示す行列の式で与えられる。すなわち、出力値 y は、行列 $[W]$ と行列 $[h(N, 1) \sim h(N, M(N))]$ （第N隠れ層のニューロン $h(N, 1) \sim h(N, M(N))$ の

値)との積に、スカラー値 $b(N+1)$ を加えた値になる。ここで、行列 $[W]$ の内容は、図 29 の下段に示すとおりであり、行列 $[W]$ の個々の成分 (パラメータ $W(u, v)$) およびスカラー値 $b(N+1)$ は、やはり、後述する学習段階によって得られた学習情報 L である。

[0180] このように、図 25 に示すダイアグラムにおける第 1 隠れ層の各値は、入力層として与えられた特徴量 $x_1 \sim x_n$ に、学習情報 L として予め準備されているパラメータ $W(1, v)$, $b(1, v)$ を作用させることにより求めることができ、第 2 隠れ層の各値は、第 1 隠れ層の各値に、学習情報 L として予め準備されているパラメータ $W(2, v)$, $b(2, v)$ を作用させることにより求めることができ、...、第 N 隠れ層の各値は、第 $(N-1)$ 隠れ層の各値に、学習情報 L として予め準備されているパラメータ $W(N, v)$, $b(N, v)$ を作用させることにより求めることができ、出力層 y の値は、第 N 隠れ層の各値に、学習情報 L として予め準備されているパラメータ $W(N+1, v)$, $b(N+1)$ を作用させることにより求めることができる。具体的な演算式は、図 26～図 29 に示すとおりである。

[0181] なお、§ 2.4 の(2) , (3) で述べたように、複数通りの画像ピラミッドを作成する変形例を採用する場合は、入力層として与えられる特徴量が、 n 次元のベクトル ($x_1 \sim x_n$) ではなく、 $V \cdot n$ 次元 (V は、画像ピラミッドの総数) のベクトルになるが、図 25 に示すダイアグラムにおける入力層の数値が $V \cdot n$ 個に増加するだけで、ニューラルネットワークの基本的な構成および動作に変わりはない。

[0182] 以上、図 25 に示すニューラルネットワークを用いて、ある 1 つの評価点 E についてのプロセスバイアスの推定値 y を求める演算を説明したが、実際には、元図形パターン 10 に含まれる図形の輪郭線上に定義された多数の評価点について同様の演算が行われ、個々の評価点について、それぞれプロセスバイアスの推定値 y が求められることになる。図 1 に示す図形パターンの形状補正装置 100 では、パターン補正ユニット 140 が、こうして求められた個々の評価点についてのプロセスバイアスの推定値 y に基づいてパター

ン形状の補正処理（図４のステップＳ５）を実行することになる。

[0183] <３．２ ニューラルネットワークの学習段階>

上述したとおり、図２４に示す実施例の場合、推定演算部１３２は、ニューラルネットワークによって構成されており、予め設定されている学習情報Ｌを利用して、各ニューロンに伝達される信号値を演算することになる。ここで、学習情報Ｌの実体は、図２６の下段、図２８の下段、図２９の下段に行列[W]，[b]の各成分として記載されたパラメータ $W(u, v)$ ， $b(u, v)$ 等の値である。したがって、このようなニューラルネットワークを構築するには、予め実行した学習段階によって、学習情報Ｌを得ておく必要がある。

[0184] すなわち、推定演算部１３２に含まれるニューラルネットワークは、多数のテストパターン図形を用いたリソグラフィプロセスによって実基板Ｓ上に実際に形成される実図形パターン２０の実寸法測定によって得られた寸法値と、各テストパターン図形から得られる特徴量と、を用いた学習段階によって得られたパラメータ $W(u, v)$ ， $b(u, v)$ 等を学習情報Ｌとして用い、プロセスバイアスの推定処理を行うことになる。このようなニューラルネットワークの学習段階の処理自体は公知の技術であるが、ここでは先願発明に利用するニューラルネットワークの学習段階に適した処理の概要を簡単に説明しておく。

[0185] 図３０は、図２４に示すニューラルネットワークが用いる学習情報Ｌを得るための学習段階の手順を示す流れ図である。まず、ステップＳ８１において、テストパターン図形の作成処理が実行される。ここで、テストパターン図形は、たとえば、図２(a)に示すような元図形パターン１０に相当するものであり、通常、長方形やＬ字型図形などの単純な図形が用いられる。実際には、サイズや形状が異なる数千個ものテストパターン図形が作成される。

[0186] 続いて、ステップＳ８２において、各テストパターン図形上に評価点Ｅが設定される。具体的には、個々のテストパターン図形の輪郭線上に、所定間隔で多数の評価点Ｅを定義する処理を行えばよい。そして、ステップＳ８３

において、各評価点Eについて、それぞれ特徴量が抽出される。このステップS 8 3の特徴量抽出処理は、§ 2で述べた処理と同様であり、特徴量抽出ユニット1 2 0と同等の機能を有するユニットによって実行される。§ 2で述べた手順によれば、個々の評価点Eについて、それぞれ特徴量 $x_1 \sim x_n$ が抽出されることになる。

[0187] ステップS 8 4の推定演算部学習処理は、ステップS 8 3で抽出された特徴量を利用して、学習情報L（すなわち、パラメータ $W(u, v)$, $b(u, v)$ 等）を決定する処理である。この学習処理を実行するためには、実際のリソグラフィプロセスにより得られた実寸法が必要になる。そこで、ステップS 8 5では、ステップS 8 1で作成されたテストパターン図形に基づいて、実際にリソグラフィプロセスが実行され、実基板Sが作成される。そして、ステップS 8 6において、この実基板S上に形成された実図形パターンに対して、個々の図形の実寸法測定が行われる。この測定結果は、ステップS 8 4の学習処理に利用されることになる。

[0188] このように、図30に示す学習段階のプロセスは、ステップS 8 1～S 8 4からなる計算機上で実行するプロセス（コンピュータプログラムにより実行されるプロセス）と、ステップS 8 5, S 8 6からなる実基板上で実行するプロセスと、によって構成される。ステップS 8 4の推定演算部学習処理は、計算機上の処理で得られた特徴量 $x_1 \sim x_n$ と実基板上で測定された実寸法とに基づいて、ニューラルネットワークに利用される学習情報Lを決定する処理ということになる。

[0189] 図31は、図30に示す流れ図におけるステップS 8 4の推定演算部学習の詳細な手順を示す流れ図である。まず、ステップS 8 4 1において、各評価点の設計位置および特徴量が入力される。ここで、評価点の設計位置は、ステップS 8 2において設定された評価点のテストパターン図形上の位置であり、評価点の特徴量は、ステップS 8 3において抽出された特徴量 $x_1 \sim x_n$ である。続いて、ステップS 8 4 2において、各評価点の実位置が入力される。ここで、評価点の実位置は、ステップS 8 6において実測された実

基板S上の各図形の実寸法に基づいて決定される。そして、ステップS843において、各評価点の実バイアスが算出される。この実バイアスは、ステップS841で入力された評価点の設計位置とステップS842で入力された評価点の実位置とのずれ量に相当する。

[0190] たとえば、ステップS81において、図3(a)に示す元図形パターン10のような長方形がテストパターン図形として作成された場合、ステップS82において、この長方形の輪郭線上に、評価点E11, E12, E13等が設定され、ステップS83において、これら各評価点E11, E12, E13のそれぞれについて、特徴量 $x_1 \sim x_n$ が抽出される。各評価点E11, E12, E13の位置および特徴量は、ステップS841において入力される。

[0191] 一方、ステップS85のリソグラフィプロセスによって、実基板S上には、たとえば、図3(b)に示すような実図形パターン20が形成され、ステップS86の実寸法測定によって、この実図形パターン20を構成する長方形の各辺について、実寸法が測定される。この測定により、各評価点E11, E12, E13を、それぞれ輪郭線の法線方向に移動させた点として、評価点の実位置E21, E22, E23が決定される。各評価点の実位置E21, E22, E23は、ステップS842において入力される。

[0192] そこで、ステップS843では、図3(b)に示すように、各評価点E11, E12, E13の設計位置と実位置E21, E22, E23との差として、実バイアス y_{11}, y_{12}, y_{13} が算出される。実際には、たとえば、図3(b)に示す実図形パターン20を構成する長方形の横幅 b と、図3(a)に示す元図形パターン10を構成する長方形の横幅 a との差「 $b - a$ 」を2で除した値 y を求める作業により、実バイアスを決定するようにしてもよい。もちろん、実際には、数千個という規模のテストパターン図形が作成され、個々の図形の輪郭線上に多数の評価点が設定されるので、ステップS841～S843の手順は、膨大な数の評価点についてそれぞれ実行されることになる。ここで、1つの評価点Eについて着目すると、当該評価点Eについ

ては、特徴量 $x_1 \sim x_n$ と実バイアス y との組み合わせが、学習材料として準備されたことになる。

[0193] 続いて、ステップ S 8 4 4 において、パラメータ W , b が初期値に設定される。ここで、パラメータ W , b は、図 2 6 の下段、図 2 8 の下段、図 2 9 の下段に行列 $[W]$, $[b]$ の各成分として記載されたパラメータ $W(u, v)$, $b(u, v)$ 等に相当し、学習情報 L を構成する値になる。初期値としては、乱数によってランダムな値を与えればよい。別言すれば、学習前の当初の段階では、学習情報 L を構成するパラメータ W , b はデタラメな値になっている。

[0194] 続いて実行されるステップ S 8 4 5 ~ S 8 4 8 の各段階は、実際の学習プロセスであり、この学習プロセスを行うことにより、当初はデタラメな値が設定されていたパラメータ W , b が徐々に更新されてゆき、やがて学習情報 L として十分に機能する程度の正確な値に修正される。まず、ステップ S 8 4 5 では、特徴量 $x_1 \sim x_n$ からプロセスバイアス y を推定する演算が実行される。

[0195] 具体的には、図 2 4 に示すようなニューラルネットワークを準備する。もちろん、この段階では、学習情報 L を構成するパラメータ W , b には、初期値として乱数が与えられており、このニューラルネットワークは、推定演算部 1 3 2 としての正常な機能を果たすことはできない不完全なものである。この不完全なニューラルネットワークの入力層に、ステップ S 8 4 1 で入力した特徴量 $x_1 \sim x_n$ を与え、不完全な値からなる学習情報 L を用いた演算を行い、出力層としてプロセスバイアスの推定値 y を算出する。もちろん、当初得られるプロセスバイアスの推定値 y は、観測された実バイアスとはかけ離れた値になる。

[0196] そこで、ステップ S 8 4 6 において、その時点における実バイアスに対する残差を算出する。すなわち、ステップ S 8 4 5 で得られたプロセスバイアスの推定値 y と、ステップ S 8 4 3 で得られた実バイアスの算出値との差を求め、この差を、当該評価点 E についての残差とする。この残差が所定の許

容値以下になれば、その時点での学習情報 L （すなわち、パラメータ $W(u, v)$, $b(u, v)$ 等）は、十分に実用性をもった学習情報であると判断できるので、学習段階を終了することができる。

[0197] ステップS847は、学習段階を終了できるか否かを判定する手順である。実際には、膨大な数の評価点のそれぞれについて残差が求められるので、実用上は、たとえば、残差二乗和の改善量が規定値を下回っていたら、学習終了とする判断を行うような判定方法を採用すればよい。学習終了と判定されなかった場合は、ステップS848において、パラメータ $W(u, v)$, $b(u, v)$ 等の更新が行われる。具体的には、残差を減少させる作用が生じるように、各パラメータ $W(u, v)$, $b(u, v)$ 等の値を所定量だけ増減する更新が行われる。具体的な更新方法については、ニューラルネットワークにおける学習手法として様々なアルゴリズムに基づく方法が知られているので、ここでは説明を省略するが、基本方針としては、多数のテストパターン図形の評価点についての残差を総合的に勘案して、それぞれの残差が全体的に少なくなるような総合的な調整を行うアルゴリズムが採用される。

[0198] こうして、ステップS847において肯定的な判定がなされるまで、ステップS845～S848の手順が繰り返し実行される。その結果、学習情報 L を構成するパラメータ $W(u, v)$, $b(u, v)$ 等の値は、残差を減少させる方向に徐々に修正されてゆき、最終的には、ステップS847において肯定的な判定がなされ、学習段階は終了する。この学習終了段階で得られた学習情報 L （パラメータ $W(u, v)$, $b(u, v)$ 等）は、実バイアスに近いプロセスバイアスの推定値 y を出力層に得るのに適した情報になっている。したがって、学習終了段階で得られた学習情報 L を含むニューラルネットワークは、先願発明における推定演算部132として機能することになる。

[0199] <<< § 4. 先願発明に係る図形パターンの形状推定方法 >>>

これまで先願発明を、図1に示す構成を有する図形パターンの形状推定装置100'もしくは図形パターンの形状補正装置100として捉え、その構

成および動作を説明した。ここでは、先願発明を、図形パターンの形状推定方法という方法発明として捉えた説明を簡単に行っておく。

[0200] 先願発明を図形パターンの形状推定方法の発明として把握した場合、当該方法は、元図形パターンを用いたリソグラフィプロセスをシミュレートすることにより、実基板上に形成される実図形パターンの形状を推定する方法ということになる。そして、この方法は、コンピュータが、図形の内部と外部との境界を示す輪郭線の情報を含む元図形パターン10を入力する元図形パターン入力段階（図4のステップS1で作成されたパターンを入力する段階）と、コンピュータが、入力した図形の輪郭線上の所定位置に評価点Eを設定する評価点設定段階（図4のステップS2）と、コンピュータが、入力した元図形パターン10について、各評価点Eの周囲の特徴を示す特徴量 $x_1 \sim x_n$ を抽出する特徴量抽出段階（図4のステップS3）と、コンピュータが、抽出した特徴量 $x_1 \sim x_n$ に基づいて、各評価点Eの元図形パターン10上の位置と実図形パターン20上の位置とのずれ量を示すプロセスバイアス y を推定するプロセスバイアス推定段階（図4のステップS4）と、によって構成される。

[0201] ここで、特徴量抽出段階（図4のステップS3）は、元図形パターン10に基づいて、それぞれ所定の画素値を有する画素Uの集合体からなる元画像Q1を作成する元画像作成段階（図7のステップS32）と、この元画像Q1に基づいて、縮小画像Qk（準備画像）を作成する縮小処理（図7のステップS35）を含む画像ピラミッド作成処理を行い、それぞれ異なるサイズをもった複数の階層画像P1～Pnからなる画像ピラミッドPPを作成する画像ピラミッド作成段階（図7のステップS33～S35）と、作成された画像ピラミッドPPを構成する各階層画像P1～Pnについて、各評価点Eの近傍の画素の画素値に基づいて特徴量 $x_1 \sim x_n$ を算出する特徴量算出段階（図7のステップS37）と、を含んでいる。

[0202] ここで、上記プロセスバイアス推定段階（図4のステップS4）は、予め実施された学習段階によって得られた学習情報Lに基づいて、評価点Eにつ

いての特微量 $x_1 \sim x_n$ に応じた推定値を求め、求めた推定値を当該評価点 E についてのプロセスバイアスの推定値 y として出力する推定演算段階を含んでいる。

[0203] なお、§ 2 で述べた特微量抽出ユニット 120 を使用した場合は、上記画像ピラミッド作成段階で、元画像 Q_1 もしくは縮小画像 Q_k に対して所定の画像処理フィルタを用いたフィルタ処理を行うフィルタ処理段階（図 7 のステップ S33）と、このフィルタ処理後の画像 P_k に対して縮小処理を行う縮小処理段階（図 7 のステップ S35）と、を交互に実行することにより、複数の階層画像 $P_1 \sim P_n$ からなる画像ピラミッド PP を作成することができる。

[0204] 具体的には、§ 2. 2 で述べた手順では、画像ピラミッド作成段階で、フィルタ処理後の画像（図 7 のステップ S33 で得られたフィルタ処理画像 P_k ）を階層画像 $P_1 \sim P_n$ とする画像ピラミッド PP が作成されている。これに対して、§ 2. 4 (1) で述べた変形例の手順では、画像ピラミッド作成段階で、フィルタ処理後の画像（フィルタ処理画像 P_k ）とフィルタ処理前の画像（準備画像 Q_k ）との差分画像 D_k が作成され、作成した差分画像を階層画像 $D_1 \sim D_n$ とする画像ピラミッド PD が作成されている。

[0205] <<< § 5. 本発明に係る形状補正装置の基本構成 >>>

さて、これまで § 1 ~ § 4 において、先願発明に係る図形パターンの形状補正装置の説明を行った。前述したとおり、本発明に係る技術は、この先願発明に係る図形パターンの形状補正装置に適用するために開発されたものである。もっとも、本発明は、先願発明への適用に限定されるものではない。本発明の特徴は、リソグラフィプロセスのシミュレーションを行う際に、ユニークな方法で評価点の設定を行い、この評価点に基づくシミュレーションの結果に基づいて元図形パターンに対する形状補正を行う点にある。したがって、本発明は、従来から知られている一般的な図形パターンの形状補正装置に広く適用可能である。ただ、この § 5 以降では、説明の便宜上、§ 1 ~ § 4 で述べた先願発明に係る図形パターンの形状補正装置に本発明を適用し

た具体的な実施例を述べることにする。

[0206] ここで述べる実施例に係る図形パターンの形状補正装置の基本構成は、§ 1～§ 4で述べた先願発明に係る装置と同じである。すなわち、ここで述べる形状補正装置100は、図1に示すとおり、評価点設定ユニット110、特徴量抽出ユニット120、バイアス推定ユニット130、パターン補正ユニット140を備えており、元図形パターン10に基づくリソグラフィプロセスによって実基板S上に実図形パターン20を形成する際に、実図形パターン20が元図形パターン10に一致するように、元図形パターン10の形状を補正して、リソグラフィプロセスで実際に用いる補正図形パターン15を作成する機能を有している。

[0207] 実際には、上記図形パターンの形状補正装置100としての機能は、汎用のコンピュータに専用のプログラムを組み込むことによって実現される。したがって、上述した評価点設定ユニット110、特徴量抽出ユニット120、バイアス推定ユニット130、パターン補正ユニット140は、それぞれコンピュータに所定のプログラムを組み込むことにより構成される。

[0208] ここで、評価点設定ユニット110は、§ 1で述べたように、元図形パターン10を構成する個々の図形（以下、単位図形と呼び、元図形パターンと同じ符号10を付して示す）について、内部と外部との境界を示す輪郭線上の所定位置に評価点Eを設定する役割を果たす。図3(a)には、矩形状の単位図形10上に3つの評価点E11, E12, E13が設定された例が示されているが、実際には、単位図形10の輪郭線上を一回りする経路上に多数の評価点が設定される。本発明の特徴は、この評価点の設定手法にある。具体的な評価点設定手法については、§ 6で詳述する。

[0209] 特徴量抽出ユニット120は、元図形パターン10について、個々の評価点Eの周囲の特徴を示す特徴量 $x_1 \sim x_n$ を抽出する役割を果たす。§ 2では、先願発明に固有の特徴量抽出方法として、画像ピラミッドPP, PDを用いる手順を示した。もちろん、本発明を実施する上でも、画像ピラミッドPP, PDを用いて特徴量を抽出する方法を採用することができる。ただ、

本発明に用いる特徴量抽出ユニット120は、§2で述べた方法で特徴量を抽出するものに限定されるものではない。本発明を実施するにあたって、特徴量抽出ユニット120として、任意の方法で特徴量を抽出するユニットを採用してかまわない。たとえば、複数 n 通りの階層画像 $P_1 \sim P_n$ からなる画像ピラミッド PP を用いる代わりに、複数 n 通りの算出関数を用いて各評価点 E についての n 通りの特徴量 $x_1 \sim x_n$ を算出する処理を行うことも可能である。各算出関数としては、座標値 X, Y に位置する評価点 $E(X, Y)$ について、当該座標値 X, Y を変数として代入することにより、評価点 $E(X, Y)$ と元図形パターン10に含まれる各図形との位置関係を定量化して所定の関数値を算出することができる関数（たとえば、先願の付加的实施形態に用いられている関数）を用いればよい。

[0210] バイアス推定ユニット130は、特徴量抽出ユニット120が抽出した各評価点についての特徴量 $x_1 \sim x_n$ に基づいて、各評価点の元図形パターン10上の位置と実図形パターン20上の位置とのずれ量を示すプロセスバイアス y を推定する役割を果たす。§3では、ニューラルネットワークによる推定演算により、プロセスバイアス y を推定する手順を示した。もちろん、本発明を実施する上でも、同様の手順を採用することができる。ただ、本発明に用いるバイアス推定ユニット130は、§3で述べた方法でプロセスバイアス y を推定するものに限定されるものではない。本発明を実施するにあたって、バイアス推定ユニット130として、任意の方法でプロセスバイアス y を推定するユニットを採用してかまわない。

[0211] パターン補正ユニット140は、バイアス推定ユニット130から出力されるプロセスバイアスの推定値 y に基づいて、元図形パターン10に対する補正を行うことにより、補正図形パターン15を作成する役割を果たす。このパターン補正ユニット140による補正処理は、評価点設定ユニット110によって設定された個々の評価点 E の位置と当該評価点 E について推定されたプロセスバイアスの推定値 y とを考慮して実行されることになる。そこで、本発明に係る具体的な補正処理については、§7で詳述する。

[0212] 結局、本発明に係る形状補正装置１００は、図１に示す先願発明に係る形状補正装置１００と同様に、評価点設定ユニット１１０、特徴量抽出ユニット１２０、バイアス推定ユニット１３０、パターン補正ユニット１４０なる構成要素を備えており、これら各構成要素の基本機能は、先願発明に係る装置と同様である。ただ、本発明の特徴は、評価点設定ユニット１１０およびパターン補正ユニット１４０に、固有の処理機能を採用した点にある。以下、この点について詳述する。

[0213] <<< § 6. 評価点設定ユニットの詳細 >>>

ここでは、本発明に係る形状補正装置１００における評価点設定ユニット１１０の詳細な構成およびその動作について説明する。図３２は、本発明の基本的実施形態に係る図形パターンの形状補正装置１００（図１参照）内の評価点設定ユニット１１０の構成を示すブロック図である。この評価点設定ユニット１１０の基本機能は、元図形パターン１０を構成する個々の単位図形について、内部と外部との境界を示す輪郭線上の所定位置に評価点Ｅを設定することにある。

[0214] そのような基本機能を実現するために、図３２に示す評価点設定ユニット１１０は、単位図形入力部１１１、単位図形分割部１１２、分割図形ベクトル定義部１１３、単位図形ベクトル定義部１１４、通常端点定義部１１５、端点再構築部１１６、評価点定義部１１７、補充端点定義部１１８を備えている。これらの各要素は、実際には、コンピュータに所定のプログラムを組み込むことにより構成される。なお、端点再構築部１１６は本発明にとって必須の構成要素ではないので、本発明を実施するにあたり、端点再構築部１１６は省略することも可能である。ただ、実用上は、端点再構築部１１６を設けた方が、より効率的な評価点設定を行うことが可能である。したがって、以下、端点再構築部１１６を設けた好適な実施例を説明する。

[0215] 図示のとおり、ここに示す評価点設定ユニット１１０には、入力として、元図形パターン１０とビーム最大成型サイズ B_{max} が与えられている。ここで、元図形パターン１０は、図１に示すように、リソグラフィプロセスの元

になる図形パターンであり、通常、多数の単位図形の集合体によって構成される。一方、ビーム最大成型サイズ $B_{\max}$ は、リソグラフィプロセスで用いる描画装置のビーム最大成型サイズであり、描画装置によって照射される光ビームや電子ビームの1回のショットによる最大照射範囲の寸法に相当する。

[0216] 一方、この評価点設定ユニット110からは、第1の端点T1と第2の端点T2とに挟まれた輪郭線分Oの情報が評価点Eの情報とともに出力される。各端点T1, T2は、元図形パターン10を構成する個々の単位図形の輪郭線上に定義された点であり、輪郭線分Oの両端を画定するための点である。また、ここに示す実施例の場合、評価点Eは、輪郭線分Oの midpoint に定義された点である。図32には、1組の輪郭線分Oしか描かれていないが、実際には、個々の単位図形の輪郭線上に多数の輪郭線分が設定され、それぞれの輪郭線分について評価点が設定されることになる。以下、このような出力を得るための具体的な手順を説明する。

[0217] <6. 1 単位図形の分割>

図32に示す単位図形入力部111は、元図形パターン10を構成する個々の単位図形を入力する構成要素である。ここで述べる実施形態の場合、個々の単位図形は多角形によって構成されており、図2(a)には、元図形パターン10として矩形が入力された例が示されている。半導体デバイスなどを製造するために用いられる実際の元図形パターン10は、様々な形状をもった微細な単位図形を多数集めた複雑な図形集合体によって構成されることになるが、通常、個々の単位図形は多角形によって構成されている。

[0218] 図32に示す単位図形分割部112は、単位図形入力部111が入力した個々の単位図形を直線状の分割線で分割して複数の分割図形を形成する処理を実行する。上述したとおり、個々の単位図形は多角形であるので、これを直線状の分割線で分割して得られる個々の分割図形も多角形になる。図33は、単位図形分割部112によって行われた具体的な分割処理の例を示す平面図である。ここでは説明の便宜上、1つの単位図形からなる元図形パター

ン 10 が入力された場合を例にとり、当該単位図形についても元図形パターンと同様に符号「10」を付して示すことにする。

- [0219] 図 33(a) は、寸法 W_{split} の幅をもった細長い帯状単位図形 10 に対して、単位図形分割部 112 による分割が行われた例を示す平面図である。具体的には、単位図形入力部 111 が、所定の長手方向軸（図示の例の場合 X 軸）に沿って伸び、互いに平行な直線状の上辺および下辺を有する細長い帯状単位図形 10 を入力したという想定で、単位図形分割部 112 が、この帯状単位図形 10 を、長手方向軸（X 軸）に沿って所定間隔で配置された複数の分割線（図では破線で示す）によって分割することにより、当該長手方向軸（X 軸）に沿って一次的に並んだ複数の分割図形 10a, 10b, 10c, …… , 10j, 10k, 10l が形成されている（後述するように、各分割図形の符号 a, b, c, …… の並び順は、左右の端から交互に中央へ向かう順になっている）。
- [0220] 図示の例の場合、左端位置と右端位置のそれぞれから、帯状単位図形 10 の幅と同じ寸法 W_{split} の間隔で破線で示す分割線が配置されている。具体的には、まず、一点鎖線で示す左端位置から右方へ W_{split} の位置に分割線を配置して分割図形 10a を画定し、次に、一点鎖線で示す右端位置から左方へ W_{split} の位置に分割線を配置して分割図形 10b を画定し、続いて、分割図形 10a の右の分割線から右方へ W_{split} の位置に分割線を配置して分割図形 10c を画定し、……というように、左右の端から交互に中央に向かって分割線を配置してゆく。そして、中央部に残された残余部分の幅 W_{rest} が、 $W_{rest} \leq 2 \times W_{split}$ になったら、この残余部分を 2 分割する分割線を配置して、分割図形 10k, 10l を画定する。このため、分割図形 10a ~ 10j は一辺が W_{split} の正方形になり、中央の分割図形 10k, 10l は横幅が端数の寸法をもつ長方形になる。
- [0221] 半導体デバイスなどには、図 33(a) に示すような細長い帯状単位図形 10 が、配線層などを構成するパターンとして多用されている。もちろん、分割線の配置間隔は、必ずしも帯状単位図形 10 の縦幅 W_{split} に一致させる

必要はなく、任意の寸法に設定してもかまわない。この場合、各分割図形は長方形になる。なお、図 3 3 (a) に示す例のように、所定間隔 W_{split} の幅で左右両端から交互に分割を行ってゆき、最後に、 $W_{rest}/2$ の幅をもつ分割図形 1 0 k, 1 0 l が中央に配置されるようにするのは、単位図形 1 0 の左右両端の直近に分割線が配置されることを避けるためである。後述するように、評価点は各分割図形の辺の中点位置に設定されるため、図示されている両端の分割図形 1 0 a, 1 0 b の横幅が小さくなると、評価点が帯状単位図形 1 0 の頂点直近に設定されることになり好ましくない。図 3 3 (a) に示す方法で分割すれば、両端の分割図形 1 0 a, 1 0 b の横幅を W_{split} にすることができ、頂点から距離 $W_{split}/2$ だけ離れた位置（頂点の直近ではなく、ある程度離れた位置）に評価点を設定することができる。

[0222] 図 3 3 (b) は、より広い縦幅をもった矩形状の単位図形 1 0 に対して、単位図形分割部 1 1 2 による分割が行われた例を示す平面図である。このように、単位図形入力部 1 1 1 が、XY 平面上に広がる単位図形 1 0 を入力したときには、単位図形分割部 1 1 2 は、まず、X 軸に平行な水平分割線を Y 軸方向に所定間隔をあけて複数配置し、これら複数の水平分割線によって単位図形 1 0 を分割することにより、X 軸方向に伸びる細長い複数の帯状図形を形成する。

[0223] 図 3 3 (b) に示す例では、破線で示す水平分割線が、上端位置と下端位置のそれぞれから、所定寸法 W_{split} を基準とする間隔で配置されている。具体的には、まず、一点鎖線で示す上端位置から下方に W_{split} の位置に水平分割線を配置して分割図形 1 0 a を画定し、次に、一点鎖線で示す下端位置から上方に W_{split} の位置に水平分割線を配置して分割図形 1 0 b を画定し、続いて、分割図形 1 0 a の下の水平分割線から下方へ W_{split} の位置に水平分割線を配置して分割図形 1 0 c を画定し、……というように、上下の端から交互に中央に向かって水平分割線を配置してゆく。そして、中央部に残された残余部分の縦幅 W_{rest} が、 $W_{rest} \leq 2 \times W_{split}$ になったら、この残余部分を 2 分割する水平分割線を配置して、分割図形 1 0 d, 1 0 e を画

定する。このため、縦幅 W_{split} をもつ細長い帯状図形10a, 10b, 10cと、縦幅が端数の寸法をもつ細長い帯状図形10d, 10eが形成されている。このように、所定間隔 W_{split} の幅で上下両端から交互に分割を行ってゆき、最後に、 $W_{rest}/2$ の幅をもつ分割図形10d, 10eが中央に配置されるようにするのは、やはり評価点が単位図形10の頂点直近に設定されることを避けるためである。

[0224] こうして、元の矩形状の単位図形10が、複数の帯状図形10a, 10b, 10c, 10d, 10eに分割されたら、更に、これら複数の帯状図形10a, 10b, 10c, 10d, 10eのそれぞれを、X軸に沿って所定間隔で配置されY軸に平行な複数の垂直分割線によって分割することにより、二次元的に並んだ複数の分割図形（図示省略）を形成することができる。すなわち、各帯状図形10a, 10b, 10c, 10d, 10eは、いずれも図33(a)に示すようなX軸方向に伸びる細長い図形になるので、図33(a)に示す例と同様に、それぞれ横方向に分割すれば、二次元的に並んだ複数の分割図形を得ることができる。たとえば、X軸に沿って W_{split} の間隔で配置された垂直分割線によって分割すれば、一辺が W_{split} の多数の正方形を含む分割図形群を得ることができる。

[0225] 図33(c)は、幅 W_{split} を有するL字型の帯からなる単位図形10に対して、単位図形分割部112による分割が行われた例を示す平面図である。この例についても、図33(b)と同じ手法を適用すれば、まず、X軸に平行な水平分割線をY軸に沿って W_{split} の間隔で配置し、これら水平分割線による分割が行われる。図33(c)は、このような分割が行われた状態を示しており、4組の正方形10a, 10b, 10c, 10dと、1組の細長い帯状図形10eが形成されている。そこで、更に、細長い帯状図形10eを、X軸に沿って所定間隔（たとえば、 W_{split} ）で配置されY軸に平行な複数の垂直分割線によって分割すれば、図33(a)と同様に、帯状図形10eを複数の分割図形（図示省略）に分割することができる。たとえば、X軸に沿って W_{split} の間隔で配置された垂直分割線によって帯状図形10eを分割

すれば、一辺が W_{split} の正方形を X 軸方向に並べた分割図形群を得ることができる。

[0226] 以上、長方形もしくは長方形の組み合わせからなる形状を有する単位図形についての分割例を示したが、元図形パターンには、斜めの辺をもった単位図形が含まれる場合もある。図34は、図32に示す単位図形分割部112によって、平行四辺形型の単位図形10に対して行われた具体的な分割処理の例を示す平面図である。

[0227] いま、単位図形入力部111によって、図34(a)に示すような平行四辺形型の単位図形10が入力されたものとしよう。この単位図形10は、高さが W_{split} の平行四辺形をしており、横方向に細長い形状をしている。このような平行四辺形型の単位図形10に対しても、前述した図33(a)に示す分割方法を適用することができる。すなわち、単位図形分割部112は、図34(b)に示すように、一点鎖線で示す左端位置から、たとえば寸法 W_{split} の間隔で破線で示す分割線を配置し、この分割線によって、単位図形10を4分割することができる。その結果、単位図形10は、図34(c)に示すように、4つの分割図形10a, 10b, 10c, 10dに分割される。

[0228] ここで、中間に位置する分割図形10b, 10cは一辺が W_{split} の正方形になるが、両端に位置する分割図形10a, 10dは高さが W_{split} の台形になる。前述した図33(b)に示す分割方法を適用すれば、任意形状の単位図形10を、複数の水平分割線によって分割することにより、 X 軸方向に伸びる細長い複数の帯状図形を形成することができ、これを更に垂直分割線によって分割することにより、最終的には、複数の台形からなる分割図形を形成することができる。実用上は、単位図形入力部111が、多角形からなる単位図形10を入力し、単位図形分割部112が、この単位図形10を分割して、台形からなる分割図形を形成するようにするのが好ましい。

[0229] ここで、「台形」とは、一般的には「1組の対辺が平行な四辺形（広辞苑第六版）」と定義されている。図35(a)は、このような定義に合致する典型的な「台形」であり、この例では、高さおよび下底が寸法 W_{split} に設定

されている。図35(b)は、一辺が W_{split} の正方形であり、上底、下底、高さがいずれも寸法 W_{split} に設定され、2組の対辺が平行な四辺形ということになる。上底、下底が寸法 W_{split} ではない場合は長方形になる。一方、図35(c)は、底辺が寸法 W_{split} より小さく、高さが W_{split} の三角形であるが、上底の長さが0の特殊な台形ということもできる。そこで、本願における「台形」なる用語は、一般的な台形の定義に該当する図形のみならず、正方形、長方形、三角形も含むような拡張した定義に該当する図形を広く含むものとする。

[0230] 上述したとおり、図33(b)に示す分割方法を適用すれば、任意の多角形からなる単位図形10を、図35に示すような「台形」からなる分割図形に分割することができる。しかも、分割線の間隔を所定間隔 W_{split} に設定すれば、図35(a), (b), (c)に示すように、いずれも縦幅 W_{split} 以下、横幅 W_{split} 以下となるような台形状の分割図形にすることが可能である。本発明を実施する上では、単位図形分割部112に、所定寸法 W_{split} を設定して、縦幅 W_{split} 以下、横幅 W_{split} 以下となる台形状の分割図形を形成させることが好ましい。以下にその理由を説明する。

[0231] 図1に示す図形パターンの形状補正装置100によって作成された補正図形パターンは、実際のリソグラフィプロセスに利用される。ここで、実際のリソグラフィプロセスには、光描画装置や電子線描画装置を用いて、光ビームや電子ビームを実基板S上に照射する露光工程が含まれる。この露光工程において、実基板S上に1ショットのビーム照射によって露光可能な最大照射可能領域は、個々の描画装置のビーム最大成型サイズ B_{max} によって定まる。

[0232] たとえば、台形状に成型したビームを照射可能な描画装置において、そのビーム最大成型サイズ B_{max} が 20 nm であった場合、当該描画装置による1ショットのビーム照射では、最大で一辺が 20 nm の正形状の照射領域しか形成することができない。したがって、たとえば、一辺が 30 nm の正形状の領域に対して露光を行いたい場合は、照射位置を縦方向および横方

向にずらしながら、合計4ショットの照射が必要になる。このような点を考慮すると、できるだけ効率的な露光工程を行うためには、実際のリソグラフィプロセスに用いられる補正図形パターン15を構成する個々の単位図形も、描画装置のビーム最大成型サイズ B_{max} の値を考慮した形状にしておくのが好ましい。

[0233] ここでは、図34(a)に示すような平行四辺形型の単位図形10についての形状補正を行う場合を考えてみる。上述した分割方法によれば、当該単位図形10は、図34(c)に示すような4組の分割図形10a, 10b, 10c, 10dに分割される。このとき、単位図形分割部112に、リソグラフィプロセスで用いる描画装置のビーム最大成型サイズ B_{max} を設定する機能を設けておけば、単位図形の分割に用いる分割線の配置間隔 W_{split} を、このビーム最大成型サイズ B_{max} 以下に設定した分割処理を行うことができる。

[0234] 上述した分割方法を採用すれば、図35に示すように、分割処理によって得られる個々の分割図形は、いずれも縦幅 W_{split} 以下、横幅 W_{split} 以下の台形になる。したがって、単位図形分割部112が、分割線の配置間隔 W_{split} を、ビーム最大成型サイズ B_{max} 以下に設定すれば、分割処理によって得られる個々の分割図形は、いずれも縦幅 B_{max} 以下、横幅 B_{max} 以下の台形になる。これは、最終的に得られる補正図形パターン15を用いて実際のリソグラフィプロセスを行う場合、露光工程の効率化（トータルショット数の低減）を図る上で有利である。

[0235] 後述するように、単位図形分割部112による分割処理で得られた個々の分割図形の辺のいくつかは、パターン補正ユニット140が補正図形パターンを作成する際に移動させる輪郭線分として採用され、移動後の輪郭線分により、補正図形パターンの輪郭が形成されることになる。

[0236] 図36は、図34(c)に示す単位図形10を構成する各分割図形10a, 10b, 10c, 10dに対する形状補正を行うことにより得られた補正後の単位図形15の一例を示す平面図である。図示のとおり、各分割図形10

a, 10b, 10c, 10dは、それぞれ分割図形15a, 15b, 15c, 15dに補正されており、これら補正後の分割図形15a, 15b, 15c, 15dによって、補正後の単位図形15が構成されている。具体的には、この例の場合、図34(c)に示す分割図形10b, 10dの上辺および下辺を輪郭線分として移動させることにより、若干縦幅が縮まった分割図形15b, 15dが得られている。

[0237] この例の場合、図1に示す形状補正装置100は、元図形パターン10に含まれていた図34(c)に示す単位図形10に対する補正を行うことにより、図36に示す補正後の単位図形15を含む補正図形パターン15を作成する形状補正処理を行うことになり、この補正図形パターン15が実際のリソグラフィプロセスに利用されることになる。ここで、この実際のリソグラフィプロセスにおいて、図36に示す補正後の単位図形15に基づいてどのような露光工程が行われるかを考えてみる。

[0238] 図37は、図36に示す補正後の単位図形15に基づく露光工程の一例を示す平面図である。ここで、太線の正方形は、一辺がビーム最大成型サイズ B_{max} の照射領域を示している。別言すれば、実際のリソグラフィプロセスに用いられる露光装置は、1ショットの照射で、この太線の正方形内の領域に対して光ビームや電子ビームを照射する機能を有している。もちろん、実際の照射領域は、単位図形15の内側領域に対応する部分のみになる。

[0239] したがって、図37に示す例の場合、合計4ショットによって、単位図形15に基づく露光工程は完了する。すなわち、1回目のショットでは、図37(a)の太線の正方形内のうちハッチングが施された分割図形15aの内部領域に対する照射露光が行われ、2回目のショットでは、図37(b)の太線の正方形内のうちハッチングが施された分割図形15bの内部領域に対する照射露光が行われ、3回目のショットでは、図37(c)の太線の正方形内のうちハッチングが施された分割図形15cの内部領域に対する照射露光が行われ、4回目のショットでは、図37(d)の太線の正方形内のうちハッチングが施された分割図形15dの内部領域に対する照射露光が行われる。

[0240] このように、補正後の単位図形 15 に基づく露光工程が、合計 4 ショットで済んだのは、単位図形分割部 112 に、縦幅 $B_{\max}$ 以下、横幅 $B_{\max}$ 以下となる台形状の分割図形を形成させ、パターン補正ユニット 140 に、この分割図形の辺を個々の輪郭線分として移動させる補正を行うことにより補正後の単位図形 15 を作成させたためである。

[0241] もちろん、このような分割処理や補正処理を採用したとしても、必ずしも露光工程時のショット数を低減させる効果が得られるとは限らない。たとえば、図 36 の例では、分割図形 15b, 15d は、分割図形 10b, 10d の縦幅を若干縮めた図形になっているが、逆に、縦幅を若干広げた図形になった場合には、図 37 (b), (d) の露光工程は、1 ショットで行うことはできなくなる。しかしながら、上述した分割処理や補正処理を採用すれば、多数の単位図形を含む図形パターン全体としてみたときに、露光工程の効率化を図ることが可能な補正図形パターン 15 を得ることができる。

[0242] 図 32 に示す評価点設定ユニット 110 のブロック図では、ビーム最大成型サイズ $B_{\max}$ が、単位図形分割部 112 に与えられているが、これは単位図形分割部 112 が、実際のリソグラフィプロセスで用いる描画装置のビーム最大成型サイズ $B_{\max}$ を設定し、単位図形の分割に用いる分割線の配置間隔をこのビーム最大成型サイズ $B_{\max}$ 以下に設定する機能を有していることを示している。もちろん、このような機能は、本発明を実施する上で必須の機能ではないが、実用上は、このような機能をもたせておけば、上述したように露光工程の効率化を図る上で好ましい。

[0243] <6. 2 ベクトルおよび通常端点の定義>

続いて、図 32 に示す評価点設定ユニット 110 における分割図形ベクトル定義部 113, 単位図形ベクトル定義部 114, 通常端点定義部 115 の処理機能について説明する。§ 6. 1 で述べたとおり、単位図形分割部 112 による分割処理によって、単位図形 10 は複数の分割図形に分割される。そして、この分割図形の辺のいくつかは、補正の際に移動させる輪郭線分として採用される。

[0244] たとえば、図34(c)に示す例の場合、単位図形10は4組の分割図形10a, 10b, 10c, 10dに分割されている。これらの分割図形はいずれも4辺を有する台形であるから、分割図形の辺の数の合計は16になる。ただ、元の単位図形10の輪郭線を構成する辺の数は合計10である。そこで、合計16組の辺の中から、輪郭線を構成する10組の辺を抽出する処理が必要になる。分割図形ベクトル定義部113および単位図形ベクトル定義部114の役割は、このように、単位図形10の輪郭線を構成する特定の辺を抽出することにより、通常端点定義部115の役割は、こうして抽出された各辺の両端に通常端点を定義することにある。

[0245] 図38は、図34(c)に示す四分分割後の平行四辺形型の単位図形10に対して、図32に示す分割図形ベクトル定義部113、単位図形ベクトル定義部114、通常端点定義部115による処理が行われた例を示す平面図である。以下、この平行四辺形型の単位図形10を例にとって、各定義部113, 114, 115による処理を具体的に説明する。

[0246] まず、分割図形ベクトル定義部113は、単位図形分割部112による分割処理によって形成された各分割図形について、当該分割図形を構成する多角形の輪郭線に沿って所定の順方向まわりに一周するように、多角形の個々の辺上にそれぞれ分割図形ベクトルを定義する処理を行う。たとえば、図34(c)に示す例の場合、4組の分割図形10a, 10b, 10c, 10dが形成されている。そこで、各分割図形について、たとえば、時計まわりの方向を順方向と定義すれば、図38(a)に示すように、各分割図形10a, 10b, 10c, 10d内に時計まわりの順方向矢印A1~A4を定義することができ、この順方向矢印A1~A4が示す方向に各分割図形を構成する台形の輪郭線を一周するように、各台形の辺上に分割図形ベクトルを定義することができる。

[0247] 図38(b)は、このようにして定義された分割ベクトルを示す平面図である。たとえば、分割図形10aを構成する台形は、図38(a)に示すように、4つの頂点 $\xi_1 \sim \xi_4$ を有しており、第1の辺 $\xi_1 - \xi_2$ と、第2の辺 ξ_2

2- ξ 3と、第3の辺 ξ 3- ξ 4と、第4の辺 ξ 4- ξ 1とを有している。このため、分割図形ベクトル定義部113は、分割図形10aについて、図38(b)に示すように、4組の分割図形ベクトル V_{a1} 、 V_{a2} 、 V_{a3} 、 V_{a4} を定義する処理を行う。

[0248] 別言すれば、分割図形ベクトル V_{a1} は、頂点 ξ 1を始点、頂点 ξ 2を終点として、第1の辺 ξ 1- ξ 2に沿ったベクトルであり、分割図形ベクトル V_{a2} は、頂点 ξ 2を始点、頂点 ξ 3を終点として、第2の辺 ξ 2- ξ 3に沿ったベクトルであり、分割図形ベクトル V_{a3} は、頂点 ξ 3を始点、頂点 ξ 4を終点として、第3の辺 ξ 3- ξ 4に沿ったベクトルであり、分割図形ベクトル V_{a4} は、頂点 ξ 4を始点、頂点 ξ 1を終点として、第4の辺 ξ 4- ξ 1に沿ったベクトルである。

[0249] しかも、これら4組のベクトル V_{a1} 、 V_{a2} 、 V_{a3} 、 V_{a4} は、順方向矢印 A_1 に沿って、分割図形10aを構成する台形の輪郭線を一周する向きに定義されており、1つのベクトルの終点が次のベクトルの始点となる関係にある。よって、4組のベクトル V_{a1} 、 V_{a2} 、 V_{a3} 、 V_{a4} の集合体は、分割図形10aを構成する台形の輪郭線を時計まわりに一周するベクトル群を構成することになる。

[0250] もちろん、分割図形ベクトル定義部113は、反時計まわりの方向を順方向と定義して同様の処理を行ってもかまわない。その場合、4組のベクトル V_{a1} 、 V_{a2} 、 V_{a3} 、 V_{a4} の集合体は、分割図形10aを構成する台形の輪郭線を反時計まわりに一周するベクトル群を構成することになる。このように、順方向を時計まわりとするか、反時計まわりとするかは、任意に設定することができるが、いずれか一方の設定をすべての分割図形について共通して適用する必要がある。たとえば、図38(a)に示す例の場合、順方向矢印 $A_1 \sim A_4$ はすべて時計まわりの矢印になっているが、これらすべてを反時計まわりの矢印にしてもかまわない。ただ、順方向矢印 A_1 は時計まわり、順方向矢印 A_2 は反時計まわり、というように、各分割図形ごとに設定を変えることはできない（後述するように、重複ベクトル対を除去するた

め)。

[0251] 図38には、分割図形が四角形の場合の例が示されているが、一般論で述べれば、分割図形ベクトル定義部113は、K角形からなる分割図形の各頂点に、輪郭線に沿って時計まわりもしくは反時計まわりの順に第1の頂点～第Kの頂点と番号を付し、第kの頂点を始点とし、第(k+1)の頂点(但し、k=Kの場合は第1の頂点)を終点とする第kの分割図形ベクトルを定義する処理を、k=1～Kまで繰り返し行うことにより、当該分割図形についての分割図形ベクトルを定義する処理を行うことになる。

[0252] かくして、分割図形ベクトル定義部113によって、図38(b)に示すように、各分割図形10a, 10b, 10c, 10dの各4辺上に、合計16組の分割図形ベクトルV a 1～V d 4が定義されることになる。なお、図38(b)では、図示の便宜上、同一の辺上に定義される2本のベクトルが、当該辺の位置から若干ずらして描かれているが、実際には、これら2本のベクトルは同一辺上に重複して定義されたベクトルである。たとえば、図38(b)に示すベクトルV a 2とV b 4とは、いずれも辺ξ2-ξ3上に定義されたベクトルである。

[0253] 単位図形ベクトル定義部114は、こうして定義された分割図形ベクトルの集合から、互いに隣接する一対の隣接多角形の重複する辺上に配置され、互いに向きが逆方向である重複ベクトル対を探索し、探索された重複ベクトル対のうちの重複区間部分を除去し、残ったベクトルにより、単位図形10の輪郭線に沿って配置された複数の単位図形ベクトルを定義する処理を行う。

[0254] たとえば、図38(b)に示す例の場合、合計16組の分割図形ベクトルV a 1～V d 4が定義されているので、これら16組の分割図形ベクトルV a 1～V d 4の中から、重複ベクトル対を探索し、探索された重複ベクトル対のうちの重複区間部分を除去する処理が行われる。ここで、重複ベクトル対とは、上述したように、「互いに隣接する一対の隣接多角形の重複する辺上に配置され、互いに向きが逆方向である一対のベクトル」であるから、図3

8 (b) に示す例の場合、ベクトル V_{a2} 、 V_{b4} が第 1 の重複ベクトル対として探索され、ベクトル V_{b2} 、 V_{c4} が第 2 の重複ベクトル対として探索され、ベクトル V_{c2} 、 V_{d4} が第 3 の重複ベクトル対として探索される。

[0255] これら各重複ベクトル対は、全区間において重複しているため、そのままそっくり除去する処理が行われる。図 38 (c) は、このような除去処理が行われた後の状態を示す平面図であり、図 38 (b) に示す重複ベクトル対 (V_{a2} 、 V_{b4})、(V_{b2} 、 V_{c4})、(V_{c2} 、 V_{d4}) がすべて除去されていることがわかる。ここでは、図 38 (b) に示す 16 組のベクトル $V_{a1} \sim V_{d4}$ を「分割図形ベクトル」と呼んでいるので、図 38 (c) に示す 10 組のベクトルは、図 38 (b) に示す 16 組のベクトルと区別するために「単位図形ベクトル」と呼ぶことにし、符号を $V_1 \sim V_{10}$ に付け直すことにする。したがって、図 38 (c) に示す単位図形ベクトル $V_1 \sim V_{10}$ は、実際には、図 38 (b) に示す分割図形ベクトル V_{a1} 、 V_{b1} 、……、 V_{a3} 、 V_{a4} と同じものである。

[0256] こうして得られた単位図形ベクトル $V_1 \sim V_{10}$ は、単位図形 10 の輪郭線上に配置されたベクトルであり、しかも順方向（図示の例の場合、時計まわり方向）に輪郭線を一周する向きに定義されており、1 つのベクトルの終点が次のベクトルの始点となる関係にある。よって、10 組の単位図形ベクトル $V_1 \sim V_{10}$ の集合体は、単位図形 10 の輪郭線を時計まわりに一周するベクトル群を構成することになる。

[0257] 通常端点定義部 115 は、こうして定義された個々の単位図形ベクトルの始点位置（終点位置でも同じ）に、それぞれ通常端点を定義する処理を行う。図 38 (d) は、図 38 (c) に示す 10 組の単位図形ベクトル $V_1 \sim V_{10}$ の始点位置に通常端点 $A \sim J$ （図では、 $\times$ 印で示す）を定義した状態を示す平面図である。こうして定義された各通常端点 $A \sim J$ は、結局、図 38 (a) に示す分割図形 10a、10b、10c、10d の各頂点のうち、単位図形 10 の輪郭線上に位置する頂点ということになる。別言すれば、各通常端点 $A \sim J$ は、各分割図形の辺のうち、単位図形 10 の輪郭線上に位置する辺の

両端を画定する端点ということになる。なお、このようにして定義された端点を「通常端点」と呼ぶのは、後述する補充端点定義部 118 によって追加定義される「補充端点」と区別するための便宜である。

[0258] 一方、図 39 は、L 字型の単位図形 10 に対して、図 32 に示す単位図形分割部 112、分割図形ベクトル定義部 113、単位図形ベクトル定義部 114、通常端点定義部 115 による処理が行われた例を示す平面図である。単位図形入力部 111 によって、図 39 (a) に示すような L 字型の単位図形 10 が入力されると、単位図形分割部 112 によって、図 39 (b) に示すような分割処理が行われ、3 組の分割図形 10 a, 10 b, 10 c が形成される。

[0259] そして、これら分割図形 10 a, 10 b, 10 c について、図示のような時計まわりの向きに順方向矢印 A1 ~ A3 が設定され、分割図形ベクトル定義部 113 によって、図 39 (c) に示すように、合計 12 組の分割図形ベクトル V a1 ~ V c4 が定義される。続いて、単位図形ベクトル定義部 114 によって、重複ベクトル対が探索され、探索された重複ベクトル対のうちの重複区間部分を除去する処理が行われる。前述した図 38 (b) に示す例の場合、探索された重複ベクトル対の全区間が重複区間部分となっていたため、重複ベクトル対がそっくり除去されることになったが、図 39 (c) に示す例の場合、一部のみが重複する重複ベクトル対が存在する。

[0260] 具体的には、図 39 (c) に示す例の場合、ベクトル V a3, V b1 が第 1 の重複ベクトル対として探索され、ベクトル V b2, V c4 が第 2 の重複ベクトル対として探索される。ここで、第 2 の重複ベクトル対 V b2, V c4 は、全区間が重複区間部分となっているため、これらをそっくり除去する処理が行われるが、第 1 の重複ベクトル対 V a3, V b1 は、図 39 (d) の上段に示すとおり、一部のみが重複区間部分（太線で示す部分）となっているため、この重複区間部分のみを除去する処理が行われる。その結果、図 39 (d) の下段に示すとおり、非重複区間部分のみが単位図形ベクトル V3 として残ることになる。

[0261] 図39(e)は、単位図形ベクトル定義部114によって、上述のような除去処理が行われた後の状態を示す平面図であり、9組の単位図形ベクトル $V_1 \sim V_9$ が残った状態が示されている。こうして得られた単位図形ベクトル $V_1 \sim V_9$ は、図39(a)に示すL字型の単位図形10の輪郭線上に配置されたベクトルであり、しかも順方向（図示の例の場合、時計まわり方向）に輪郭線を一周する向きに定義されており、1つのベクトルの終点が次のベクトルの始点となる関係にある。よって、これら9組の単位図形ベクトル $V_1 \sim V_9$ の集合体は、単位図形10の輪郭線を時計まわりに一周するベクトル群を構成することになる。

[0262] 通常端点定義部115は、こうして定義された個々の単位図形ベクトルの始点位置に、それぞれ通常端点を定義する処理を行う。図39(f)は、図39(e)に示す9組の単位図形ベクトル $V_1 \sim V_9$ の始点位置に通常端点 $A \sim I$ （図では、×印で示す）を定義した状態を示す平面図である。

[0263] 続いて、内部に図形外の領域が存在する、いわゆる穴開きの単位図形についての処理例を述べる。図40は、口の字型の単位図形10に対して、図32に示す単位図形分割部112、分割図形ベクトル定義部113、単位図形ベクトル定義部114、通常端点定義部115による処理が行われた例を示す平面図である。この例は、単位図形入力部111によって、図40(a)に示すような口の字型の単位図形10が入力された場合の例である。図示の口の字型の単位図形10の場合、図形本体部10Xはハッチングを施した領域であるが、内部に矩形状の穴開き部10Yが存在する。すなわち、このような穴開きの単位図形10は、外側輪郭線 O_{out} と内側輪郭線 O_{in} とを有していることになる。このような穴開きの単位図形10が入力された場合でも、基本的には、これまで述べてきた処理と同様の処理を実行すればよい。

[0264] まず、単位図形分割部112によって、図40(b)に示すような分割処理が行われ、8組の分割図形10a $\sim$ 10hが形成される。そして、これら分割図形10a $\sim$ 10hについて、図40(c)に示すように、時計まわりの向きに順方向矢印 $A_1 \sim A_8$ が設定され、分割図形ベクトル定義部113によ

って、図40(d)に示すように、合計32組の分割図形ベクトル $V_{a1} \sim V_{h4}$ が定義される。続いて、単位図形ベクトル定義部114によって、重複ベクトル対が探索され、探索された重複ベクトル対のうちの重複区間部分を除去する処理が行われる。

[0265] 図40(e)は、単位図形ベクトル定義部114によって、上述した除去処理が行われた後の状態を示す平面図であり、20組の単位図形ベクトル $V_1 \sim V_{20}$ が残った状態が示されている。ここで留意すべき点は、単位図形ベクトル $V_1 \sim V_{12}$ は、図40(a)に示す口の字型の単位図形10の外側輪郭線 O_{out} 上に配置されたベクトルであり、しかも順方向（図示の例の場合、時計まわり方向）に輪郭線を一周する向きに定義されているのに対して、単位図形ベクトル $V_{13} \sim V_{20}$ は、図40(a)に示す口の字型の単位図形10の内側輪郭線 O_{in} 上に配置されたベクトルであり、しかも逆方向（図示の例の場合、反時計まわり方向）に輪郭線を一周する向きに定義されている点である。

[0266] 本願では、このような穴開きの単位図形について、外側輪郭線 O_{out} を構成する図形を「正図形」と呼び、内側輪郭線 O_{in} を構成する図形を「反図形」と呼ぶことにする。図示の例の場合、単位図形ベクトル $V_1 \sim V_{12}$ は、正図形を順方向に一周するベクトル群であり、単位図形ベクトル $V_{13} \sim V_{20}$ は、反図形を逆方向に一周するベクトル群である。したがって、穴開きの単位図形の場合、ベクトル群が輪郭線に沿って順方向に一周する場合、当該輪郭線は単位図形の外側輪郭線 O_{out} を構成する正図形であると判断でき、ベクトル群が輪郭線に沿って反方向に一周する場合、当該輪郭線は単位図形の内側輪郭線 O_{in} を構成する反図形であると判断できる。このような正図形と反図形の相違は、§6.3で述べる端点再構築処理に利用することができる。

[0267] 通常端点定義部115は、こうして定義された個々の単位図形ベクトルの始点位置に、それぞれ通常端点を定義する処理を行う。図40(f)は、図40(e)に示す20組の単位図形ベクトル $V_1 \sim V_{20}$ の始点位置に通常端点

A～T（図では、×印で示す）を定義した状態を示す平面図である。図示のとおり、穴開きの単位図形10の外側輪郭線O_{out}と内側輪郭線O_{in}との双方に、それぞれ通常端点が定義されることになる。

[0268] <6. 3 端点の再構築>

続いて、図32に示す評価点設定ユニット110における端点再構築部116の機能について説明する。この端点再構築部116は、通常端点定義部115によって単位図形の輪郭線上に定義された通常端点に対して、数もしくは位置を修正する再構築を行う構成要素である。§6.2で述べたとおり、通常端点定義部115は、個々の分割図形の頂点位置を基準にして通常端点の定義を行うことになるので、単位図形分割部112による分割処理が適切に行われていれば、定義された通常端点の数や位置も適切になされるはずである。ただ、実際には、こうして定義された通常端点について、数や位置を調整した方が、より効率的な評価点設定が可能になる。

[0269] そこで、ここで述べる実施形態の場合、通常端点定義部115によって定義された通常端点に対して、端点再構築部116による再構築を行っている。もちろん、このような再構築処理は、本発明にとって必須の処理ではない。したがって、本発明を実施するにあたり、端点再構築部116は省略することも可能である。ただ、実用上は、端点再構築部116を設けた方が、より効率的な評価点設定を行うことが可能になる。端点再構築部116は、通常端点定義部115によって定義された通常端点について、その数を減らしたり位置を修正したりして再構築を行う。以下、具体的に再構築を行ういくつかの方法を説明する。

[0270] (1) 3点式の端点除去処理による再構築

図41は、図32に示す端点再構築部116による端点再構築処理として、3点式の端点除去処理を採用した場合の手順を示す模式図（図41(a)）および流れ図（図41(b)）である。端点除去処理は、通常端点定義部115によって定義された一部もしくは全部の単位図形について、通常端点（実際には、単位図形を構成する多角形の頂点を構成しない通常端点）の一部も

しくは全部を除去する処理であり、通常端点の数を減じることによって再構築を行う処理ということができる。

[0271] § 6. 2で述べたとおり、通常端点定義部 115によって定義された通常端点は、単位図形 10の輪郭線上に所定の順序で定義されている。たとえば、図 38(d)に示す例の場合、単位図形 10の輪郭線上にA～Jという順序で通常端点が定義されており、図 39(f)に示す例の場合、単位図形 10の輪郭線上にA～Iという順序で通常端点が定義されており、図 40(f)に示す例の場合、単位図形 10の外側輪郭線O_{out}上にA～L、内側輪郭線O_{in}上にM～Tという順序で通常端点が定義されている。ここで述べる3点式の端点除去処理は、判定対象となる通常端点について、1つ前の通常端点と1つ後の通常端点との距離を参照して、除去すべきか否かの判定を行う手法を採用するものである。

[0272] いま、単位図形の1つの輪郭線上に合計N個の通常端点 $T(1) \sim T(N)$ が定義されており、そのうちの第nの通常端点 $T(n)$ を判定対象として、当該通常端点 $T(n)$ を除去すべきか否かを判定する場合を考える。この場合、図 41(a)に示すように、除去判定対象となる第nの通常端点 $T(n)$ と1つ前の第 $(n-1)$ の通常端点 $T(n-1)$ との距離 L_1 と、除去判定対象となる第nの通常端点 $T(n)$ と1つ後の第 $(n+1)$ の通常端点 $T(n+1)$ との距離 L_2 と、を求める。

[0273] ただし、距離 L_1 については、若干留意する点がある。たとえば、図 41(a)において、通常端点 $T(n-1)$ が既に除去対象となっていた場合には、その1つ前の通常端点 $T(n-2)$ と $T(n)$ との距離を L_1 とする。また、通常端点 $T(n-2)$ も除去対象となっていた場合には、更にその1つ前の通常端点 $T(n-3)$ と $T(n)$ との距離を L_1 とする。結局、ここでの距離 L_1 は、一般論として言えば、除去判定対象となる第nの通常端点 $T(n)$ と、その直前に位置する除去対象となっていない通常端点 $T(n-m)$ (但し、mは、自然数であり、 $n-m < 1$ の場合は、 $T(n-m+N)$)との距離と定義されることになる。こうして、距離 L_1 、 L_2 を定義した上で

、次の2つの判定条件が共に満たされた場合に、第 n の通常端点 $T(n)$ を除去対象と判定する。

[0274] 第1の判定条件は、「 L_1 および L_2 の少なくとも一方が所定の基準値 W_{merge} 以下である」という条件である。ここで、 W_{merge} は、予め設定された所定の基準値であり、一般的には、単位図形分割部112が分割処理に用いた W_{split} よりも小さな値を設定するのが好ましい。基準値 W_{merge} は、除去判定対象となる通常端点 $T(n)$ が、隣接する通常端点 $T(n-m)$ もしくは $T(n+1)$ にある程度接近していた場合に、通常端点 $T(n)$ を除去して通常端点 $T(n-m)$ もしくは $T(n+1)$ に融合させるべき、と判定する際の臨界となる接近距離を定める値である。

[0275] この基準値 W_{merge} を大きく設定すれば、通常端点 $T(n)$ を除去すべきと判定される可能性は高くなり、小さく設定すれば可能性は低くなる。もちろん、除去すべきと判定される可能性が高くなれば、それだけ演算負担は軽減されるがシミュレーションの精度は低下し、可能性が低くなれば、それだけシミュレーションの精度は向上するが演算負担は大きくなる。よって、実用上は、シミュレーションの精度維持と演算負担の軽減とのバランスを考慮して、基準値 W_{merge} を適切な値に設定すればよい。

[0276] 第2の判定条件は、「通常端点 $T(n-1)$ 、 $T(n)$ 、 $T(n+1)$ の3点が一直線上にある」という条件である。図41(a)には、そのような条件が満足された状態が図示されているが、たとえば、通常端点 $T(n)$ が単位図形の頂点である場合、通常端点 $T(n-1)$ と $T(n+1)$ とは別々の辺上の端点になるため、当該条件は満足されないことになる。したがって、この第2の判定条件は、「上記3点が同一の辺上にある」という条件ということになる。別言すれば、単位図形を構成する多角形の頂点を構成しない通常端点を中間通常端点と呼ぶことにすれば、第2の判定条件は、「除去判定対象である通常端点 $T(n)$ が中間通常端点である」という条件になる。

[0277] 図41(b)は、このような3点式の端点除去処理に基づく端点再構築の手順を示す流れ図である。まず、ステップS51において、除去判定対象とな

る通常端点の番号を示すパラメータ n を $n = 1$ に設定し、ステップ $SS2$ において、 $L1 \leq W_{merge}$ か否かを判定する。ここで否定的な判定がなされた場合には、ステップ $SS3$ において、 $L2 \leq W_{merge}$ か否かを判定する。ステップ $SS2$ 、 $SS3$ のいずれかにおいて肯定的な判定がなされたら、続いて、ステップ $SS3$ において、 $T(n-1)$ 、 $T(n)$ 、 $T(n+1)$ の3点が一直線上にあるか否かが判定され、肯定的な判定がなされたら、ステップ $SS5$ において、通常端点 $T(n)$ を除去対象とする処理が行われる。一方、ステップ $SS3$ もしくは $SS4$ において否定的な判定がなされたら、ステップ $SS6$ において、通常端点 $T(n)$ を除去対象としない処理が行われる。

[0278] このような処理が、ステップ $SS7$ および $SS8$ を介して、パラメータ n を1ずつ増加させながら、 $n = N$ に到達するまで繰り返し実行される。そして、 $n = N$ に到達したら、ステップ $SS7$ から $SS9$ へと移行し、除去対象とされた通常端点が一括して除去される。

[0279] なお、上記処理手順において、 $n = 1$ の場合は、除去判定対象が第1の通常端点 $T(1)$ ということになるので、上記各判定条件における通常端点 $T(n-1)$ は、第 N の通常端点 $T(N)$ と読み替える必要がある。同様に、 $n = N$ の場合は、除去判定対象が第 N の通常端点 $T(N)$ ということになるので、上記各判定条件における通常端点 $T(n+1)$ は、第1の通常端点 $T(1)$ と読み替える必要がある。

[0280] 結局、ここで述べる3点式の端点除去処理による再構築を採用する場合を一般論として述べれば、端点再構築部 116 は、1つの単位図形の1本の輪郭線について定義されている合計 N 個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1)$ ~ 第 N の通常端点 $T(N)$ と呼び、第 n の通常端点 $T(n)$ に着目して、第 $(n-1)$ の通常端点 $T(n-1)$ (但し、 $n = 1$ の場合は第 N の通常端点 $T(N)$: 以下同様) と第 n の通常端点 $T(n)$ との間の距離を $L1$ とし、第 n の通常端点 $T(n)$ と第 $(n+1)$ の通常端点 $T(n+1)$ (但し、 $n = N$ の場合は第1の通常端点 $T(1)$: 以下同様) との間の距離を $L2$ としたときに、 $L1$ および $L2$ の少なくと

も一方が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ 、 $T(n)$ 、 $T(n+1)$ の3点が一直線上にある場合に、第 n の通常端点 $T(n)$ を除去対象とする判定処理を、 $n=1 \sim N$ まで繰り返し実行し、除去対象となった通常端点を除去する端点除去処理を行うことになる。

[0281] 図42は、図38(d)に示す平行四辺形型の単位図形10上の通常端点 $A \sim J$ に対して、図41に示す3点式の端点除去処理を適用して端点の再構築を行った例を示す平面図である。ここで、図42(a)は、再構築前に、通常端点 D 、 I （図では、白抜き三角形で示されている）が除去対象と判定された状態を示し、図42(b)は、これら除去対象と判定された通常端点 D 、 I を除去した再構築後の状態を示す。

[0282] 図42に示す例において、通常端点 D が除去対象と判定された理由は、端点 D 、 E 間の距離 L_{de} について、「 $L_{de} \leq W_{merge}$ 」なる第1の判定条件が満たされるとともに、「3つの端点 C 、 D 、 E が一直線上にある」という第2の判定条件が満たされたためである。同様に、通常端点 I が除去対象と判定された理由は、端点 I 、 J 間の距離 L_{ij} について、「 $L_{ij} \leq W_{merge}$ 」なる第1の判定条件が満たされるとともに、「3つの端点 H 、 I 、 J が一直線上にある」という第2の判定条件が満たされたためである。結局、端点再構築処理により、端点 D は端点 E に融合され、端点 I は端点 J に融合されたことになり、10個の通常端点は8個に減少したことになる。

[0283] 端点 D 、 I を除去したことにより、後に行われるシミュレーションの精度は若干低下することになるが、距離 L_{de} 、 L_{ij} は、いずれも基準値 W_{merge} 以下であるため、実用上、大きな支障はない。一方、端点 D 、 I を除去したことにより、シミュレーションの演算負担は軽減される。

[0284] 図43は、図39(f)に示すL字型の単位図形10上の通常端点 $A \sim I$ に対して、図41に示す3点式の端点除去処理を適用して端点の再構築を行った例を示す平面図である。ここで、図43(a)は、再構築前に、通常端点 D （図では、白抜き三角形で示されている）が除去対象と判定された状態を示し、図43(b)は、この除去対象と判定された通常端点 D を除去した再構築

後の状態を示す。

[0285] 図43に示す例において、通常端点Dが除去対象と判定された理由は、端点C、D間の距離 L_{cd} について、「 $L_{cd} \leq W_{merge}$ 」なる第1の判定条件が満たされるとともに、「3つの端点C、D、Eが一直線上にある」という第2の判定条件が満たされたためである。結局、端点再構築処理により、端点Dは端点Cに融合されたことになり、9個の通常端点は8個に減少したことになる。やはり、この再構築により、後に行われるシミュレーションの精度は若干低下することになるが、距離 L_{cd} は基準値 W_{merge} 以下であるため、実用上、大きな支障はない。一方、端点Dを除去したことにより、シミュレーションの演算負担は軽減される。

[0286] (2) 4点式の端点除去処理による再構築

図44は、図32に示す端点再構築部116による端点再構築処理として、4点式の端点除去処理を採用した場合の手順を示す模式図(図44(a))および流れ図(図44(b))である。ここで説明する端点除去処理は、上述した3点式の端点除去処理と同様に、通常端点定義部115によって定義された一部もしくは全部の単位図形について、中間通常端点の一部もしくは全部を除去する処理であり、通常端点の数を減じることによって再構築を行う処理である。ただ、ここで述べる4点式の端点除去処理は、隣接する2組の通常端点を除去判定対象とするものであり、この除去判定対象となる2組の通常端点について、1つ前の通常端点と1つ後の通常端点との距離を参照して、除去すべきか否かの判定とともに、どちらを除去すべきかの判定を行う手法を採用するものである。

[0287] いま、単位図形の1つの輪郭線上に合計N個の通常端点 $T(1) \sim T(N)$ が定義されており、そのうちの第nの通常端点 $T(n)$ および第 $(n+1)$ の通常端点 $T(n+1)$ の2点を判定対象として、当該判定対象を除去すべきか否か、除去すべき場合は2点のうちのどちらを除去すべきか、を判定する場合を考える。この場合、図44(a)に示すように、除去判定対象となる第nの通常端点 $T(n)$ および第 $(n+1)$ の通常端点 $T(n+1)$ の2

点間の距離 L_2 を求め、次の2つの判定条件が共に満たされた場合に、いずれか一方を除去対象と判定する。

[0288] 第1の判定条件は、「 L_2 が所定の基準値 W_{merge} 以下である」という条件である。ここで、 W_{merge} は、上述した3点式の端点除去処理と同様に、予め設定された所定の基準値であり、一般的には、単位図形分割部112が分割処理に用いた W_{split} よりも小さな値を設定するのが好ましい。基準値 W_{merge} は、除去判定対象となる2組の通常端点 $T(n)$ 、 $T(n+1)$ が、ある程度接近していた場合に、いずれか一方を除去することにより両者を融合させるべき、と判定する際の臨界となる接近距離を定める値である。したがって、3点式の端点除去処理と同様に、シミュレーションの精度維持と演算負担の軽減とのバランスを考慮して、基準値 W_{merge} を適切な値に設定すればよい。

[0289] 第2の判定条件は、第 n の通常端点 $T(n)$ の1つ前の第 $(n-1)$ の通常端点 $T(n-1)$ と、第 $(n+1)$ の通常端点 $T(n+1)$ の1つ後の第 $(n+2)$ の通常端点 $T(n+2)$ とを考慮して、「通常端点 $T(n-1)$ 、 $T(n)$ 、 $T(n+1)$ 、 $T(n+2)$ の4点が一直線上にある」という条件である。図44(a)には、そのような条件が満足された状態が図示されているが、たとえば、通常端点 $T(n)$ が単位図形の頂点である場合、通常端点 $T(n-1)$ と $T(n+1)$ とは別々の辺上の端点になるため、当該条件は満足されないことになる。したがって、この第2の判定条件は、「上記4点が同一の辺上にある」という条件ということになる。別言すれば、単位図形を構成する多角形の頂点を構成しない通常端点を中間通常端点と呼ぶことにすれば、第2の判定条件は、「除去判定対象である通常端点 $T(n)$ および $T(n+1)$ が中間通常端点である」という条件になる。

[0290] 上記2つの判定条件が共に満たされた場合には、通常端点 $T(n)$ および $T(n+1)$ のいずれか一方が除去対象と判定される。ここで、いずれを除去対象とすべきかは、通常端点 $T(n)$ と $T(n-1)$ との距離を L_1 とし、通常端点 $T(n+1)$ と $T(n+2)$ との距離を L_3 としたときに、距離

L_1 , L_3 の大小関係に基づいて決定すればよい。すなわち、 $L_1 < L_3$ の場合は通常端点 $T(n)$ を除去対象とし、 $L_1 > L_3$ の場合は通常端点 $T(n+1)$ を除去対象とする。そうすれば、除去対象となった端点を挟む2組の端点間距離（前者の場合は $L_1 + L_2$ 、後者の場合は $L_2 + L_3$ ）をできるだけ小さくすることができるので、除去処理によるシミュレーション精度低下の影響をできるだけ小さくすることができる。なお、 $L_1 = L_3$ の場合は、通常端点 $T(n)$, $T(n+1)$ のいずれを除去対象にしてもよい。また、4点は一直線上にないが、3点は一直線上にある、という場合は、前述した3点式の端点除去処理に準じた方法を実行すればよい。

[0291] なお、ここでも距離 L_1 については、若干留意する点がある。たとえば、図44(a)において、通常端点 $T(n-1)$ が既に除去対象となっていた場合には、その1つ前の通常端点 $T(n-2)$ と $T(n)$ との距離を L_1 とする。また、通常端点 $T(n-2)$ も除去対象となっていた場合には、更にその1つ前の通常端点 $T(n-3)$ と $T(n)$ との距離を L_1 とする。結局、ここでの距離 L_1 は、一般論として言えば、除去判定対象となる第 n の通常端点 $T(n)$ と、その直前に位置する除去対象となっていない通常端点 $T(n-m)$ （但し、 m は、自然数であり、 $n-m < 1$ の場合は、 $T(n-m+N)$ ）との距離と定義されることになる。

[0292] 図44(b)は、このような4点式の端点除去処理に基づく端点再構築の手順を示す流れ図である。まず、ステップSS11において、除去判定対象となる通常端点の番号を示すパラメータ n を $n=1$ に設定し、ステップSS12において、 $L_2 \leq W_{\text{merge}}$ か否かを判定する。ここで肯定的な判定がなされた場合には、ステップSS13において、 $T(n-1)$, $T(n)$, $T(n+1)$, $T(n+2)$ の4点が一直線上にあるか否かが判定される。ここで、肯定的な判定がなされたら、 $T(n)$, $T(n+1)$ のいずれか一方を除去対象とすることになるが、どちらを除去対象とすべきかを決めるために、ステップSS14において、 $L_1 \leq L_3$ なる条件判定を行う（代わりに、 $L_1 < L_3$ なる条件判定を行ってもよい）。ここで肯定的な判定がなされた場

合には、ステップSS15において端点 $T(n)$ を除去対象とし、否定的な判定がなされた場合には、ステップSS16において端点 $T(n+1)$ を除去対象とする。

[0293] 一方、ステップSS13において否定的な判定がなされたら、ステップSS17において $T(n-1)$ 、 $T(n)$ 、 $T(n+1)$ の3点が一直線上にあるか否かが判定される。ここで肯定的な判定がなされたら、ステップSS15へと進み、端点 $T(n)$ を除去対象とし、否定的な判定がなされたら、ステップSS18へと進み、 $T(n)$ 、 $T(n+1)$ 、 $T(n+2)$ の3点が一直線上にあるか否かが判定される。そして、このステップSS18で肯定的な判定がなされたら、ステップSS16へと進み、端点 $T(n+1)$ を除去対象とする。これは、前述した3点式の端点除去処理と同様の処理である。また、ステップSS12で否定的な判定がなされた場合や、ステップSS18で否定的な判定がなされた場合は、ステップSS19へ進み、 $T(n)$ 、 $T(n+1)$ はいずれも除去対象としない処理が行われる。

[0294] このような処理が、ステップSS20およびSS21を介して、 $n \geq N$ が満たされるまで、パラメータ n を1ずつ増加させながら繰り返し実行される。なお、通常端点 $T(n)$ が既に除去対象となっていた場合には、当該パラメータ n についての処理手順はスキップすることができる。そして、 $n \geq N$ が満たされたら、ステップSS20からSS22へと移行し、除去対象とされた通常端点が一括して除去される。

[0295] なお、上記処理手順において、 $n=1$ の場合は、図44における左側の除去判定対象が第1の通常端点 $T(1)$ ということになるので、上記各判定条件における通常端点 $T(n-1)$ は、第 N の通常端点 $T(N)$ と読み替える必要がある。同様に、 $n=N-1$ の場合は、左側の除去判定対象が第 $(N-1)$ の通常端点 $T(N-1)$ 、右側の除去判定対象が第 N の通常端点 $T(N)$ ということになるので、上記各判定条件における通常端点 $T(n+2)$ は、第1の通常端点 $T(1)$ と読み替える必要がある。また、 $n=N$ の場合は、左側の除去判定対象が第 N の通常端点 $T(N)$ ということになるので、上

記各判定条件における通常端点 $T(n+1)$ は、第1の通常端点 $T(1)$ と読み替え、通常端点 $T(n+2)$ は、第2の通常端点 $T(2)$ と読み替える必要がある。

[0296] 結局、ここで述べる4点式の端点除去処理による再構築を採用する場合を一般論として述べれば、端点再構築部116は、1つの単位図形の1本の輪郭線について定義されている合計 N 個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1) \sim$ 第 N の通常端点 $T(N)$ と呼び、除去判定対象となる第 n の通常端点 $T(n)$ と、その直前に位置する除去対象となっていない通常端点 $T(n-m)$ (但し、 m は、自然数であり、 $n-m < 1$ の場合は、 $T(n-m+N)$) との距離を $L1$ とし、第 n の通常端点 $T(n)$ と第 $(n+1)$ の通常端点 $T(n+1)$ (但し、 $n=N$ の場合は第1の通常端点 $T(1)$: 以下同様) との間の距離を $L2$ とし、第 $(n+1)$ の通常端点 $T(n+1)$ と第 $(n+2)$ の通常端点 $T(n+2)$ (但し、 $n=N-1$ の場合は第1の通常端点 $T(1)$ 、 $n=N$ の場合は第2の通常端点 $T(2)$: 以下同様) との間の距離を $L3$ としたときに、

$L2$ が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ (但し、 $n=1$ の場合は第 N の通常端点 $T(N)$: 以下同様)、 $T(n)$ 、 $T(n+1)$ 、 $T(n+2)$ の4点が一直線上にある場合には、 $L1 < L3$ であれば通常端点 $T(n)$ を除去対象とし、 $L1 > L3$ であれば通常端点 $T(n+1)$ を除去対象とし、 $L1 = L3$ であれば通常端点 $T(n)$ もしくは通常端点 $T(n+1)$ のいずれかを除去対象とし、

$L2$ が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ 、 $T(n)$ 、 $T(n+1)$ の3点は一直線上にあるが、通常端点 $T(n+2)$ は当該一直線上にない場合には、通常端点 $T(n)$ を除去対象とし、

$L2$ が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n)$ 、 $T(n+1)$ 、 $T(n+2)$ の3点は一直線上にあるが、通常端点 $T(n-1)$ は当該一直線上にない場合には、通常端点 $T(n+1)$ を除去対象とする、

という処理を、 $n \geq N$ が満たされるまで、 n を1ずつ増加させながら繰り返す。

返し実行し、除去対象となった通常端点を除去する端点除去処理を行うことになる。

[0297] 端点再構築部 116 が実行する端点再構築の処理として、このような 4 点式の端点除去処理を採用した場合も、上述した 3 点式の端点除去処理を採用した場合とほぼ同様の結果が得られる。ただ、上述したように、4 点式の端点除去処理の方が 3 点式の端点除去処理に比べて、除去対象としてより好ましい端点を選択することができるので、除去処理によるシミュレーション精度低下の影響をより小さくすることができる。

[0298] (3) 端点再配置処理による再構築

これまで述べてきた 3 点式もしくは 4 点式の端点除去処理は、端点再構築部 116 によって、既存の通常端点を除去することにより端点の再構築を行うものであったが、ここで述べる端点再配置処理は、一部の輪郭線について、中間通常端点を一旦除去し、当該一部の輪郭線の各辺に新たな中間通常端点を配置する端点再配置処理により、通常端点の再構築を行うものである。

[0299] 図 45 は、図 32 に示す端点再構築部 116 による端点再構築処理として、端点再配置処理を採用した場合の手順を示す模式図である。いま、特定の単位図形の特定の輪郭線を構成する一辺に、図 45 (a) に示すように、通常端点 A～H が定義されているものとしよう（これらの通常端点は、通常端点定義部 115 によって定義されたものである）。ここでは、説明の便宜上、これら通常端点 A～H のうち、単位図形となる多角形の頂点を構成する通常端点 A、H を隅部通常端点と呼び、頂点を構成しない通常端点 B～G を中間通常端点と呼ぶことにする。

[0300] ここで述べる端点再配置処理による再構築を行う場合、端点再構築部 116 は、まず、この輪郭線上に定義されている中間通常端点を一旦除去する処理を行う。図 45 (b) は、図 45 (a) に示す通常端点 A～H から、中間通常端点 B～G を除去した状態を示している。なお、図 45 には、1 本の輪郭線を構成する多角形の上辺のみが示されているが、実際には、1 本の輪郭線は多角形の各辺に沿ったループを構成しており、輪郭線の個々の辺について、

中間通常端点を一旦除去する処理が行われる。図45(b)には、輪郭線の上辺についての中間通常端点を除去することにより、隅部通常端点A、Hのみが残された状態が示されている。

[0301] 続いて、端点再構築部116は、中間通常端点の除去を行った輪郭線の各辺上に所定間隔で中間通常端点を再配置することにより、端点再配置処理を行う。図45(c)、(d)、(e)は、いずれも上記輪郭線の上辺について中間通常端点を再配置した例を示している。より具体的には、図45(c)は、所定の基準間隔 L_{reg} を設定し、左端の隅部通常端点Aの位置を始点として、基準間隔 L_{reg} ごとに中間通常端点を再配置した例である。通常端点A～Eまでは、隣接端点の間隔は基準間隔 L_{reg} になるが、通常端点E～H間は端数間隔 L_{odd} になる。たとえば、辺AHの長さが45nmであり、基準間隔 $L_{reg}=10nm$ に設定した場合、端数間隔 $L_{odd}=5nm$ になる。

[0302] これに対して、図45(d)は、中央部分の通常端点C、D間を端数間隔 L_{odd} とし、通常端点A～B間、B～C間、D～E間、E～H間を基準間隔 L_{reg} とした例である。一般に、半導体デバイスのような微細な図形パターンの場合、多角形の頂点付近における露光時の誤差が大きくなる傾向にあり、本来は角ばっている形状が丸みを帯びる傾向にある。このような事情から、隅部通常端点とこれに隣接する中間通常端点との間隔はできるだけ離れた方が、シミュレーションの演算負担を軽減できる。このような観点から、端数間隔 L_{odd} は、図45(c)に示すように、隅部通常端点Hに隣接する位置に配置するよりも、図45(d)に示すように、辺AHの中央付近に配置した方が好ましい。辺AHの長さが45nmの場合、各端点の間隔は、左から順に、10, 10, 5, 10, 10nmということになる。

[0303] 図45(e)は、すべての端点間隔を等間隔 L_{even} に設定した例である。 L_{even} の値として予め最適値を定めておけば、当該最適値に最も近い等間隔 L_{even} が得られるように、辺AHを等分割した位置に中間通常端点を再配置することができる。たとえば、 L_{even} の最適値が10nmに設定されている場合、辺AHの長さが45nmであったとすると、これを4分割する場合は L_{even}

$n = 11$ 、 $2.5nm$ 、5分割する場合は $L_{even} = 9nm$ 、6分割する場合は $L_{even} = 7.5nm$ になるので、最適値の $10nm$ に最も近い値として、 $L_{even} = 9nm$ が得られる。そこで、図示のように、辺 AH を5分割して、等間隔 $L_{even} = 9nm$ おきに中間通常端点を再配置すればよい。

[0304] 図46は、図45に示す端点再配置処理を採用した場合の端点の再構築の基本手順を示す流れ図である。まず、ステップS31において、処理対象となる特定の輪郭線上の中間通常端点（頂点を構成しない通常端点）を一旦除去する処理が行われる。続いて、ステップS32において、当該輪郭線を構成する多角形の各辺に、新たな中間通常端点を配置する処理が行われる。具体的な方法は、図45(c)～(e)に示したとおりである。

[0305] (4) 端点除去処理と端点再配置処理の使い分け

これまで3点式もしくは4点式の端点除去処理（図41～図44）と端点再配置処理（図45～図46）とを説明した。ここで、端点除去処理は、任意の単位図形の任意の輪郭線に対して適用することができ、元図形パターン10に含まれるすべての単位図形のすべての輪郭線に適用してもかまわない。これに対して、端点再配置処理は、輪郭線によっては、適用すると不適切なケースがある。

[0306] これは、そもそも本発明は、単位図形を複数の分割図形に分割し、この分割図形の頂点位置に基づいて通常端点を定義することを特徴としているため、分割図形の頂点位置に基づいて定義された一部の通常端点を除去する処理（通常端点の数を減少させる処理）を行っても、「分割図形の頂点位置に基づいて定義された通常端点」という特徴が失われることはないが、通常端点を再配置する処理（位置を修正する処理）を行うと、再配置された中間通常端点については上記特徴が失われてしまうためである。

[0307] たとえば、図42に示す例では、白抜き三角形で示す通常端点D、Iが除去されることになるが、それ以外の各通常端点は、「図34(c)に示す4組の分割図形10a～10dの頂点位置に基づいて定義された通常端点」という特徴を有している。ここで、各通常端点が分割図形10a～10dの頂点

位置に基づく位置に定義されていると、図36、図37で説明したように、補正図形パターン15に基づく露光工程において、露光ビームのショット数を低減できるというメリットが得られる。これは、図38(d)に示す単位図形10の上辺に定義された中間通常端点B、C、Dの水平方向位置と、下辺に定義された中間通常端点I、H、Gの水平方向位置とが揃うためである。

[0308] ところが、図38(d)に示す単位図形10の例に対して端点再配置処理を行うと、隅部通常端点A、E、F、Jの位置は変わらないものの、中間通常端点B、C、D、G、H、Iの位置は修正され、4組の分割図形10a~10dの頂点位置とは無関係な位置に再配置されることになる。そうすると、単位図形10の上辺に定義された中間通常端点の水平位置と下辺に定義された中間通常端点の水平位置とが揃わない可能性があり、露光ビームのショット数を低減できるというメリットが得られなくなるおそれがある。

[0309] したがって、図42に示す単位図形や、図43に示す単位図形については、端点除去処理(図41~図44)に基づく端点の再配置を行うのが好ましい。これに対して、図40(f)に示すような口の字型の単位図形(いわゆる穴開きの単位図形)については、外側輪郭線O_{out}に対しては端点除去処理(図41~図44)に基づく端点の再配置を行い、内側輪郭線O_{in}に対しては端点再配置処理(図45~図46)に基づく端点の再配置を行うのが好ましい。

[0310] 図47は、図40(f)に示す口の字型の単位図形10の内側輪郭線O_{in}上の通常端点M~Tに対して、図46の手順に基づく端点再配置処理を適用して端点の再構築を行った例を示す平面図である。具体的には、内側輪郭線O_{in}を構成する矩形の下辺NQ上の中間通常端点O、Pと上辺MR上の中間通常端点S、Tとを一旦除去し、下辺NQ上には新たな中間通常端点Uを配置し、上辺MR上には新たな中間通常端点Vを配置したものである。

[0311] このような端点再配置処理を行うことにより、内側輪郭線O_{in}上の中間通常端点U、Vは、図40(b)に示す8組の分割図形10a~10hの頂点位置とは無関係な位置に配置されることになる。しかしながら、外側輪郭線O_o

utで囲まれた図形内部は露光領域になるのに対して、内側輪郭線O_{in}で囲まれた図形内部は非露光領域になるため、内側輪郭線O_{in}上の中間通常端点U, Vの位置が任意の位置に設定されていても、露光ビームのショット数を低減できるメリットに影響は及ばない。

[0312] その一方で、図47(a)に示す各中間通常端点の配置を見ればわかるとおり、外側輪郭線O_{out}上の中間通常端点B, C, E, F, H, I, K, Lは、必ず隅部通常端点A, D, G, Jからある程度離れた位置（基本的には、Wsplitだけ離れた位置）に定義されるのに対して、内側輪郭線O_{in}上の中間通常端点O, P, S, Tは、隅部通常端点M, N, Q, Rのごく近傍に定義される可能性がある。しかも、前述したように、多角形の頂点付近における露光時の誤差は大きくなり、本来は角ばっている形状が丸みを帯びる傾向にある。そのため、隅部通常端点とこれに隣接する中間通常端点との間隔はできるだけ離れた方が、シミュレーションの演算負担を軽減できる。

[0313] このような事情を踏まえると、外側輪郭線O_{out}上に定義された中間通常端点については、端点除去処理（図41～図44）に基づく端点の再配置を行い、各分割図形の頂点位置に応じた配置がそのまま維持されるようにし、露光ビームのショット数を低減できるメリットが享受できるようにするのが好ましい。これに対して、内側輪郭線O_{in}上に定義された中間通常端点については、端点再配置処理（図45～図46）に基づく端点の再配置を行い、隅部通常端点からできるだけ離れた位置に新たな中間通常端点が配置されるようにし、シミュレーションの演算負担を軽減できるメリットが享受できるようにするのが好ましい。

[0314] §6.2で述べたとおり、穴開きの単位図形10について、外側輪郭線O_{out}を構成する図形を「正図形」と呼び、内側輪郭線O_{in}を構成する図形を「反図形」と呼ぶことにすると、単位図形ベクトル群が輪郭線に沿って順方向（ここに述べる実施例の場合は時計まわりの方向）に一周する場合、当該輪郭線は単位図形10の外側輪郭線O_{out}を構成する正図形であると判断でき、単位図形ベクトル群が輪郭線に沿って反方向（ここに述べる実施例の場合は

反時計まわりの方向)に一周する場合、当該輪郭線は単位図形10の内側輪郭線O_{in}を構成する反図形であると判断できる。

[0315] そこで、穴開きの単位図形を取り扱う可能性がある場合は、端点再構築部116に、端点除去処理(図41～図44)と端点再配置処理(図45～図46)と、を単位図形の輪郭線ごとに選択的に実行させる機能をもたせるようにし、順方向まわりの単位図形ベクトルによって囲まれた正図形上の通常端点(図47(a)に示す例の場合は、外側輪郭線O_{out}上の通常端点)に対しては、端点除去処理により端点再構築を行い、順方向とは逆の逆方向まわりの単位図形ベクトルによって囲まれた反図形上の通常端点(図47(a)に示す例の場合は、内側輪郭線O_{in}上の通常端点)に対しては、端点再配置処理により端点再構築を行うようにすればよい。

[0316] <6.4 評価点の定義>

ここでは、図32に示す評価点設定ユニット110における評価点定義部117の機能について説明する。この評価点定義部117は、通常端点定義部115によって単位図形の輪郭線上に定義された通常端点に基づいて、もしくは、§6.3で述べた端点再構築部116を設ける実施形態の場合は、当該端点再構築部116による再構築によって得られた通常端点に基づいて、評価点Eを定義する構成要素である。すなわち、評価点定義部117は、個々の単位図形の輪郭線上において互いに隣接して配置された2つの通常端点の間に、それぞれ評価点を定義する処理を行う。

[0317] 個々の評価点Eを定義する位置は、隣接する一对の通常端点の間であれば、原理的には、どの位置であってもかまわないが、実用上は、一对の通常端点の midpoint 位置に定義するのが好ましい。これは、一对の通常端点によって挟まれる輪郭線分についてのずれ量(プロセスバイアス)をシミュレーションによって求める際に、当該輪郭線分の中心点に評価点を設定した方が、より適切なシミュレーション結果が得られると考えられるためである。

[0318] そこで、ここで述べる実施形態では、評価点定義部117は、単位図形の輪郭線上において互いに隣接して配置された2つの通常端点の midpoint 位置にそ

れぞれ評価点を定義する処理を行う。図48(a)は、図42(b)に示す端点再構築後の平行四辺形型の単位図形10上の通常端点(x印)を示し(各端点の符号はA~Hに付け直されている)、図48(b)は、これらの通常端点A~Hに基づいて、評価点1~8(黒丸)を定義した例を示す。具体的には、一对の通常端点A-Bに挟まれる輪郭線分AB上には、その中心位置に評価点1が定義されており、一对の通常端点B-Cに挟まれる輪郭線分BC上には、その中心位置に評価点2が定義されており、……、一对の通常端点H-Aに挟まれる輪郭線分HA上には、その中心位置に評価点8が定義されている。

[0319] 同様に、図49(a)は、図43(b)に示す端点再構築後のL型の単位図形10上の通常端点(x印)を示し(各端点の符号はA~Hに付け直されている)、図49(b)は、これらの通常端点A~Hに基づいて、評価点1~8(黒丸)を定義した例を示す。いずれの評価点も、各輪郭線分の中点位置に定義されている。また、図50(a)は、図47(b)に示す端点再構築後の口の字型の単位図形10上の通常端点(x印)を示し(各端点の符号はA~Rに付け直されている)、図50(b)は、これらの通常端点A~Rに基づいて、評価点1~18(黒丸)を定義した例を示す。やはり、いずれの評価点も、各輪郭線分の中点位置に定義されている。

[0320] こうして定義された各評価点は、図3に示す評価点E11, E12, E13に対応するものであり、図1に示す特徴量抽出ユニット120によって、これら各評価点についての特徴量 $x_1 \sim x_n$ が抽出され、バイアス推定ユニット130によって、これら各評価点についてのプロセスバイアスの推定値 y が求められる。そして、パターン補正ユニット140によって、個々の評価点について求められたプロセスバイアスの推定値 y に基づいて、当該評価点を含む輪郭線分の位置が補正される。

[0321] <6.5 補充端点の定義>

続いて、図32に示す評価点設定ユニット110における補充端点定義部118の機能について説明する。この補充端点定義部118は、個々の単位

図形について、多角形の頂点を構成しない通常端点である中間通常端点を探索し、探索された中間通常端点と同じ位置に補充端点を追加定義する機能を実行する。前述したとおり、通常端点定義部 115 によって定義された通常端点や、端点再構築部 116 による再構築によって得られた通常端点は、多角形の頂点を構成する隅部通常端点と多角形の頂点を構成しない中間通常端点とに分けられる。補充端点定義部 118 は、このうち、中間通常端点について、同じ位置に補充端点を追加定義することになる。

[0322] ここでは、まず、通常端点に加えて、補充端点を追加定義する必要性について説明する。上述したように、評価点定義部 117 による評価点定義処理が完了すると、隣接する一对の通常端点で挟まれた輪郭線分のそれぞれについて、中点位置に評価点が定められる。たとえば、図 48(b) に示す例の場合、一对の通常端点 A-B で挟まれた輪郭線分 AB 上に評価点 1 が定義され、一对の通常端点 B-C で挟まれた輪郭線分 BC 上に評価点 2 が定義される。そして、各評価点について、特徴量が抽出され、プロセスバイアスの推定値 y が求められ、当該プロセスバイアスの推定値 y に基づいて、当該評価点を含む輪郭線分の位置が補正される。

[0323] たとえば、図 48(b) に示す例において、シミュレーションの結果、評価点 1 についてはプロセスバイアスの推定値 $y(1) = -1 \text{ nm}$ が求められ、評価点 2 についてはプロセスバイアスの推定値 $y(2) = -2 \text{ nm}$ が求められたものとしよう。この場合、図 48(b) に示す平行四辺形状の元図形パターン 10 をそのまま用いてリソグラフィプロセスを実行した場合、実基板 S 上に得られる実図形パターン 20 上の評価点 1 に対応する点は 1 nm 内側にずれた位置になり、評価点 2 に対応する点は 2 nm 内側にずれた位置になることが、シミュレーションによって推定されたことになる。そこで、パターン補正ユニット 140 は、たとえば、評価点 1 を有する輪郭線分 AB を外側に 1 nm 移動させ、評価点 2 を有する輪郭線分 BC を外側に 2 nm 移動させる処理を行って、補正図形パターン 15 を作成する補正処理を行うことになる。

[0324] このとき、輪郭線分 A B と輪郭線分 B C とを別個独立して移動させることは、幾何学的な概念的としては容易に理解できよう。しかしながら、コンピュータ上のデータ処理としては、中間通常端点 B が単一の端点のままでは不都合が生じることになる。本発明では、図 3 (b) の例に示すように、各評価点を輪郭線に直交する方向に移動させる補正が行われる。したがって、輪郭線分 A B と輪郭線分 B C とをそれぞれ別個の補正量に応じて移動させるためには、輪郭線分 A B の右端としての役割を果たす通常端点 B と、輪郭線分 B C の左端としての役割を果たす通常端点 B とは、データ処理上、別個の端点として取り扱う必要が生じる。このようなコンピュータ処理の事情から、本発明では、中間通常端点 B について、全く同じ位置に補充端点 B' を追加定義するようにしている。こうして追加定義された補充端点 B' は、一方の輪郭線分のみの端点として取り扱うことができ、既存の中間通常端点 B は、他方の輪郭線分のみの端点として取り扱うことができる。

[0325] たとえば、図 4 8 (b) に示す平行四辺形状の単位図形の場合、中間通常端点 B, C, F, G について、それぞれ同じ位置に補充端点 B', C', F', G' を追加定義すれば、図 5 1 (a) に示すように、8 個の通常端点 A ~ H (×印) に加えて、更に 4 個の補充端点 B', C', F', G' (四角印) が追加され、合計 12 個の端点が定義されることになる。そうすれば、たとえば、輪郭線分 A B の左端を通常端点 A、右端を補充端点 B' とし、輪郭線分 B C の左端を通常端点 B、右端を補充端点 C' とし、輪郭線分 C D の左端を通常端点 C、右端を通常端点 D とし、輪郭線分 D E の上端を通常端点 D、下端を通常端点 E とし、輪郭線分 E F の右端を通常端点 E、左端を補充端点 F' とし、輪郭線分 F G の右端を通常端点 F、左端を補充端点 G' とし、輪郭線分 G H の右端を通常端点 G、左端を通常端点 H とし、輪郭線分 H A の下端を通常端点 H、上端を通常端点 A とすることができる。

[0326] このようにして 8 組の輪郭線分の両端を定義すれば、たとえば、輪郭線分 A B と輪郭線分 B C とを別個独立して移動させることができる (コンピュータ処理上、通常端点 B と補充端点 B' とを別な座標位置に移動させることが

できる)。なお、補充端点の追加定義を、中間通常端点についてのみ行う理由は、1つの中間通常端点は、上述したように、補正後に異なる2つの位置を占める2つの端点に分離する可能性があるのに対して、1つの隅部通常端点は補正後も1つの隅部通常端点として残るためである。このような事情は、§ 7において、具体例を示しながら詳述する。

[0327] 同様に、図49(b)に示すL字状の単位図形の場合、中間通常端点F、Hについて、それぞれ同じ位置に補充端点F'、H'を追加定義すれば、図51(b)に示すように、8個の通常端点A~H(x印)に加えて、更に2個の補充端点F'、H'(四角印)が追加され、合計10個の端点が定義されることになる。そうすれば、たとえば、輪郭線分EFの右端を通常端点E、左端を補充端点F'とし、輪郭線分FGの右端を通常端点F、左端を通常端点Gとすることができ、また、輪郭線分GHの下端を通常端点G、上端を補充端点H'とし、輪郭線分HAの下端を通常端点H、上端を通常端点Aとすることができ、各輪郭線分を別個独立して移動させる処理が可能になる。

[0328] また、図50(b)に示す口の字状の単位図形の場合、中間通常端点B、C、E、F、H、I、K、L、O、Rについて、それぞれ同じ位置に補充端点B'、C'、E'、F'、H'、I'、K'、L'、O'、R'を追加定義すれば、図51(c)に示すように、18個の通常端点A~R(x印)に加えて、更に10個の補充端点B'、C'、E'、F'、H'、I'、K'、L'、O'、R'(四角印)が追加され、合計28個の端点が定義されることになる。そうすれば、たとえば、輪郭線分ABの左端を通常端点A、右端を補充端点B'とし、輪郭線分BCの左端を通常端点B、右端を補充端点C'とし、輪郭線分CDの左端を通常端点C、右端を通常端点Dとし、……というようにすることができ、各輪郭線分を別個独立して移動させる処理が可能になる。

[0329] 図52は、図32に示す補充端点定義部118によって行われる補充端点定義処理の具体的な手順を示す模式図(図(a)、(b))および流れ図(図(c))である。上述したとおり、補充端点定義部118は、個々の単位図形につ

いて中間通常端点を探索し、探索された中間通常端点と同じ位置に補充端点を追加定義する処理を行う。ここで、中間通常端点の探索処理は、具体的には次のような手順で行うことができる。

[0330] いま、処理対象となる特定の単位図形の特定の輪郭線上に合計 N 個の通常端点が定義されているものとする。これら N 個の通常端点のうち、一部は多角形の頂点を構成する隅部通常端点であり、残りの一部は多角形の頂点を構成しない中間通常端点である。ここで行う探索処理は、 N 個の通常端点の中から中間通常端点を見つける処理ということになる。

[0331] そこで、この全 N 個の通常端点を、輪郭線に沿った順に、それぞれ第1の通常端点 $T(1)$ ～第 N の通常端点 $T(N)$ とし、図52(a)に示すように、第 n の通常端点 $T(n)$ を判定対象として、中間通常端点か否かの判定を行う。この判定には、図示のとおり、第 $(n-1)$ の通常端点 $T(n-1)$ 、第 n の通常端点 $T(n)$ 、第 $(n+1)$ の通常端点 $T(n+1)$ の3点を用いる。図の Oa は、2つの端点 $T(n-1)$ 、 $T(n)$ を結ぶ輪郭線分であり、図の Ob は、2つの端点 $T(n)$ 、 $T(n+1)$ を結ぶ輪郭線分である。そして、第 n の通常端点 $T(n)$ が中間通常端点と判定された場合には、図52(b)に示すように、当該第 n の通常端点 $T(n)$ と同じ位置に補充端点 $T(n)'$ を追加定義する。図では、通常端点 $T(n)$ を $\times$ 印、補充端点 $T(n)'$ を四角印で示している。

[0332] 第 n の通常端点 $T(n)$ が中間通常端点であった場合に、同じ位置に補充端点 $T(n)'$ を追加定義すれば、輪郭線分 Oa の左端を通常端点 $T(n-1)$ とし、右端を補充端点 $T(n)'$ とすることができる。また、輪郭線分 Ob の左端を通常端点 $T(n)$ とし、右端を通常端点 $T(n+1)$ とすることができる。端点 $T(n)$ と $T(n)'$ とは別個独立して移動させることができるので、輪郭線分 Oa と Ob とを別個独立して移動させることができるようになり、パターン補正ユニット140によって補正処理を行う際に支障は生じない。

[0333] 図52(c)は、補充端点定義部118によって行われる補充端点定義処理

の具体的な手順を示す流れ図である。まず、ステップSS41において、中間通常端点であるか否かの判定対象となる通常端点の番号を示すパラメータ n を $n = 1$ に設定し、ステップSS42において、第 $(n - 1)$ の通常端点 $T(n - 1)$ 、第 n の通常端点 $T(n)$ 、第 $(n + 1)$ の通常端点 $T(n + 1)$ の3点が一直線上にあるか否かが判定される。ここで、肯定的な判定がなされたら、判定対象となる端点 $T(n)$ は、中間通常端点（多角形の頂点を構成しない端点）ということになるので、ステップSS43において、この中間通常端点 $T(n)$ と同じ位置に、補充端点 $T(n)'$ が追加定義される。ステップSS42において否定的な判定がなされたら、判定対象となる端点 $T(n)$ は、隅部通常端点（多角形の頂点を構成する端点）ということになるので、補充端点 $T(n)'$ の追加定義は行われない。

[0334] このような処理が、ステップSS44およびSS45を介して、パラメータ n を1ずつ増加させながら、 $n = N$ に到達するまで繰り返し実行される。そして、 $n = N$ に到達したら、補充端点定義処理は完了である。このような処理を行うことにより、図51(a), (b), (c) に四角印で示す補充端点が定義されることになる。

[0335] 一般論で述べると、補充端点定義部118は、1つの輪郭線について定義されている合計 N 個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1)$ ～第 N の通常端点 $T(N)$ と呼んだときに、第 n の通常端点 $T(n)$ に着目して、第 $(n - 1)$ の通常端点 $T(n - 1)$ （但し、 $n = 1$ の場合は第 N の通常端点 $T(N)$ ）、第 n の通常端点 $T(n)$ 、第 $(n + 1)$ の通常端点 $T(n + 1)$ （但し、 $n = N$ の場合は第1の通常端点 $T(1)$ ）の3点が一直線上にある場合に、第 n の通常端点 $T(n)$ を中間通常端点と判定し、この中間通常端点と同じ位置に補充端点を追加定義する処理を、 $n = 1 \sim N$ まで繰り返し実行すればよい。

[0336] <<< § 7. パターン補正ユニットの詳細 >>>

続いて、パターン補正ユニット140について詳細な説明を行う。図53は、図1に示す本発明の基本的実施形態に係る図形パターンの形状補正装置

１００内のパターン補正ユニット１４０およびこれに関連する構成要素を示すブロック図である。パターン補正ユニット１４０の基本機能は、評価点設定ユニット１１０が設定した各評価点Ｅの位置を、バイアス推定ユニット１３０から出力されたプロセスバイアスの推定値 y に基づいて修正することにより、元図形パターン１０に対する補正を行い、補正図形パターン１５を作成することにある。

[0337] ここで、評価点設定ユニット１１０の機能は、§６で詳述したとおりであり、元図形パターン１０を構成する各単位図形について、通常端点Ｔ，補充端点Ｔ'，評価点Ｅを定義する処理が行われる。これら各点の情報は、輪郭線分Ｏの情報ということができる。すなわち、個々の通常端点Ｔおよび個々の補充端点Ｔ'の位置情報は、特定の輪郭線分Ｏの両端の位置を示す情報であり、評価点Ｅの位置情報は、当該輪郭線分Ｏ上に定義された評価点の位置を示す情報である。パターン補正ユニット１４０は、この輪郭線分Ｏの情報に基づいて、輪郭線分Ｏに対する移動処理および伸縮処理、ならびに新たな輪郭線分の追加処理を行うことにより、補正図形パターン１５を作成する。

[0338] 図５３に示すとおり、パターン補正ユニット１４０は、輪郭線分移動処理部１４１、輪郭線分追加処理部１４２、輪郭線分伸縮処理部１４３を有している。以下、これらの各構成要素の機能を、図５１(b)に示すＬ字型の単位図形を例にとって具体的に説明する。なお、上述したとおり、本発明に係る図形パターンの形状補正装置１００は、実際には、コンピュータに専用のプログラムを組み込むことにより構成される。したがって、パターン補正ユニット１４０に含まれる上記各構成要素１４１，１４２，１４３も、実際には、コンピュータに専用のプログラムを組み込むことにより構成される。

[0339] < ７．１ 輪郭線分移動処理 >

ここでは、まず、図５３に示す輪郭線分移動処理部１４１の機能について説明する。図示のとおり、輪郭線分移動処理部１４１には、元図形パターン１０の情報とともに、評価点設定ユニット１１０によって定義された輪郭線分Ｏの情報（通常端点Ｔ，補充端点Ｔ'，評価点Ｅの位置座標およびこれら

の相互関係を示す情報）が与えられる。また、バイアス推定ユニット130からは、各評価点Eについて、プロセスバイアスの推定値 y が与えられる。

[0340] 図54は、図51(b)に示すL字型の単位図形上に定義された各端点および各評価点を示す拡大平面図である。この図では、各通常端点A～Hを×印で示し、各補充端点F'、H'を四角印で示し、各評価点1～8を丸印で示し、各評価点の符号1～8を当該丸印内に記述している。また、各隣接端点によって挟まれた輪郭線分については、O1～O8の符号を付して示している。たとえば、通常端点A－Bによって挟まれた部分は輪郭線分O1であり、通常端点B－Cによって挟まれた部分は輪郭線分O2である。8個の通常端点A～Hのうち、6個の通常端点A、B、C、D、E、Gは、多角形の頂点を構成する隅部通常端点であり、残りの2個の通常端点F、Hは、多角形の頂点を構成しない中間通常端点である。この2個の中間通常端点F、Hについては、前述したとおり、それぞれ補充端点F'、H'が同じ位置に定義されている。

[0341] 図55は、図54に示す各端点および各評価点の相互関係を表すデータフォーマットの一例を示す図である。上述したとおり、評価点設定ユニット110から輪郭線分移動処理部141に対しては、輪郭線分Oの情報が与えられるが、この輪郭線分Oの情報は、具体的には、図55に示すデータフォーマットで与えられることになる。このデータフォーマットでは、1つの端点と1つの評価点とを対応させたデータ対（図では、括弧で囲まれている）になっており、これらのデータ対が単位図形の輪郭線に沿って、時計まわりの順に並んでいる。

[0342] たとえば、1番目のデータ対である（A，1）は、端点Aと評価点1とを対応づけたデータ対であり、端点Aを始端、端点Bを終端とする輪郭線分O1上に評価点1が定義されていることを示している。同様に、2番目のデータ対である（B，2）は、端点Bと評価点2とを対応づけたデータ対であり、端点Bを始端、端点Cを終端とする輪郭線分O2上に評価点2が定義されていることを示している。

[0343] なお、第6番目のデータ対である(F' , Φ)は、補充端点 F' と評価点 Φ とを対応づけたデータ対である。ここで、評価点 Φ は実際には存在しない評価点であり、データ「 Φ 」は、補充端点 F' をデータ対という形式で表現する便宜のために設けられた空データである。同様に、第9番目のデータ対である(H' , Φ)も、補充端点 H' と評価点 Φ とを対応づけたデータ対であり、データ「 Φ 」は空データである。

[0344] 図55に示すデータフォーマットを採用すれば、通常端点と補充端点とをひっくるめたすべての端点を、それぞれデータ対として表現することができる。もちろん、各データ対を構成するデータには、各端点や各評価点の位置座標値が含まれているので、各端点や各評価点の二次元平面上における位置を特定することができる。しかも、各データ対は、単位図形の輪郭線に沿った時計まわりの順に羅列されているため、隣接端点で挟まれた輪郭線分を認識することができ、当該輪郭線分上に定義された評価点を認識することができる。たとえば、(A , 1), (B , 2)なる2組のデータ対によって、隣接端点 $A-B$ に挟まれる輪郭線分 $O1$ の位置を特定することができ、また、当該輪郭線分 $O1$ 上の評価点1の位置を特定することができる。

[0345] 図51(c)に示すような穴開きの単位図形の場合、外側輪郭線 O_{out} を構成する正図形と、内側輪郭線 O_{in} を構成する反図形とが形成されるが、正図形については、上例と同様に輪郭線に沿った時計まわりの順に各データ対を羅列し、反図形については、上例とは逆に輪郭線に沿った反時計まわりの順に各データ対を羅列するのが好ましい。そうすれば、各データ対の羅列順序に基づいて、個々の輪郭が外側輪郭線 O_{out} か、内側輪郭線 O_{in} かを識別することができるようになり、後述する輪郭線分の移動処理を行う際に、単位図形の内側と外側とを認識することができる。

[0346] なお、図53に示す実施例では、輪郭線分移動処理部141に対して、元図形パターン10を与えているが、輪郭線分移動処理を行う上では、評価点設定ユニット110から輪郭線分移動処理部141に対して図55に示すデータを与え、バイアス推定ユニット130から輪郭線分移動処理部141に

対して各評価点についてのプロセスバイアスの推定値 y を与えれば十分である。すなわち、輪郭線分移動処理部 141 にとって元図形パターン 10 は必須の情報ではない。ただ、元図形パターン 10 は、単位図形の形状を直接的に示す情報であるので、実用上は、これを輪郭線分移動処理部 141 に与え、輪郭線分移動処理を行う上での参考に供するのが好ましい。

[0347] 図 56 は、輪郭線分移動処理部 141 によって、図 54 に示す L 字型の単位図形 10 に対して輪郭線分の移動処理が行われた状態を示す平面図である。図に破線で示す輪郭線分は移動処理前の状態を示し、図に実線で示す輪郭線分は移動処理後の状態を示している。輪郭線分の移動処理は、個々の評価点について、単位図形の輪郭線に沿って当該評価点の一方の側に隣接する通常端点もしくは補充端点を第 1 の端点とし、他方の側に隣接する通常端点もしくは補充端点を第 2 の端点とする輪郭線分を定義し、個々の評価点についての輪郭線分を、当該評価点のプロセスバイアスの推定値に応じた補正量だけ、輪郭線分に直交する方向に移動させることによって行われる。このとき、1 つの輪郭線分の各端点が他の輪郭線分の各端点として重複して利用されないように、個々の輪郭線分についてそれぞれ通常端点もしくは補充端点のいずれか一方を選択する。

[0348] 図 56 に示す例は、各評価点 1～8 のプロセスバイアスの推定値 y が負の値をとった例（リソグラフィプロセスのシミュレーションにより、元の単位図形 10 の輪郭線が内側にずれてしまうという結果が出た例）についての補正処理を示している。すなわち、元の単位図形 10 をそのまま用いてリソグラフィプロセスを実行すると、この元の単位図形 10 よりも縮小した実図形 20 が得られることになるので、元の単位図形 10 を若干拡大した補正後の単位図形 15 を得るための補正処理が行われる。

[0349] 図 56 に示す太線矢印は、各評価点の移動方向を示している。ここに示す例では、評価点 1～8 のプロセスバイアスの推定値 y が負の値をとっているため、評価点 1～8 はいずれも単位図形の外側方向に移動させられることになる。たとえば、評価点 1 のプロセスバイアスの推定値が $y(1) = -1n$

mであった場合、評価点1を図の太線矢印で示すように1nmだけ外側に移動させる処理が行われる。前述したように、本発明では、評価点の移動は、当該評価点を含む輪郭線分に直交する方向に行われる。しかも、当該評価点を含む輪郭線分ごと、評価点の移動が行われる。したがって、図示の例の場合、評価点1を太線矢印で示すように1nmだけ外側に移動させるとともに、評価点1を含む輪郭線分O1も同じように平行移動させることになる。

[0350] 図56に示す例では、同様に、輪郭線分O2～O8が外側に平行移動させられている。ここで留意すべき点は、中間通常端点F、Hと、これらと同じ位置に追加定義された補充端点F'、H'の挙動である。図示の例では、評価点5についてのプロセスバイアスの推定値yと、評価点6についてのプロセスバイアスの推定値yとが異なる値となっているため、輪郭線分O5の移動量と輪郭線分O6の移動量も異なる値になる。このため、輪郭線分O5、O6は、移動方向が同じであっても移動量が異なるため、図示のとおり、移動処理後は、両者が分離した状態になる。ただ、輪郭線分O5の両端は端点E、F'であり、輪郭線分O6の両端は端点F、Gであるため、端点F、F'の位置が異なることになっても、処理上、問題は生じない。

[0351] 輪郭線分O7、O8についても同様である。図示のとおり、輪郭線分O7と輪郭線分O8は移動量が異なっているため、移動処理後、両者は分離した状態になる。ただ、輪郭線分7の両端は端点G、H'であり、輪郭線分8の両端は端点H、Aであるため、端点H、H'の位置が異なることになっても、処理上、問題は生じない。すなわち、この移動処理におけるコンピュータ上での実体的な処理は、図55に示すデータの修正ということになるが、図55に示すデータフォーマットの構造自体には何ら修正を加える必要はなく、各端点や各評価点の座標値を修正する処理を行うだけで済む。補充端点定義部118によって、中間通常端点F、Hについて補充端点F'、H'を追加定義したのは、このような便宜を図るためである。

[0352] 一方、隅部通常端点A、B、C、D、E、Gについては、補充端点の追加定義はなされていない。そのため、これら隅部通常端点は、図示のとおり、

移動処理後に異なる2つの位置に分離してしまっている。たとえば、隅部通常端点Bは、移動処理後に、輪郭線分O1の右端の位置と輪郭線分O2の上端位置とに分離している。このように移動処理後に2つに分離した隅部通常端点の取り扱いは、後述する輪郭線分伸縮処理部143によってなされる。

[0353] <7. 2 輪郭線分追加処理>

続いて、図53に示す輪郭線分追加処理部142の機能について説明する。この輪郭線分追加処理部142は、移動処理前に互いに同じ位置にあった中間通常端点および補充端点が、移動処理後に異なる2つの位置に分離した場合、それぞれの移動処理後の位置を連結する新たな輪郭線分を追加する追加処理を行う機能を有している。

[0354] 図56に示す例の場合、輪郭線分の移動処理によって、中間通常端点F，Hと補充端点F'，H'とは、それぞれ異なる位置に配置されることになる。このため、中間通常端点Fと補充端点F'との間には隙間が生じ、中間通常端点Hと補充端点H'との間にも隙間が生じた状態になる。輪郭線分追加処理部142は、この隙間を埋めるために新たな輪郭線分を追加する処理を行う。

[0355] 図57は、輪郭線分追加処理部142によって、図56に示す移動処理後の単位図形に対して輪郭線分の追加処理が行われた状態を示す平面図である。図に太線で示す部分が、新たに追加された輪郭線分である。具体的には、移動処理前に互いに同じ位置にあった中間通常端点Fと補充端点F'とについて、移動処理後の位置を連結する新たな輪郭線分O5'が追加され、移動処理前に互いに同じ位置にあった中間通常端点Hと補充端点H'とについて、移動処理後の位置を連結する新たな輪郭線分O7'が追加される。

[0356] もっとも、ここで述べる実施例の場合、前述したように、図55に例示するデータフォーマットで単位図形を構成する各端点や各評価点の情報を表現しているため、実質的には、このようなデータフォーマットで示される各データ対における端点や評価点の座標値を修正する処理（前述した輪郭線分移動処理）を行うことにより、輪郭線分追加処理も自動的に行われることにな

る。

[0357] たとえば、図55における第6番目のデータ対(F' , Φ)は、図57において新たに追加された輪郭線分 $O5'$ に対応するデータになっている。具体的には、データ対(F' , Φ)におけるデータ「 F' 」は輪郭線分 $O5'$ の始端である端点 F' の座標値を示す情報であり、データ「 Φ 」は輪郭線分 $O5'$ についての評価点の位置を示す情報である。実際には、データ「 Φ 」は空データであるので、輪郭線分 $O5'$ は評価点をもたない輪郭線分ということになる。なお、輪郭線分 $O5'$ の終端である端点 F の座標値は、第7番目のデータ対(F , 6)におけるデータ「 F 」によって示されている。

[0358] 同様に、図55における第9番目のデータ対(H' , Φ)は、図57において新たに追加された輪郭線分 $O7'$ に対応するデータになっている。具体的には、データ対(H' , Φ)におけるデータ「 H' 」は輪郭線分 $O7'$ の始端である端点 H' の座標値を示す情報であり、データ「 Φ 」は輪郭線分 $O7'$ についての評価点の位置を示す情報である。実際には、データ「 Φ 」は空データであるので、輪郭線分 $O7'$ は評価点をもたない輪郭線分ということになる。なお、輪郭線分 $O7'$ の終端である端点 H の座標値は、第10番目のデータ対(H , 8)におけるデータ「 H 」によって示されている。

[0359] 結局、図55に例示するデータフォーマットを利用すれば、輪郭線分移動処理部141による移動処理を行うことにより、輪郭線分追加処理部142による追加処理も自動的に行われることになる。すなわち、移動処理前に同じ位置に定義されていた端点 F , F' や端点 H , H' が異なる位置に移動し、図55に示す各データ対において、これら端点 F , F' , H , H' の座標値が移動処理後の新たな座標値に書き換えられた時点で、端点 F , F' を連結する新たな輪郭線分 $O5'$ が自動的に追加され、端点 H , H' を連結する新たな輪郭線分 $O7'$ が自動的に追加されることになる。したがって、図55に例示するデータフォーマットを利用する場合、輪郭線分追加処理部142による追加処理は、輪郭線分移動処理部141による移動処理によって実現されることになり、データ処理上は、新たなデータ処理作業を行う必要は

ない。

[0360] < 7. 3 輪郭線分伸縮処理 >

次に、図 5 3 に示す輪郭線分伸縮処理部 1 4 3 の機能について説明する。

§ 7. 1 で述べたように、移動処理を行うと、隅部通常端点は 2 つの位置に分離してしまう。たとえば、図 5 7 に示す例の場合、隅部通常端点 A（図では、A 0, A 1, A 2 と示されている）、B, C, D, E, G は、移動処理後にいずれも 2 つの位置に分離している。ここで述べる輪郭線分伸縮処理は、このように分離した 2 つの隅部通常端点を融合させて、新たな隅部通常端点を作成することを目的としている。

[0361] そのため、輪郭線分伸縮処理部 1 4 3 は、移動処理前に多角形の頂点を構成していた通常端点である隅部通常端点が、移動処理後に異なる 2 つの位置に分離した場合、分離後の 2 つの隅部通常端点が融合して新たな隅部通常端点となるように、分離後の隅部通常端点を含む各輪郭線分をそれぞれ伸縮する伸縮処理を行う。

[0362] 具体的には、ここで述べる実施例の場合、輪郭線分伸縮処理部 1 4 3 は、ある 1 つの隅部通常端点が移動処理により第 1 の移動端点と第 2 の移動端点とに分離した場合に、第 1 の移動端点を一方の端点とする第 1 の移動後輪郭線分を含む直線と第 2 の移動端点を一方の端点とする第 2 の移動後輪郭線分を含む直線との交点を求め、当該交点を新たな隅部通常端点とし、第 1 の移動後輪郭線分および第 2 の移動後輪郭線分に対して一方の端点が新たな隅部通常端点となるような伸縮処理を行う。

[0363] ここでは、図 5 7 に示す例における隅部通常端点 A に着目して、上記伸縮処理の具体的な手順を説明する。ここでは、図 5 7 に示すように、移動処理前の隅部通常端点 A を符号 A 0 で示し、この隅部通常端点 A 0 が、移動処理後に第 1 の移動端点 A 1 と第 2 の移動端点 A 2 とに分離した場合を考える。この場合、第 1 の移動端点 A 1 を一方の端点とする第 1 の移動後輪郭線分 O 8 を含む直線（図では一点鎖線で示す）と第 2 の移動端点 A 2 を一方の端点とする第 2 の移動後輪郭線分 O 1 を含む直線（図では一点鎖線で示す）との

交点 J を求める。そして、この交点 J を新たな隅部通常端点 A とし、第 1 の移動後輪郭線分 O 8 および第 2 の移動後輪郭線分 O 1 に対して、一方の端点が新たな隅部通常端点 A となるような伸縮処理を行う。

[0364] 図 5 8 は、図 5 7 に示す単位図形に対して輪郭線分の伸縮処理が行われた状態を示す平面図である。第 1 の移動後輪郭線分 O 8 は、その上端が新たな隅部通常端点 A となるように伸ばされ、第 2 の移動後輪郭線分 O 1 は、その左端が新たな隅部通常端点 A となるように伸ばされている。かくして、図 5 7 に示す 2 つの移動端点 A 1, A 2 は、新たな隅部通常端点 A に融合されたことになる。残りの隅部通常端点 B, C, D, E, G についても同様の伸縮処理を行えば、図 5 8 に実線で示すような補正後の単位図形（補正図形パターン 1 5）が得られる。

[0365] 図示のとおり、補正後の単位図形（実線で示す補正図形パターン 1 5）は、補正前の単位図形（破線で示す元図形パターン 1 0）に比べてひとまわり大きな図形になっている。ここで、元図形パターン 1 0 に基づくシミュレーションの結果が正しく、当該シミュレーション結果に基づいて適切な補正が行われていれば、この補正図形パターン 1 5 を用いて実際のリソグラフィプロセスを行うと、実基板 S 上には、破線で示す元図形パターン 1 0 と同じ実図形パターン 2 0 が形成されることになる。

[0366] この補正図形パターン 1 5（実線）は、結局、元図形パターン 1 0（破線）に対して、§ 7. 1 で述べた輪郭線分の移動処理、§ 7. 2 で述べた輪郭線分の追加処理、§ 7. 3 で述べた輪郭線分の伸縮処理を行った後の輪郭線分の集合体を輪郭線とする単位図形によって構成されるパターンということになる。もっとも、元図形パターン 1 0 を構成する単位図形を表現する上で、図 5 5 に示すデータフォーマットのデータを用いた場合、補正図形パターン 1 5 を構成する単位図形も同じく図 5 5 に示すデータフォーマットのデータで表現されることになる。元図形パターン 1 0 を示すデータと補正図形パターン 1 5 を示すデータとの相違は、図 5 5 に示す各データ対に含まれる端点および評価点の位置情報（座標値）のみである。

[0367] < 7. 4 補正処理の繰り返し >

§ 7. 3 では、図 5 8 に破線で示す元図形パターン 1 0 に基づいて、パターン補正ユニット 1 4 0 による補正を施すことにより、図 5 8 に実線で示す補正図形パターン 1 5 を得る例を説明した。そして、正しいシミュレーション結果に基づいて適切な補正が施されていれば、補正図形パターン 1 5 を用いて実際のリソグラフィプロセスを行うことにより、実基板 S 上に、元図形パターン 1 0 と同じ実図形パターン 2 0 が形成される旨の説明を行った。

[0368] しかしながら、実際には、必ずしも適切な補正を 1 回で行うことは困難である。たとえば、§ 7. 1 の「輪郭線分移動処理」では、ある評価点のプロセスバイアスの推定値が y が -1 nm であった場合、当該評価点を $+1 \text{ nm}$ だけ移動させる補正処理を例示したが、このような補正処理が必ずしも適切な補正処理であるとは限らず、実際には、当該評価点を $+1.1 \text{ nm}$ だけ移動させる補正処理が最適な補正処理であるケースもあり得る。

[0369] 図 1 には、元図形パターン 1 0 を形状補正装置 1 0 0 に与えることにより、補正図形パターン 1 5 が得られる例が示されているが、通常、こうして得られた補正図形パターン 1 5 を用いてリソグラフィプロセスを実行し、実基板 S 上に実図形パターンを形成しても、得られる実図形パターン 2 0 は、設計当初の元図形パターン 1 0 には正確には一致しない。これは、元図形パターン 1 0 に含まれる図形と補正図形パターン 1 5 に含まれる図形とでは、サイズや形状が異なるため、リソグラフィプロセスを実行した場合の近接効果などの影響にも相違が生じるためである。

[0370] このため、実際には、図 1 に示すとおり、パターン補正ユニット 1 4 0 から出力された補正図形パターン 1 5 を、再び、図形パターンの形状補正装置 1 0 0 に与える処理を繰り返し行う必要がある。すなわち、補正図形パターン 1 5 は、形状補正装置 1 0 0 に元図形パターンの代わりに与えられ、この補正図形パターン 1 5 について再度の補正処理を行い、新たな補正図形パターンを得て、この新たな補正図形パターンを再び形状補正装置 1 0 0 に与えて、更に新たな補正図形パターンを得る、という処理が繰り返される。

[0371] このような補正処理の繰り返しは、図4の流れ図にも示されている。すなわち、ステップS2～S5の補正処理は、最初はステップS1で作成された元図形パターンに対して行われるが、2回目以降は、ステップS5で得られた補正図形パターンに対して行われることになり、ステップS6で補正完了と判断されるまで、ステップS2～S5の補正処理が繰り返し実行されることになる。ステップS7のリソグラフィプロセスを構成する露光、現像、エッチングという実工程は、このような繰り返し処理によって最終的に得られた最後の補正図形パターンに基づいて実行されることになる。

[0372] このような補正処理の繰り返しを行うためには、図1に示す図形パターンの形状補正装置100において、特徴量抽出ユニット120に、パターン補正ユニット140によって作成された補正図形パターン15について、当該補正図形パターン15上の評価点の周囲の特徴を示す特徴量を抽出する機能をもたせ、バイアス推定ユニット130に、当該特徴量に基づいて、評価点の補正図形パターン15上の位置と実図形パターン20上の位置とのずれ量を示すプロセスバイアスを推定する機能をもたせ、パターン補正ユニット140に、バイアス推定ユニット130から出力されるプロセスバイアスの推定値に基づいて、補正図形パターン15に対する更なる補正を行うことにより、新たな補正図形パターンを作成する機能をもたせておけばよい。

[0373] そうすれば、補正図形パターン15に対する補正を繰り返し実行することができ、より適切な補正図形パターンを作成することが可能になる。なお、補正図形パターン15に対して、再び補正を実行する際には、既に端点や評価点の定義は完了しているので、評価点設定ユニット110による処理は省略することができる。すなわち、図1に示すブロック図の下端に描かれている補正図形パターン15を、上方へ向かう矢印に従って再びこの形状補正装置100に戻す場合、評価点設定ユニット110ではなく、特徴量抽出ユニット120へ戻せばよい。特徴量抽出ユニット120は、戻された補正図形パターン15に既存の評価点について特徴量の抽出処理を行い、バイアス推定ユニット130は、戻された補正図形パターン15に既存の評価点につい

てプロセスバイアスの推定値 y を求めればよい。

[0374] なお、このような補正処理の繰り返しを行う際には、パターン補正ユニット 140 内の輪郭線分伸縮処理部 143 に、輪郭線分を伸縮する処理を行った後、伸縮処理後の輪郭線分についての評価点の位置を修正する処理を行う機能をもたせておくのが好ましい。

[0375] たとえば、§ 7. 3 で説明した輪郭線分の伸縮処理では、図 58 に示す補正図形パターン 15 が作成されている。この図 58 に示す補正図形パターン 15 を、再び形状補正装置 100 に戻す場合、上述したように、評価点設定ユニット 110 による処理は省略することができる。これは、図 58 に示す補正図形パターン 15 には、既に各端点 A ~ H, F', H' や各評価点 1 ~ 8 が定義されているためである。しかしながら、補正図形パターン 15 上の各評価点 1 ~ 8 の位置は、図示のとおり、必ずしも適切な位置にはなっていない。§ 6. 4 で述べたように、より適切なシミュレーション結果を得る上では、個々の評価点を輪郭線分の中点位置に定義するのが好ましい。ところが、輪郭線分伸縮処理部 143 による伸縮処理を行うと、元の評価点位置は、輪郭線分の中点位置から外れてしまう。

[0376] そこで、補正処理の繰り返しを行う際には、輪郭線分伸縮処理部 143 が、輪郭線分を伸縮する処理を行った後、評価点の位置を修正する処理を行うようにするのが好ましい。具体的には、各評価点の位置を、新たな輪郭線分の中点位置に修正すればよい。図 59 は、図 53 に示す輪郭線分伸縮処理部 143 によって、図 58 に示す伸縮処理後の単位図形の各輪郭線分 O1 ~ O8 について、評価点 1 ~ 8 の位置を修正する処理を行った状態を示す平面図である。この修正により、各評価点 1 ~ 8 の位置は、各輪郭線分 O1 ~ O8 の中点位置となっているので、以後、より適切なシミュレーションを行うことができる。

[0377] <<< § 8. 本発明の特徴とそのメリット >>>

最後に、本発明の特徴とそのメリットを述べておく。

[0378] <8. 1 本発明に係る形状補正方法の特徴>

これまで、本発明を図形パターンの形状補正装置という装置発明として把握した説明を行ってきたが、本発明は図形パターンの形状補正方法という方法発明として把握することもできる。このように方法発明として把握した場合、本発明は、元図形パターンに基づくリソグラフィプロセスによって実基板上に実図形パターンを形成する際に、実図形パターンが元図形パターンに一致するように、元図形パターンの形状を補正して、リソグラフィプロセスで実際に用いる補正図形パターンを作成する図形パターンの形状補正方法ということになる。

[0379] この形状補正方法は、コンピュータが、元図形パターンを構成する個々の単位図形について、内部と外部との境界を示す輪郭線上の所定位置に評価点を設定する評価点設定段階（前述した評価点設定ユニット１１０によって実行される段階）と、コンピュータが、元図形パターンについて、評価点の周囲の特徴を示す特徴量を抽出する特徴量抽出段階（前述した特徴量抽出ユニット１２０によって実行される段階）と、コンピュータが、特徴量に基づいて、評価点の元図形パターン上の位置と実図形パターン上の位置とのずれ量を示すプロセスバイアスを推定するバイアス推定段階（前述したバイアス推定ユニット１３０によって実行される段階）と、コンピュータが、このバイアス推定段階によって得られるプロセスバイアスの推定値に基づいて、元図形パターンに対する補正を行うことにより、補正図形パターンを作成するパターン補正段階（前述したパターン補正ユニット１４０によって実行される段階）と、を有している。

[0380] ここで、評価点設定段階は、多角形からなる個々の単位図形を入力する単位図形入力ステップ（前述した単位図形入力部１１１によって実行されるステップ）と、単位図形を直線状の分割線で分割して複数の多角形からなる分割図形を形成する単位図形分割ステップ（前述した単位図形分割部１１２によって実行されるステップ）と、各分割図形について、当該分割図形を構成する多角形の輪郭線に沿って所定の順方向まわりに一周するように、多角形の個々の辺上にそれぞれ分割図形ベクトルを定義する分割図形ベクトル定義

ステップ（前述した分割図形ベクトル定義部 113 によって実行されるステップ）と、分割図形ベクトルの集合から、互いに隣接する一对の隣接多角形の重複する辺上に配置され、互いに向きが逆方向である重複ベクトル対を探索し、探索された重複ベクトル対のうちの重複区間部分を除去し、残ったベクトルにより、単位図形の輪郭線に沿って配置された複数の単位図形ベクトルを定義する単位図形ベクトル定義ステップ（前述した単位図形ベクトル定義部 114 によって実行されるステップ）と、個々の単位図形ベクトルの始点位置に、それぞれ通常端点を定義する通常端点定義ステップ（前述した通常端点定義部 115 によって実行されるステップ）と、単位図形の輪郭線上において互いに隣接して配置された 2 つの通常端点の間に、それぞれ評価点を定義する評価点定義ステップ（前述した評価点定義部 117 によって実行されるステップ）と、個々の単位図形について、多角形の頂点を構成しない通常端点である中間通常端点を探索し、探索された中間通常端点と同じ位置に補充端点を追加定義する補充端点定義ステップ（前述した補充端点定義部 118 によって実行されるステップ）と、を有している。

[0381] なお、実用上は、評価点設定段階において、上記各ステップに加えて、通常端点定義ステップによって定義された通常端点に対して、数もしくは位置を修正する再構築を行う端点再構築ステップ（前述した端点再構築部 116 によって実行されるステップ）を行うようにし、評価点定義ステップにおいて、上記再構築によって得られた通常端点に基づいて評価点を定義するようにするのが好ましい。

[0382] 一方、パターン補正段階は、個々の評価点について、単位図形の輪郭線に沿って当該評価点の一方の側に隣接する通常端点もしくは補充端点を第 1 の端点とし、他方の側に隣接する通常端点もしくは補充端点を第 2 の端点とする輪郭線分を定義し、個々の評価点についての輪郭線分を、当該評価点のプロセスバイアスの推定値に応じた補正量だけ、輪郭線分に直交する方向に移動させる移動処理を行う輪郭線分移動ステップ（前述した輪郭線分移動処理部 141 によって実行されるステップ）と、移動処理前に互いに同じ位置に

あった中間通常端点および補充端点が、移動処理後に異なる2つの位置に分離した場合、それぞれの移動処理後の位置を連結する新たな輪郭線分を追加する追加処理を行う輪郭線分追加ステップ（前述した輪郭線分追加処理部142によって実行されるステップ）と、移動処理前に多角形の頂点を構成していた通常端点である隅部通常端点が、移動処理後に異なる2つの位置に分離した場合、分離後の2つの隅部通常端点が融合して新たな隅部通常端点となるように、分離後の隅部通常端点を含む各輪郭線分をそれぞれ伸縮する伸縮処理を行う輪郭線分伸縮ステップ（前述した輪郭線分伸縮処理部143によって実行されるステップ）と、を有している。

[0383] そして、本発明に係る図形パターンの形状補正方法では、上記移動処理、上記追加処理、上記伸縮処理を行った後の輪郭線分の集合体を輪郭線とする単位図形によって構成される補正図形パターンが作成される。このような形状補正方法を採用すれば、元図形パターンに対する補正に必要なシミュレーションを行う際に、実用上十分なシミュレーション精度を維持しつつ、できるだけ演算負担を軽減できる効率的な評価点設定を行うことが可能になる。

[0384] <8. 2 分割図形を利用した端点定義のメリット>

ここでは、本発明を採用することにより得られる露光工程におけるメリットを説明する。本発明の重要な特徴は、元図形パターンを構成する単位図形を多角形からなる分割図形に分割した上で、この多角形の頂点位置を基準として端点を定義し、隣接して配置された2つの端点間を結ぶ輪郭線分ごとに評価点を定義し、輪郭線分ごとに位置の補正を行う点にある。このような方法で評価点の定義を行えば、適切な位置に適切な数の評価点を定義することができるので、元図形パターンに対する補正を行う際に、実用上十分なシミュレーション精度を維持しつつ、できるだけ演算負担を軽減できる効率的な評価点設定を行うことが可能になる。

[0385] しかも、分割図形を構成する多角形の頂点位置を基準として端点の定義を行うと、露光工程において、トータルショット数を低減できるというメリットも得られる。すなわち、従来の一般的な手法では、単位図形の個々の辺ご

とに所定間隔で端点を定義する方法が採用されているが、このような従来方法では、トータルショット数が増大するというデメリットが生じる。本発明では、このようなデメリットを解消することができる。以下、この点について具体例を挙げながら説明する。

[0386] 図60は、単位図形の個々の辺ごとに所定間隔で端点を定義するという従来手法の手順を示す平面図である。いま、図60(a)に示すような単位図形10を含む元図形パターンが与えられた場合に、この単位図形10の輪郭線上に複数の端点を定義し、隣接する2つの端点を結ぶ輪郭線分ごとに評価点を定義して補正を行うことを考えてみる。この場合、まず、単位図形10を構成する多角形の頂点が端点として定義され、更に、多角形の各辺上に所定間隔で端点が定義される。ここでは、説明の便宜上、この単位図形10の縦幅の寸法を W_{split} とし、単位図形10の各辺上に、それぞれ W_{split} 以下の所定間隔で端点を定義することにする。

[0387] 図60(b)は、このような方針で、単位図形10上の輪郭線上に端点A～Nを定義した例である。ここで、端点A, E, F, G, H, L, M, Nは、単位図形10を構成する多角形の頂点であるが、残りの端点B, C, D, I, J, Kは、当該多角形の辺上に定義された端点である。上述したように、隣接する端点の間隔が W_{split} 以下となるように端点の定義が行われているため、いずれの端点間隔も W_{split} 以下になっており、隣接する2つの端点到挟まれた輪郭線分の全長は、いずれも W_{split} 以下になっている。

[0388] しかしながら、単位図形10の個々の辺ごとに別個独立して端点を定義しているため、多角形の上辺に定義された端点B, C, Dと、多角形の下辺に定義された端点I, J, Kとの間には、位置に関する整合性が全くない。別言すれば、端点B, C, Dの横方向位置と端点I, J, Kの横方向位置とは揃っていない(図60(b)の破線参照)。

[0389] 単位図形10に対する形状補正は、個々の輪郭線分を移動させることによって行われるので、補正後の単位図形15は、たとえば、図60(c)に示すような形状になる。すなわち、図60(c)に示す補正後の単位図形15は、

図60(b)に示す補正前の単位図形10における輪郭線分BC, CD, DEを若干内側に移動させ、輪郭線分IJを若干外側に移動させ、輪郭線分JKを若干内側に移動させたものになっている。

[0390] リソグラフィプロセスに用いる一般的な描画装置では、1回のショットで台形状の領域を露光させる運用が採られており、そのような描画装置でこの図60(c)に示す補正後の単位図形15に基づく露光工程を行うためには、図に破線で示すような多数の矩形領域に分割して、それぞれ個別のショットにより露光を行う必要がある。すなわち、実際の露光工程では、図60(d)にそれぞれハッチングを施して示した9個の独立した矩形ごとに個別のショットを行う必要があり、トータルショット数は9回になる。このように、従来方法では、輪郭線を構成する多角形の対向辺に定義される端点の位置が揃わないため、当該輪郭線が正図形の場合、トータルショット数が増大するというデメリットが生じる（当該輪郭線が反図形の場合、このようなデメリットはあまり生じない）。

[0391] 一方、図61は、本発明に係る方法により、分割図形を構成する多角形の頂点位置を基準として端点の定義を行う場合の手順を示す平面図である。図61(a)に示す単位図形10は、図60(a)に示す単位図形10と全く同じものであり、ここでは、本発明に係る方法により、単位図形10の輪郭線上に複数の端点を定義し、隣接する2つの端点を結ぶ輪郭線分ごとに評価点を定義して補正を行うことを考えてみる。

[0392] この場合、まず、単位図形10が複数の分割図形に分割される。図61(b)は、単位図形10を5つの分割図形に分割した状態を示している。前述したとおり、本発明における分割処理は、個々の分割図形のサイズが所定の分割間隔 W_{split} 以下となるように行われるので、図61(b)に示す5つの分割図形の縦幅および横幅の寸法は、いずれも W_{split} 以下になっている。そして、これら5つの分割図形の各頂点位置に通常端点が定義される。図61(b)には、このような方針で、単位図形10上の輪郭線上に通常端点A~Nが定義された状態が示されている。ここで、端点A, E, F, G, H, L, M, N

は、単位図形 10 を構成する多角形の頂点となる隅部通常端点であり、残りの端点 B, C, D, I, J, K は、中間通常端点である。

[0393] 図 60 (b) と図 61 (b) とを比較すると、いずれも 14 個の端点 A ~ N が定義されている点は同じであるが、前者では、上辺に定義された端点 B, C, D の位置と下辺に定義された端点 I, J, K の横方向位置が揃っておらず整合性がないのに対して、後者では、上下の端点の横方向位置が揃って整合性が確保されている。

[0394] 図 61 (c) は、図 61 (b) に示す単位図形 10 に対する補正により得られた補正後の単位図形 15 である。この補正後の単位図形 15 は、図 61 (b) に示す補正前の単位図形 10 おける輪郭線分 BC, CD, DE を若干内側に移動させ、輪郭線分 IJ を若干外側に移動させ、輪郭線分 JK を若干内側に移動させたものになっている。図 61 (d) は、この補正後の単位図形 15 に基づく露光工程を行うために分割された個々の矩形領域を示す図である。

[0395] 図 61 (b) に示す分割処理を行う際に、分割間隔 W_{split} を描画装置のビーム最大成型サイズ B_{max} 以下に設定しておけば ($W_{split} \leq B_{max}$ なる設定)、図 61 (d) に示す個々の矩形領域のサイズは、ビーム最大成型サイズ B_{max} 以下になる可能性が高い。露光工程は、これら個々の矩形領域ごとにそれぞれ個別のショットを行うことによりなされ、サイズが B_{max} 以下となる矩形領域は、描画装置による 1 ショットで露光することが可能である。図 61 (d) に示す例の場合、ハッチングを施して示すいずれの矩形領域もサイズが B_{max} 以下となっているので、実際の露光工程では、5 個の独立した矩形領域ごとにそれぞれ個別の 1 ショットを行えば足りる。

[0396] 結局、上述した実施例の場合、従来手法では露光工程時に合計 9 回のショット (図 60 (d) 参照) が必要になっていたのに対し、本発明の手法では合計 5 回のショットで済むことになる。このように、本発明では、分割図形を利用した端点定義が行われるため、露光工程の効率化 (トータルショット数の低減) を図ることができるメリットが得られる。もちろん、図 61 (c) に示す補正後の単位図形 15 を作成する際の各輪郭線分の移動方向および移

動量に応じて、補正後に得られる個々の矩形領域のサイズがビーム最大成型サイズ $B_{\max}$ を超える可能性もあり、その場合には、当該矩形領域を 1 ショットで露光することはできなくなるが、本発明に係る手法を採用すれば、多数の単位図形を含む図形パターン全体についてのトータルショット数は、統計的には必ず低減することになる。

[0397] < 8. 3 補充端点を定義するメリット >

本発明では、上述したとおり、分割図形を利用して通常端点が定義されるとともに、多角形の頂点を構成しない通常端点である中間通常端点については、同じ位置に補充端点が追加定義される。ここでは、中間通常端点について補充端点を定義するメリットを説明する。

[0398] 本発明において補充端点を定義する理由は、§ 6. 5 で説明したとおり、1 つの中間通常端点の両側に位置する 2 組の輪郭線分を別個独立して移動させる処理を行うためである。たとえば、図 5 6 に示す実施例では、中間通常端点 F の位置に補充端点 F' を追加定義したことにより、輪郭線分 O 5 と輪郭線分 O 6 とを別個独立して移動させることができるようになり、中間通常端点 H の位置に補充端点 H' を追加定義したことにより、輪郭線分 O 7 と輪郭線分 O 8 とを別個独立して移動させることができるようになる。

[0399] しかも本発明では、輪郭線分移動処理部 1 4 1 による移動処理は、各評価点を当該評価点が位置する輪郭線に対して直交する方向（法線方向：図 5 6 の太線矢印方向）に移動させることを前提としており、評価点とともに平行移動させる輪郭線分の移動方向も、その法線方向ということになる。このような移動処理を行うと、移動後の中間通常端点およびその補充端点は、いずれも移動前の位置についての法線上に存在することになる。たとえば、図 5 6 に示す例の場合、移動後の端点 F, F' の横方向の位置は移動前と変わらず、移動後の端点 H, H' の縦方向の位置は移動前と変わらない。このような特徴は、露光工程の効率化（トータルショット数の低減）を図る上で貢献する。

[0400] 図 6 2 は、補充端点を定義するメリットを示す平面図である。いま、図 6

2(a) に示すような矩形からなる単位図形 10 が与えられ、この矩形状の単位図形 10 に対して、本発明に係る方法で形状補正を行う場合を考えてみる。ここでは、この単位図形 10 が、縦幅 W_{split} 、横幅 $4 \times W_{split}$ の寸法をもった矩形であるものとする。ここで、 W_{split} は、単位図形分割部 112 によって行われる分割処理の分割間隔であり、説明の便宜上、描画装置のビーム最大成型サイズを B_{max} として、 $W_{split} = B_{max}$ に設定した場合を考える。

[0401] 図 6 2 (b) は、単位図形分割部 112 によって、図 6 2 (a) に示す単位図形 10 を分割した状態を示す平面図である。図示のとおり、単位図形 10 は、それぞれハッチングを施して示す 4 個の正形状の分割図形に分割され、これら各分割図形の頂点位置には、通常端点定義部 115 によって、図示のとおり 10 個の通常端点 A ~ J (×印) が定義される。また、補充端点定義部 118 によって、中間通常端点 B, C, D, G, H, I について同位置に補充端点 B', C', D', G', H', I' (四角印) が追加定義される。

[0402] ここでは、この図 6 2 (b) に示す単位図形 10 に対して、パターン補正ユニット 140 による補正処理が行われた結果、図 6 2 (c) に示すような補正後の単位図形 15 が得られたものとしよう。具体的には、輪郭線分移動処理部 141 によって、輪郭線分 BC', CD', GH', HI' が若干内側に移動されたことになり、端点 B' - B の間、端点 C' - C の間、端点 D' - D の間、端点 G' - G の間、端点 H' - H の間、端点 I' - I の間には、輪郭線分追加処理部 142 によって、それぞれ新たな輪郭線分が追加されたことになる。

[0403] ただ、各輪郭線分はその法線方向に移動されているため、移動処理後の端点 B', B, C', C, D', D, G', G, H', H, I', I の横方向の位置は、移動処理前の位置と変わっていない。したがって、この補正後の単位図形 15 に基づいて、実際の露光工程を行う場合、上述したように $W_{split} = B_{max}$ なる設定をしておけば、図 6 2 (d) に太線の正方形 (一辺が描画装置のビーム最大成型サイズ B_{max} である正方形) で示すように、合計 4 回のシ

ショットを行うことにより露光が完了する。もちろん、図62(c)に示す補正処理において、各輪郭線分を図形の外側に移動させる移動処理が行われると、縦幅が B_{max} を超える部分が生じ、合計4回のショットでは済まないことになるが、多数の単位図形を含む図形パターン全体についてのトータルショット数は、統計的には必ず低減することになる。

[0404] このようなメリットは、§7.4で述べた補正処理の繰り返しを行う際にも有効である。たとえば、図62(a)に示す単位図形10に対する1回目の補正により、図62(c)に示す補正後の単位図形15が得られた後、更に、この補正後の単位図形15に対して2回目の補正を行うことにより、図63(a)に示す補正後の単位図形16が得られた場合を考えてみる。図63(a)に示す単位図形16は、図62(c)に示す単位図形15の輪郭線分 BC' 、 CD' 、 GH' 、 HI' を更に内側に移動したものであるが、移動処理後の端点 B' 、 B 、 C' 、 C 、 D' 、 D 、 G' 、 G 、 H' 、 H 、 I' 、 I の横方向の位置は、やはり移動処理前の位置と変わっていない。

[0405] 図55のデータフォーマット例に示すとおり、通常端点 A 、 B 、 C 、 D 、 E 、 F 、 G 、 H については、それぞれ対応する評価点1～8が記録されるのに対して、補充端点 F' 、 H' については「Φ」という存在しない評価点の空データが記録されている。これは、輪郭線分追加処理部142によって追加された輪郭線分上には、評価点が定義されていないことを意味する。したがって、図63(a)に示す例の場合、輪郭線分追加処理部142によって追加された輪郭線分 $B'B$ 、 $C'C$ 、 $D'D$ 、 $G'G$ 、 $H'H$ 、 $I'I$ の上には、評価点は存在しないことになり、これらの輪郭線分に対しては、評価点位置の移動に基づく移動処理は行われない。別言すれば、これらの輪郭線分は、補正処理を何度繰り返しても、その横方向位置は不変ということになる。

[0406] 本発明において、中間通常端点と同位置に補充端点を定義する意義は、補正処理を行った場合に、中間通常端点と補充端点の位置が、輪郭線の法線方向にのみ変化し、輪郭線に沿った方向には変化しない運用を可能にすること

にあり、そのような運用により、露光工程を効率化（トータルショット数の低減）するメリットが得られるのである。

[0407] 図64は、中間通常端点および補充端点を任意方向に移動させた場合のデメリットを示す平面図である。ここで、図64(a)は、図62(b)に示す単位図形10に対して、中間通常端点および補充端点の位置を任意方向に移動させる補正を行うことによって得られた補正後の単位図形17を示している。図62(c)に示す補正後の単位図形15では、端点B'、B、C'、C、D'、D、G'、G、H'、H、I'、Iの横方向位置が不変であったのに対して、図64(a)に示す補正後の単位図形17では、これらの端点の横方向位置が変わっている。

[0408] そのため、この図64(a)に示す補正後の単位図形17に基づいて、実際の露光工程を行うと、図64(b)に太線の矩形（ビーム最大成型サイズ B_{max} をもつ描画装置の1ショットで露光可能なサイズの矩形）で示すように、合計7回のショットが必要になる。このように、本発明では、補充端点を定義して、中間通常端点と補充端点の移動方向を輪郭線の法線方向に制限する運用を採用しているため、露光工程を効率化するメリットが得られることになる。

[0409] <8.4 端点再構築によるメリット>

上述した本発明の実施形態では、§6.3で述べたように、端点の再構築が行われる。この端点の再構築は、本発明を実施する上で必須要件ではないが、実用上は、端点の再構築を行った方が、端点の数や位置が修正されるため、より効率的な評価点設定を行うことができる。以下、この端点再構築によるメリットを、具体例に基づいて説明する。

[0410] 図54に示すL字型の単位図形10は、図49(b)に示すような端点定義および評価点定義が行われたものであり、図43に示す端点再構築により、三角印で示した端点Dを除去する処理が行われたものである。この図54に示す補正前の単位図形10に対して補正処理を行うと、既に述べたとおり、図58に示す補正後の単位図形15が得られる。この補正後の単位図形15

は、補正前の単位図形 10 をひとまわり大きくした形状を有し、元の L 字型の面影を残した図形になっている。

[0411] 一方、図 65 に示す補正後の単位図形 18 は、図 43 に示す端点再構築を行わなかった場合に得られる図形である。すなわち、図 65 に示す端点 Z（三角印）は、図 43 に示す端点 D に相当するものであり、補正後の単位図形 18 は、この端点 Z が除去されないで残っていたと想定した場合に得られる補正後の図形ということになる。端点 Z は、中間通常端点であるので、端点再構築によって除去されないで残っていると、補充端点を定義する処理によって、補充端点 Z' が追加定義されることになる。図示の例は、中間通常端点 Z が図の上方に移動し、補充端点 Z' が図の下方に移動した例である。この移動処理後に行われる輪郭線分の追加処理により、新たに輪郭線分 Z' Z が追加されている。

[0412] ここで、図 58 に示す補正後の単位図形 15 と、図 65 に示す補正後の単位図形 18 とを比較すると、後者では、頂点 C の近傍の輪郭線が著しく変動していることがわかる。一般に、リソグラフィプロセスを実施した場合のコーナーラウンディング効果（多角形の頂点付近の角ばった部分が丸まってしまう効果）が大きいケースでは、評価点が頂点の近くに存在していると、当該評価点を含む輪郭線分が大きく変動し、図 65 に示す例のように、頂点近傍の輪郭線が著しく変動することになる。このような頂点近傍の輪郭線が著しく変動すると、補正前の図形の基本形状を歪める結果となり、実用上は好ましくない。

[0413] § 6. 3 で述べた端点の再構築を行うと、単位図形の頂点に近い位置の端点を除去することができ、結果的に、評価点を頂点から遠ざけることができる。このため、頂点近傍の輪郭線が著しく変動することを抑制することができ、より効率的な評価点設定を行うことが可能になるというメリットが得られる。

[0414] 最後に、「発明を実施するための形態」の部分についての目次を掲載しておく。各見出しの後の [XXXX] なる数字は段落番号である。

- § 1. 先願発明に係る形状補正装置の基本構成 [0034]
 - 1. 1 図形パターンの形状推定装置 [0035]
 - 1. 2 図形パターンの形状補正装置 [0056]
 - 1. 3 先願発明における特徴量抽出の基本概念 [0071]
- § 2. 特徴量抽出ユニットの詳細 [0086]
 - 2. 1 元画像作成部121による処理手順 [0087]
 - 2. 2 画像ピラミッド作成部122による処理手順 [0103]
 - 2. 3 特徴量算出部123による処理手順 [0128]
 - 2. 4 特徴量抽出処理の変形例 [0144]
- § 3. バイアス推定ユニットの詳細 [0167]
 - 3. 1 ニューラルネットワークによる推定演算 [0168]
 - 3. 2 ニューラルネットワークの学習段階 [0183]
- § 4. 先願発明に係る図形パターンの形状推定方法 [0199]
- § 5. 本発明に係る形状補正装置の基本構成 [0205]
- § 6. 評価点設定ユニットの詳細 [0213]
 - 6. 1 単位図形の分割 [0217]
 - 6. 2 ベクトルおよび通常端点の定義 [0243]
 - 6. 3 端点の再構築 [0268]
 - 6. 4 評価点の定義 [0316]
 - 6. 5 補充端点の定義 [0321]
- § 7. パターン補正ユニットの詳細 [0336]
 - 7. 1 輪郭線分移動処理 [0339]
 - 7. 2 輪郭線分追加処理 [0353]
 - 7. 3 輪郭線分伸縮処理 [0360]
 - 7. 4 補正処理の繰り返し [0367]
- § 8. 本発明の特徴とそのメリット [0377]
 - 8. 1 本発明に係る形状補正方法の特徴 [0378]
 - 8. 2 分割図形を利用した端点定義のメリット [0384]

8. 3 補充端点を定義するメリット [0397]

8. 4 端点再構築によるメリット [0409]

産業上の利用可能性

[0415] 本発明に係る図形パターンの形状補正装置および形状補正方法は、半導体デバイスの製造プロセスなど、特定の材料層に対して微細なパターンニング加工を施す必要がある分野において、元図形パターンに基づくリソグラフィプロセスによって実基板上に実図形パターンを形成する際に、正確な寸法をもった実図形パターンを形成する技術として広く利用することができる。

符号の説明

[0416] 1～18：評価点

10：元図形パターン（補正前の単位図形）

10a～10l：分割図形

10X：図形本体部

10Y：穴開き部

15：補正図形パターン（補正後の単位図形）

15a～10d：補正後の分割図形

16, 17, 18：補正後の単位図形

20：実図形パターン

100：図形パターンの形状補正装置

100'：図形パターンの形状推定装置

110：評価点設定ユニット

111：単位図形入力部

112：単位図形分割部

113：分割図形ベクトル定義部

114：単位図形ベクトル定義部

115：通常端点定義部

116：端点再構築部

117：評価点定義部

1 1 8 : 補充端点定義部

1 2 0 : 特徴量抽出ユニット

1 2 1 : 元画像作成部

1 2 2 : 画像ピラミッド作成部

1 2 3 : 特徴量算出部

1 3 0 : バイアス推定ユニット

1 3 1 : 特徴量入力部

1 3 2 : 推定演算部

1 4 0 : パターン補正ユニット

1 4 1 : 輪郭線分移動処理部

1 4 2 : 輪郭線分追加処理部

1 4 3 : 輪郭線分伸縮処理部

A ~ D : 個々の画素 / 個々の画素の画素値

A ~ Z : 端点

a ~ d : 評価点 E と各画素の中心点との横方向距離もしくは縦方向距離

A 1 ~ A 4 : 順方向矢印

A 0 : 移動処理前の隅部通常端点

A 1 , A 2 : 移動処理後の隅部通常端点

B_{max} : ビーム最大成型サイズ

b , b (1 , 1) ~ b (i + 1 , M (i + 1)) , b (N + 1) : ニューラルネットワークのパラメータ

C 1 , C 2 : 参考円

D 1 ~ D_n : 差分画像

E , E 1 1 ~ E 2 3 : 評価点

F 1 ~ F 5 : 元図形パターン 1 0 を構成する図形 (長方形)

f (ξ) : ニューラルネットワークの演算に用いる関数

G : 按分值

G F 3 3 , G F 5 5 : ガウシアンフィルタ

H : 按分值

$h(1, 1) \sim h(N, M(N))$: ニューラルネットワークの隠れ層のニューロン／その演算値

i : ニューラルネットワークの隠れ層の段数を示すパラメータ

J : 二直線の交点

k : 画像番号を示すパラメータ

L : 学習情報

L F 3 3, L F 5 5 : ラプラシアンフィルタ

L 1, L 2, L 3 : 端点間の距離

L cd, L de, L ij : 端点間の距離

L even : 隣接端点の間隔 (等間隔)

L odd : 隣接端点の間隔 (端数間隔)

L reg : 隣接端点の間隔 (基準間隔)

M 1 : 面積密度マップ

M 2 : エッジ長密度マップ

M 3 : ドーズ密度マップ

M (1) ~ M (N) : ニューラルネットワークの各隠れ層の次元

N : ニューラルネットワークの隠れ層の段数／端点の総数

n : 画像ピラミッドの階層数／端点の番号

O, O 1 ~ O 8, O 5', O 7' : 輪郭線分

O a, O b : 輪郭線分

O in : 内側輪郭線 (反図形)

O out : 外側輪郭線 (正図形)

P 1 ~ P n : 階層画像

P k : 第 k 番目の階層画像

P C : 補正画像

P D : 画像ピラミッド (副画像ピラミッド)

P P : 画像ピラミッド (主画像ピラミッド)

Q 1 : 第 1 番目の準備画像 (元画像)

Q k : 第 k 番目の準備画像

S : 実基板

S 1 ~ S 8 4 8 : 流れ図の各ステップ

S S 1 ~ S S 4 4 : 流れ図の各ステップ

T 1 , T 2 , T (n - 1) , T (n) , T (n + 1) , T (n + 2) : 通常
端点

T ' , T (n) ' , B ' ~ Z ' , : 補充端点

U : 画素

u : 画素寸法

V 1 ~ V 2 0 : 単位図形ベクトル

V a 1 ~ V h 4 : 分割図形ベクトル

W , W (1 , 1 1) ~ W (N + 1 , 1 M (N)) : ニューラルネットワーク
のパラメータ

Wmerge : 融合基準値

Wrest : 残余部分の幅

Wsplit : 分割間隔

X : 二次元 X Y 平面上の座標軸

x , x 1 ~ x n : 特徴量

Y : 二次元 X Y 平面上の座標軸

y , y 1 1 ~ y 1 3 : プロセスバイアス

Z : 削除された端点

ξ : 関数 f の引数

$\xi 1 \sim \xi 4$: 分割図形の頂点

Φ : 存在しない評価点の空データ

請求の範囲

[請求項1]

元図形パターン（１０）に基づくリソグラフィプロセスによって実基板（Ｓ）上に実図形パターン（２０）を形成する際に、前記実図形パターン（２０）が前記元図形パターン（１０）に一致するように、前記元図形パターン（１０）の形状を補正して、前記リソグラフィプロセスで実際に用いる補正図形パターン（１５）を作成する図形パターンの形状補正装置（１００）であって、

前記元図形パターン（１０）を構成する個々の単位図形（１０）について、内部と外部との境界を示す輪郭線上の所定位置に評価点（Ｅ）を設定する評価点設定ユニット（１１０）と、

前記元図形パターン（１０）について、前記評価点（Ｅ）の周囲の特徴を示す特徴量（ $x_1 \sim x_n$ ）を抽出する特徴量抽出ユニット（１２０）と、

前記特徴量（ $x_1 \sim x_n$ ）に基づいて、前記評価点（Ｅ）の前記元図形パターン（１０）上の位置と前記実図形パターン（２０）上の位置とのずれ量を示すプロセスバイアス（ y ）を推定するバイアス推定ユニット（１３０）と、

前記バイアス推定ユニット（１３０）から出力されるプロセスバイアスの推定値（ y ）に基づいて、前記元図形パターン（１０）に対する補正を行うことにより、補正図形パターン（１５）を作成するパターン補正ユニット（１４０）と、

を備え、

前記評価点設定ユニット（１１０）が、

多角形からなる個々の単位図形（１０）を入力する単位図形入力部（１１１）と、

前記単位図形（１０）を直線状の分割線で分割して複数の多角形からなる分割図形（１０_a～１０_l）を形成する単位図形分割部（１１２）と、

前記各分割図形（10a～10l）について、当該分割図形を構成する多角形の輪郭線に沿って所定の順方向まわりに一周するように、多角形の個々の辺上にそれぞれ分割図形ベクトル（Va1～Vh4）を定義する分割図形ベクトル定義部（113）と、

前記分割図形ベクトル（Va1～Vh4）の集合から、互いに隣接する一对の隣接多角形の重複する辺上に配置され、互いに向きが逆方向である重複ベクトル対を探索し、探索された重複ベクトル対のうちの重複区間部分を除去し、残ったベクトルにより、前記単位図形（10）の輪郭線に沿って配置された複数の単位図形ベクトル（V1～V20）を定義する単位図形ベクトル定義部（114）と、

個々の単位図形ベクトル（V1～V20）の始点位置に、それぞれ通常端点（T）を定義する通常端点定義部（115）と、

個々の単位図形（10）の輪郭線上において互いに隣接して配置された2つの通常端点（T）の間に、それぞれ評価点（E）を定義する評価点定義部（117）と、

個々の単位図形（10）について、多角形の頂点を構成しない通常端点（T）である中間通常端点を探索し、探索された中間通常端点と同じ位置に補充端点（T'）を追加定義する補充端点定義部（118）と、

を有し、

前記パターン補正ユニット（140）が、

個々の評価点（E）について、単位図形（10）の輪郭線に沿って当該評価点（E）の一方の側に隣接する通常端点（T）もしくは補充端点（T'）を第1の端点とし、他方の側に隣接する通常端点（T）もしくは補充端点（T'）を第2の端点とする輪郭線分を定義し、個々の評価点（E）についての輪郭線分を、当該評価点（E）のプロセスバイアスの推定値（y）に応じた補正量だけ、輪郭線分に直交する方向に移動させる移動処理を行う輪郭線分移動処理部（141）と、

移動処理前に互いに同じ位置にあった中間通常端点および補充端点（ T' ）が、移動処理後に異なる2つの位置に分離した場合、それぞれの移動処理後の位置を連結する新たな輪郭線分を追加する追加処理を行う輪郭線分追加処理部（142）と、

移動処理前に多角形の頂点を構成していた通常端点（ T ）である隅部通常端点が、移動処理後に異なる2つの位置に分離した場合、分離後の2つの隅部通常端点が融合して新たな隅部通常端点となるように、分離後の隅部通常端点を含む各輪郭線分をそれぞれ伸縮する伸縮処理を行う輪郭線分伸縮処理部（143）と、

を有し、前記移動処理、前記追加処理、前記伸縮処理を行った後の輪郭線分の集合体を輪郭線とする単位図形（10）によって構成される補正図形パターン（15）を作成することを特徴とする図形パターンの形状補正装置（100）。

[請求項2] 請求項1に記載の図形パターンの形状補正装置（100）において、

単位図形分割部（112）が、単位図形（10）を分割して、台形からなる分割図形を形成することを特徴とする図形パターンの形状補正装置（100）。

[請求項3] 請求項1または2に記載の図形パターンの形状補正装置（100）において、

単位図形入力部（111）が、所定の長手方向軸に沿って伸び、互いに平行な直線状の上辺および下辺を有する細長い帯状単位図形（10）を入力したときに、

単位図形分割部（112）が、前記帯状単位図形（10）を、前記長手方向軸に沿って所定間隔で配置された複数の分割線によって分割することにより、前記長手方向軸に沿って一次元的に並んだ複数の分割図形を形成することを特徴とする図形パターンの形状補正装置（100）。

- [請求項4] 請求項1または2に記載の図形パターンの形状補正装置（100）において、
- 単位図形入力部（111）が、XY平面上に定義された単位図形（10）を入力したときに、
- 単位図形分割部（112）が、X軸に平行な水平分割線をY軸方向に所定間隔をあけて複数配置し、これら複数の水平分割線によって前記単位図形（10）を分割することにより、X軸方向に伸びる細長い複数の帯状図形を形成し、更に、前記複数の帯状図形のそれぞれを、X軸に沿って所定間隔で配置されY軸に平行な複数の垂直分割線によって分割することにより、二次元的に並んだ複数の分割図形を形成することを特徴とする図形パターンの形状補正装置（100）。
- [請求項5] 請求項1～4のいずれかに記載の図形パターンの形状補正装置（100）において、
- 単位図形分割部（112）が、リソグラフィプロセスで用いる描画装置のビーム最大成型サイズを設定する機能を有し、単位図形（10）の分割に用いる分割線の配置間隔を前記ビーム最大成型サイズ以下に設定することを特徴とする図形パターンの形状補正装置（100）。
- [請求項6] 請求項1～5のいずれかに記載の図形パターンの形状補正装置（100）において、
- 分割図形ベクトル定義部（113）が、K角形からなる分割図形の各頂点に、輪郭線に沿って時計まわりもしくは反時計まわりの順に第1の頂点～第Kの頂点と番号を付し、第kの頂点を始点とし、第（k+1）の頂点（但し、k=Kの場合は第1の頂点）を終点とする第kの分割図形ベクトル（ $V_{a1} \sim V_{h4}$ ）を定義する処理を、k=1～Kまで繰り返し行うことにより、当該分割図形についての分割図形ベクトルを定義することを特徴とする図形パターンの形状補正装置（100）。

[請求項7] 請求項1～6のいずれかに記載の図形パターンの形状補正装置（100）において、

評価点定義部（117）が、単位図形（10）の輪郭線上において互いに隣接して配置された2つの通常端点の中点位置に評価点を定義することを特徴とする図形パターンの形状補正装置（100）。

[請求項8] 請求項1～7のいずれかに記載の図形パターンの形状補正装置（100）において、

補充端点定義部（118）が、1つの輪郭線について定義されている合計N個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1)$ ～第Nの通常端点 $T(N)$ と呼んだときに、第 n の通常端点 $T(n)$ に着目して、第 $(n-1)$ の通常端点 $T(n-1)$ （但し、 $n=1$ の場合は第Nの通常端点 $T(N)$ ）、第 n の通常端点 $T(n)$ 、第 $(n+1)$ の通常端点 $T(n+1)$ （但し、 $n=N$ の場合は第1の通常端点 $T(1)$ ）の3点が一直線上にある場合に、前記第 n の通常端点 $T(n)$ を中間通常端点と判定し、この中間通常端点と同じ位置に補充端点を追加定義する処理を、 $n=1\sim N$ まで繰り返し実行することを特徴とする図形パターンの形状補正装置（100）。

[請求項9] 請求項1～8のいずれかに記載の図形パターンの形状補正装置（100）において、

輪郭線分伸縮処理部（143）が、ある1つの隅部通常端点が移動処理により第1の移動端点と第2の移動端点とに分離した場合に、前記第1の移動端点を一方の端点とする第1の移動後輪郭線分を含む直線と前記第2の移動端点を一方の端点とする第2の移動後輪郭線分を含む直線との交点を求め、当該交点を新たな隅部通常端点とし、前記第1の移動後輪郭線分および前記第2の移動後輪郭線分に対して一方の端点が前記新たな隅部通常端点となるような伸縮処理を行うことを特徴とする図形パターンの形状補正装置（100）。

[請求項10] 請求項1～9のいずれかに記載の図形パターンの形状補正装置（100）において、

評価点設定ユニット（110）が、通常端点定義部（115）によって単位図形（10）の輪郭線上に定義された通常端点に対して、数もしくは位置を修正する再構築を行う端点再構築部（116）を更に有し、

評価点定義部（117）が、前記再構築によって得られた通常端点に基づいて評価点を定義することを特徴とする図形パターンの形状補正装置（100）。

[請求項11] 請求項10に記載の図形パターンの形状補正装置（100）において、

端点再構築部（116）が、一部もしくは全部の単位図形（10）について、中間通常端点の一部もしくは全部を除去する端点除去処理により、通常端点（T）の再構築を行うことを特徴とする図形パターンの形状補正装置（100）。

[請求項12] 請求項11に記載の図形パターンの形状補正装置（100）において、

端点再構築部（116）が、1つの単位図形（10）の1本の輪郭線について定義されている合計N個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1)$ ～第Nの通常端点 $T(N)$ と呼び、第nの通常端点 $T(n)$ に着目して、第nの通常端点 $T(n)$ と、その直前に位置する除去対象となっていない通常端点 $T(n-m)$ （但し、mは、自然数であり、 $n-m < 1$ の場合は、 $T(n-m+N)$ ）との距離を $L1$ とし、第nの通常端点 $T(n)$ と第 $(n+1)$ の通常端点 $T(n+1)$ （但し、 $n=N$ の場合は第1の通常端点 $T(1)$ ：以下同様）との間の距離を $L2$ としたときに、

$L1$ および $L2$ の少なくとも一方が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ （但し、 $n=1$ の場合は第Nの通常

端点 $T(N)$), $T(n)$, $T(n+1)$ の3点が一直線上にある場合に、前記第 n の通常端点 $T(n)$ を除去対象とする判定処理を、 $n=1\sim N$ まで繰り返し実行し、除去対象となった通常端点を除去する端点除去処理を行うことを特徴とする図形パターンの形状補正装置(100)。

[請求項13] 請求項11に記載の図形パターンの形状補正装置(100)において、

端点再構築部(116)が、1つの単位図形(10)の1本の輪郭線について定義されている合計 N 個の通常端点を、当該輪郭線に沿った順に、それぞれ第1の通常端点 $T(1)\sim$ 第 N の通常端点 $T(N)$ と呼び、第 n の通常端点 $T(n)$ と、その直前に位置する除去対象となっていない通常端点 $T(n-m)$ (但し、 m は、自然数であり、 $n-m<1$ の場合は、 $T(n-m+N)$)との距離を L_1 とし、第 n の通常端点 $T(n)$ と第 $(n+1)$ の通常端点 $T(n+1)$ (但し、 $n=N$ の場合は第1の通常端点 $T(1)$:以下同様)との間の距離を L_2 とし、第 $(n+1)$ の通常端点 $T(n+1)$ と第 $(n+2)$ の通常端点 $T(n+2)$ (但し、 $n=N-1$ の場合は第1の通常端点 $T(1)$)、 $n=N$ の場合は第2の通常端点 $T(2)$:以下同様)との間の距離を L_3 としたときに、

L_2 が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ (但し、 $n=1$ の場合は第 N の通常端点 $T(N)$:以下同様)、 $T(n)$ 、 $T(n+1)$ 、 $T(n+2)$ の4点が一直線上にある場合には、 $L_1<L_3$ であれば通常端点 $T(n)$ を除去対象とし、 $L_1>L_3$ であれば通常端点 $T(n+1)$ を除去対象とし、 $L_1=L_3$ であれば通常端点 $T(n)$ もしくは通常端点 $T(n+1)$ のいずれかを除去対象とし、

L_2 が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n-1)$ 、 $T(n)$ 、 $T(n+1)$ の3点は一直線上にあるが、通常端点

$T(n+2)$ は当該一直線上にない場合には、通常端点 $T(n)$ を除去対象とし、

L_2 が所定の基準値 W_{merge} 以下であり、かつ、通常端点 $T(n)$ 、 $T(n+1)$ 、 $T(n+2)$ の3点は一直線上にあるが、通常端点 $T(n-1)$ は当該一直線上にない場合には、通常端点 $T(n+1)$ を除去対象とする、

という処理を、 $n \geq N$ が満たされるまで、 n を1ずつ増加させながら繰り返し実行し、除去対象となった通常端点を除去する端点除去処理を行うことを特徴とする図形パターンの形状補正装置（100）。

[請求項14] 請求項10に記載の図形パターンの形状補正装置（100）において、

端点再構築部（116）が、一部の輪郭線について、中間通常端点を一旦除去し、当該一部の輪郭線の各辺に新たな中間通常端点を配置する端点再配置処理により、通常端点の再構築を行うことを特徴とする図形パターンの形状補正装置（100）。

[請求項15] 請求項14に記載の図形パターンの形状補正装置（100）において、

端点再構築部（116）が、特定の輪郭線についての中間通常端点を除去した後、前記特定の輪郭線の各辺上に所定間隔で中間通常端点を再配置することにより、端点再配置処理を行うことを特徴とする図形パターンの形状補正装置（100）。

[請求項16] 請求項10に記載の図形パターンの形状補正装置（100）において、

端点再構築部（116）が、請求項11～13のいずれかに記載の端点除去処理と、請求項14または15に記載の端点再配置処理と、を単位図形（10）の輪郭線ごとに選択的に実行する機能を有し、

順方向まわりの単位図形ベクトル（ $V_1 \sim V_{20}$ ）によって囲まれた正図形上の通常端点に対しては、前記端点除去処理により端点再構

築を行い、前記順方向とは逆の逆方向まわりの単位図形ベクトル（ $V_1 \sim V_{20}$ ）によって囲まれた反図形上の通常端点に対しては、前記端点再配置処理により端点再構築を行うことを特徴とする図形パターンの形状補正装置（100）。

[請求項17] 請求項1～16のいずれかに記載の図形パターンの形状補正装置（100）において、

特徴量抽出ユニット（120）が、パターン補正ユニット（140）によって作成された補正図形パターン（15）について、当該補正図形パターン（15）上の評価点（E）の周囲の特徴を示す特徴量（ $x_1 \sim x_n$ ）を抽出する機能を有し、

バイアス推定ユニット（130）が、前記特徴量（ $x_1 \sim x_n$ ）に基づいて、前記評価点（E）の前記補正図形パターン（15）上の位置と実図形パターン（20）上の位置とのずれ量を示すプロセスバイアス（ y ）を推定する機能を有し、

パターン補正ユニット（140）が、前記バイアス推定ユニット（130）から出力されるプロセスバイアス（ y ）の推定値に基づいて、前記補正図形パターン（15）に対する更なる補正を行うことにより、新たな補正図形パターン（15）を作成する機能を有し、

補正図形パターン（15）に対する補正が繰り返し実行されることを特徴とする図形パターンの形状補正装置（100）。

[請求項18] 請求項17に記載の図形パターンの形状補正装置（100）において、

輪郭線分伸縮処理部（143）が、輪郭線分を伸縮する処理を行った後、伸縮処理後の輪郭線分についての評価点（E）の位置を修正する処理を行うことを特徴とする図形パターンの形状補正装置（100）。

[請求項19] 請求項1～18のいずれかに記載の図形パターンの形状補正装置（100）または当該図形パターンの形状補正装置（100）に含まれ

る評価点設定ユニット（１１０）もしくはパターン補正ユニット（１４０）としてコンピュータを機能させるプログラム。

[請求項20]

元図形パターン（１０）に基づくリソグラフィプロセスによって実基板（Ｓ）上に実図形パターン（２０）を形成する際に、前記実図形パターン（２０）が前記元図形パターン（１０）に一致するように、前記元図形パターン（１０）の形状を補正して、前記リソグラフィプロセスで実際に用いる補正図形パターン（１５）を作成する図形パターンの形状補正方法であって、

コンピュータが、前記元図形パターン（１０）を構成する個々の単位図形（１０）について、内部と外部との境界を示す輪郭線上の所定位置に評価点（Ｅ）を設定する評価点設定段階（Ｓ２）と、

コンピュータが、前記元図形パターン（１０）について、前記評価点（Ｅ）の周囲の特徴を示す特徴量（ $x_1 \sim x_n$ ）を抽出する特徴量抽出段階（Ｓ３）と、

コンピュータが、前記特徴量（ $x_1 \sim x_n$ ）に基づいて、前記評価点（Ｅ）の前記元図形パターン（１０）上の位置と前記実図形パターン（２０）上の位置とのずれ量を示すプロセスバイアス（ y ）を推定するバイアス推定段階（Ｓ４）と、

コンピュータが、前記バイアス推定段階（Ｓ４）によって得られるプロセスバイアスの推定値（ y ）に基づいて、前記元図形パターン（１０）に対する補正を行うことにより、補正図形パターン（１５）を作成するパターン補正段階（Ｓ５）と、

を有し、

前記評価点設定段階（Ｓ２）が、

多角形からなる個々の単位図形（１０）を入力する単位図形入力ステップ（１１１）と、

前記単位図形（１０）を直線状の分割線で分割して複数の多角形からなる分割図形を形成する単位図形分割ステップ（１１２）と、

前記各分割図形について、当該分割図形を構成する多角形の輪郭線に沿って所定の順方向まわりに一周するように、多角形の個々の辺上にそれぞれ分割図形ベクトル ($V_{a1} \sim V_{h4}$) を定義する分割図形ベクトル定義ステップ (113) と、

前記分割図形ベクトル ($V_{a1} \sim V_{h4}$) の集合から、互いに隣接する一対の隣接多角形の重複する辺上に配置され、互いに向きが逆方向である重複ベクトル対を探索し、探索された重複ベクトル対のうちの重複区間部分を除去し、残ったベクトルにより、前記単位図形 (10) の輪郭線に沿って配置された複数の単位図形ベクトル ($V_1 \sim V_{20}$) を定義する単位図形ベクトル定義ステップ (114) と、

個々の単位図形ベクトル ($V_1 \sim V_{20}$) の始点位置に、それぞれ通常端点 (T) を定義する通常端点定義ステップ (115) と、

前記単位図形 (10) の輪郭線上において互いに隣接して配置された2つの通常端点 (T) の間に、それぞれ評価点 (E) を定義する評価点定義ステップ (117) と、

個々の単位図形 (10) について、多角形の頂点を構成しない通常端点である中間通常端点を探索し、探索された中間通常端点と同じ位置に補充端点を追加定義する補充端点定義ステップ (118) と、
を有し、

前記パターン補正段階 (S5) が、

個々の評価点 (E) について、単位図形 (10) の輪郭線に沿って当該評価点 (E) の一方の側に隣接する通常端点 (T) もしくは補充端点 (T') を第1の端点とし、他方の側に隣接する通常端点 (T) もしくは補充端点 (T') を第2の端点とする輪郭線分を定義し、個々の評価点 (E) についての輪郭線分を、当該評価点 (E) のプロセスバイアスの推定値 (y) に応じた補正量だけ、輪郭線分に直交する方向に移動させる移動処理を行う輪郭線分移動ステップ (141) と、

移動処理前に互いに同じ位置にあった中間通常端点および補充端点（ T' ）が、移動処理後に異なる2つの位置に分離した場合、それぞれの移動処理後の位置を連結する新たな輪郭線分を追加する追加処理を行う輪郭線分追加ステップ（142）と、

移動処理前に多角形の頂点を構成していた通常端点である隅部通常端点が、移動処理後に異なる2つの位置に分離した場合、分離後の2つの隅部通常端点が融合して新たな隅部通常端点となるように、分離後の隅部通常端点を含む各輪郭線分をそれぞれ伸縮する伸縮処理を行う輪郭線分伸縮ステップ（143）と、

を有し、前記移動処理、前記追加処理、前記伸縮処理を行った後の輪郭線分の集合体を輪郭線とする単位図形（10）によって構成される補正図形パターン（15）を作成することを特徴とする図形パターンの形状補正方法。

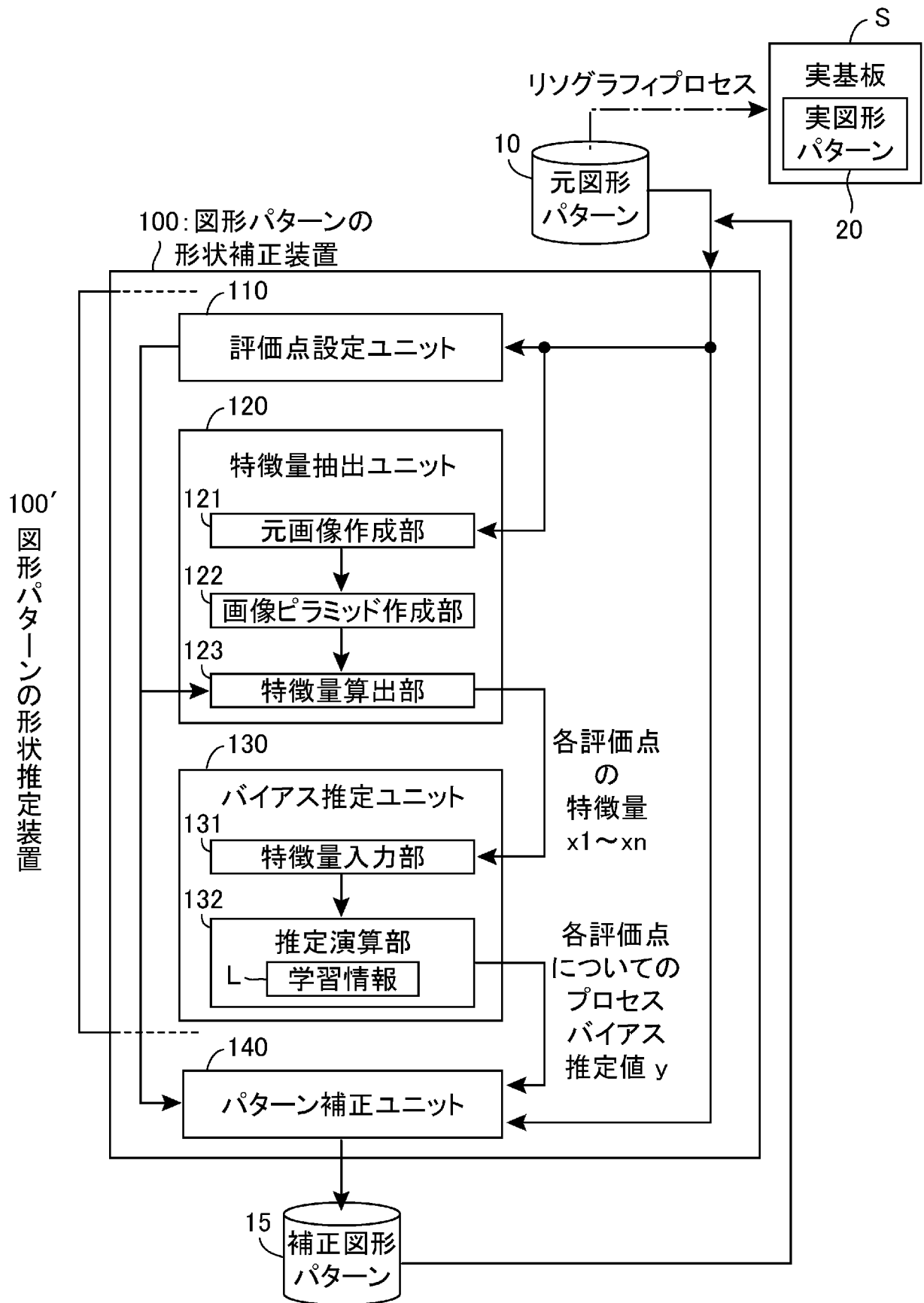
[請求項21]

請求項20に記載の図形パターンの形状補正方法において、

評価点設定段階（S2）が、通常端点定義ステップ（115）によって定義された通常端点（ T ）に対して、数もしくは位置を修正する再構築を行う端点再構築ステップ（116）を更に有し、

評価点定義ステップ（117）において、前記再構築によって得られた通常端点（ T ）に基づいて評価点（ E ）を定義することを特徴とする図形パターンの形状補正方法。

[図1]



元図形パターン

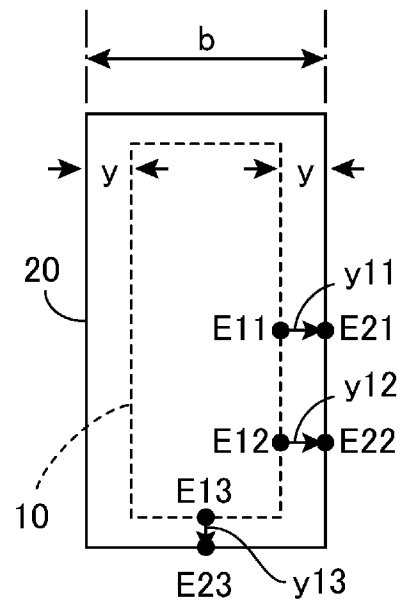


実図形パターン



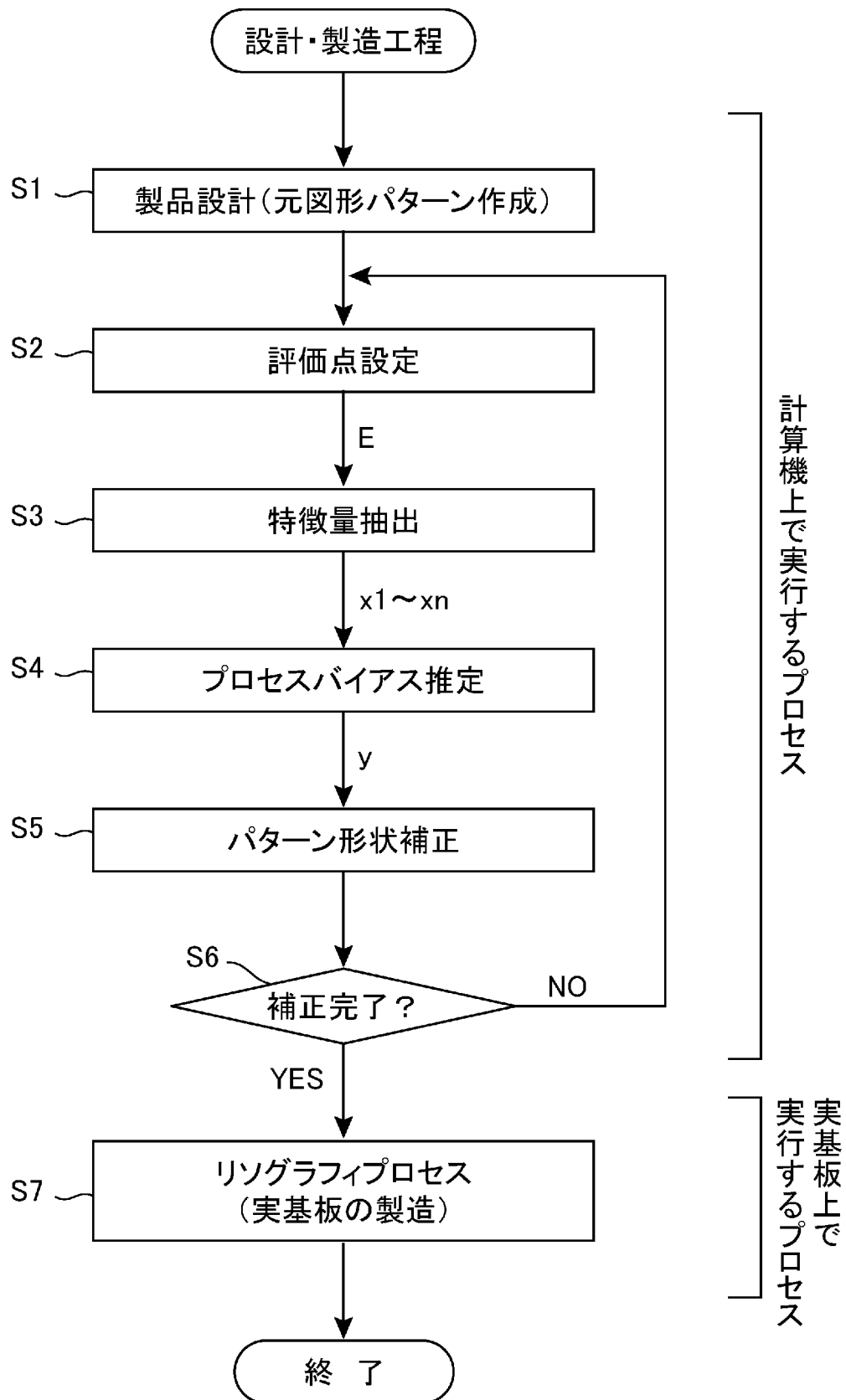
A diagram of a rectangular specimen. The width is labeled a and the height is labeled 10 . Three electrodes are indicated by black dots: E11 is on the right edge, E12 is on the right edge below E11, and E13 is on the bottom edge.

(a)

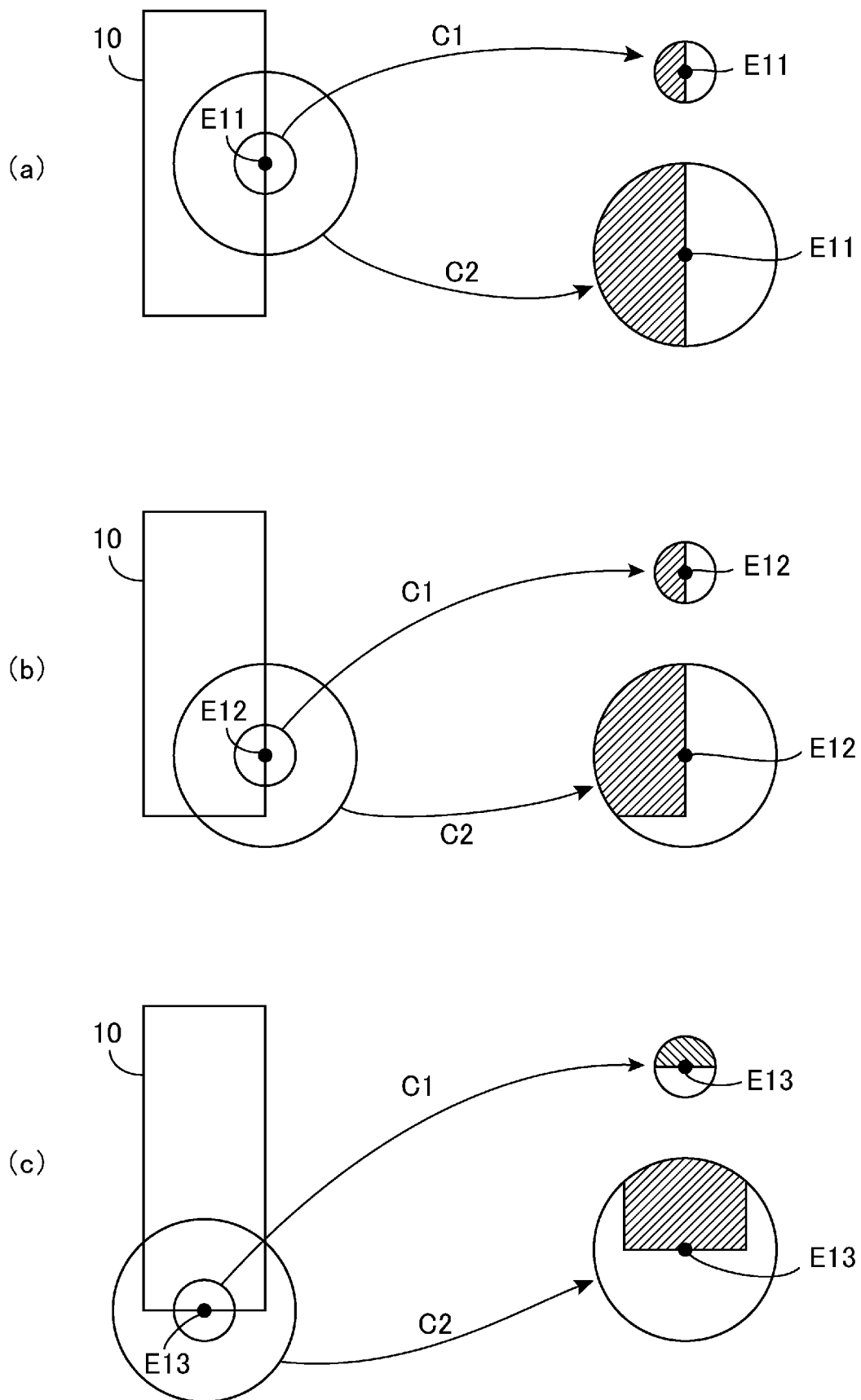


(b)

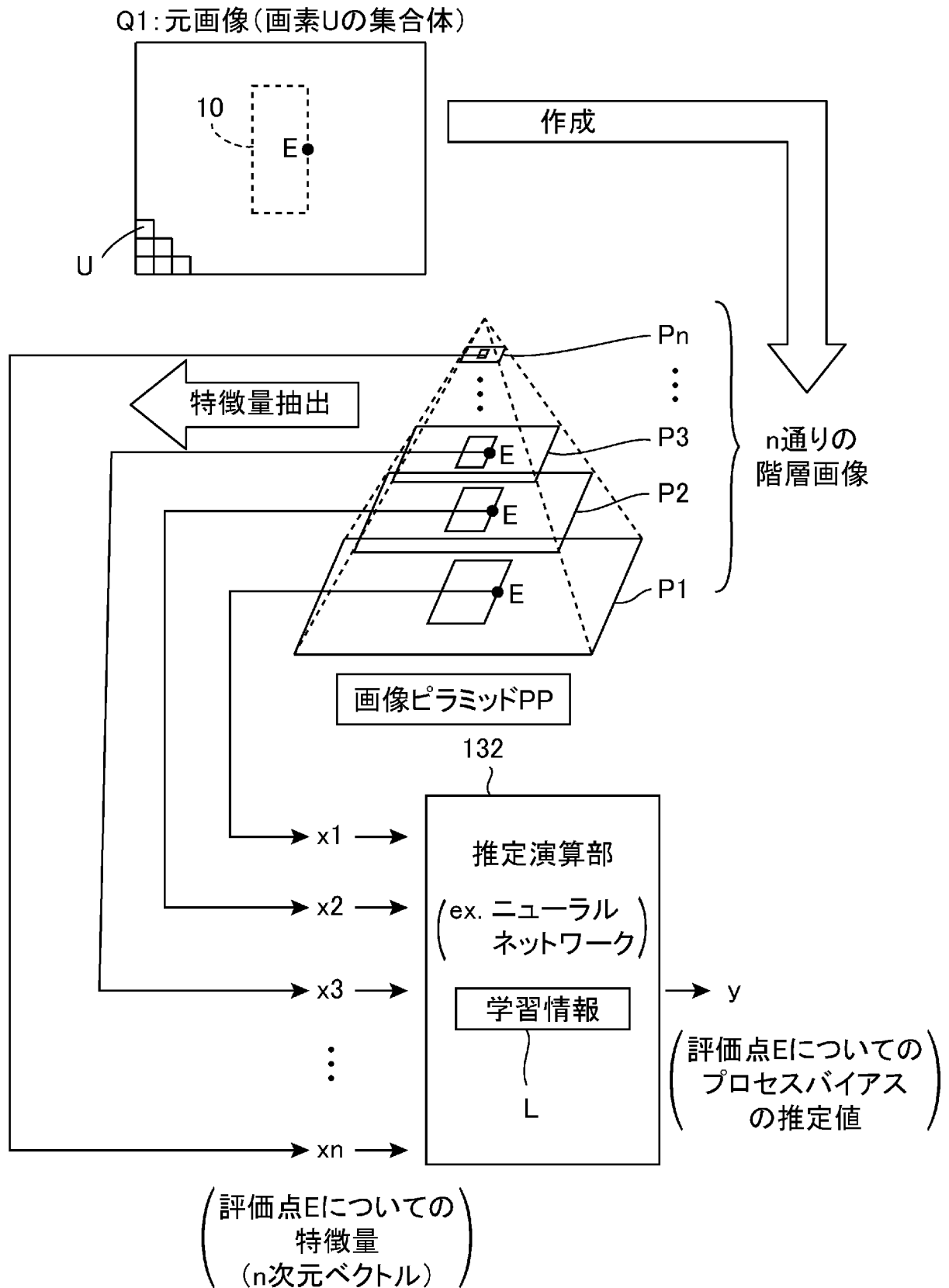
[図4]



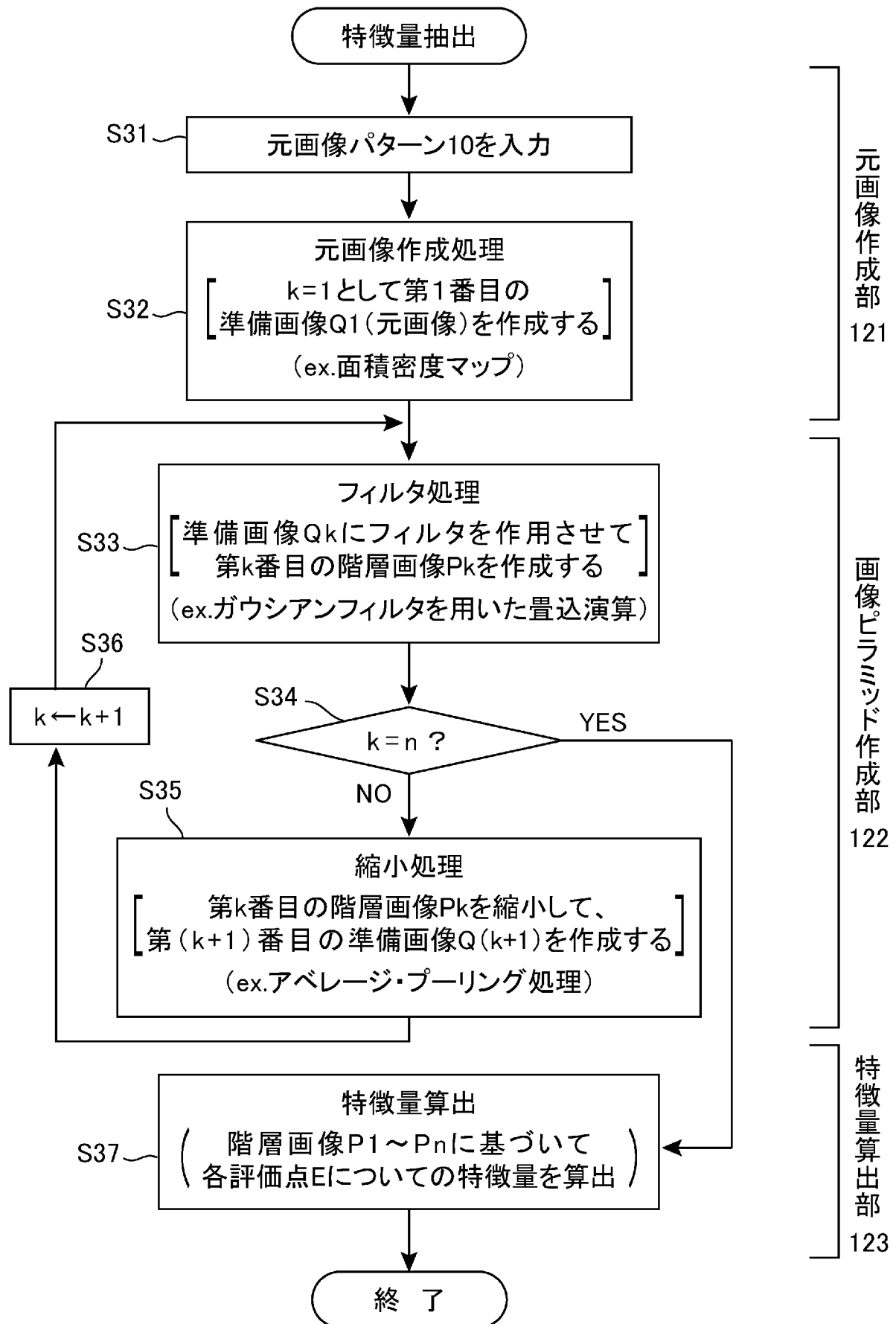
[図5]



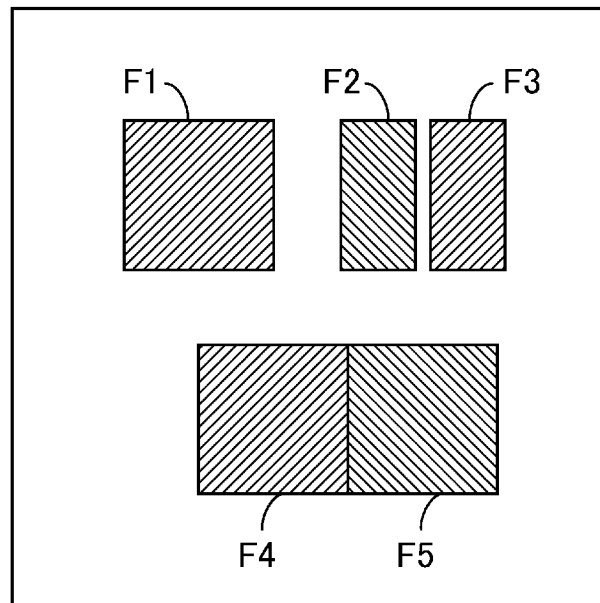
[図6]



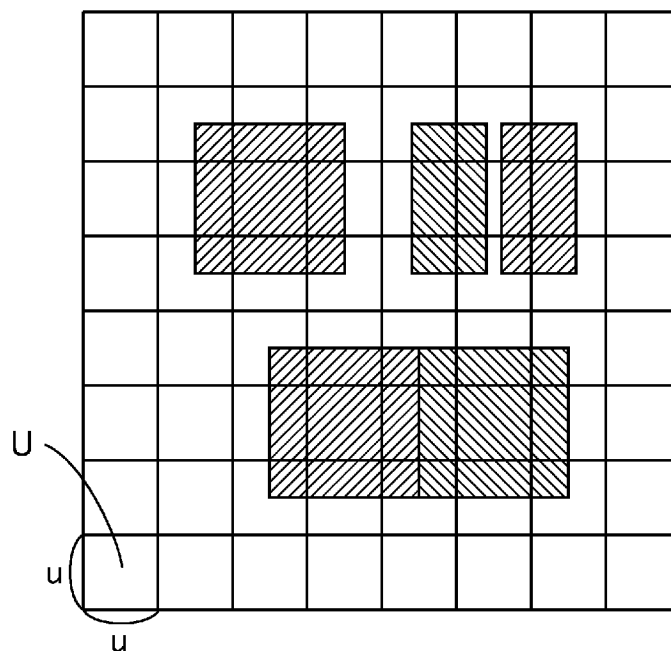
[図7]



[図8]

10: 元図形パターン

[図9]

元図形パターン10+画素の二次元配列

[図10]

M1:面積密度マップ

	0.25	0.5	0.25	0.3	0.4	0.3	
	0.5	1.0	0.5	0.6	0.8	0.6	
	0.25	0.5	0.25	0.3	0.4	0.3	
		0.25	0.5	0.5	0.5	0.25	
		0.5	1.0	1.0	1.0	0.5	
		0.25	0.5	0.5	0.5	0.25	

(空白のセルは値0)

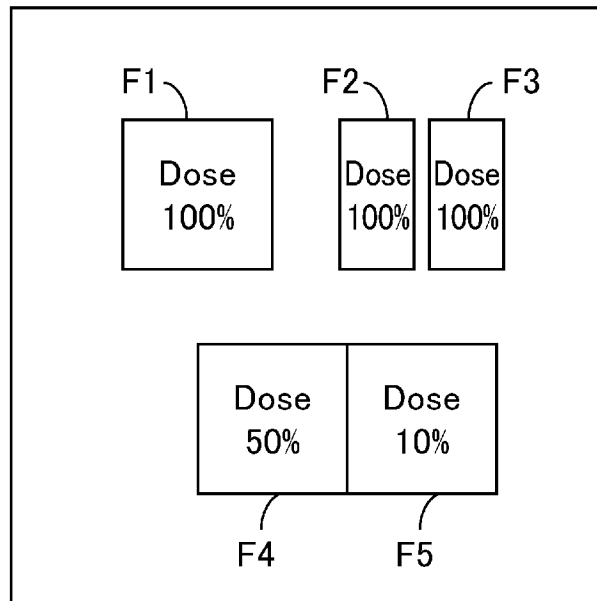
[図11]

M2:エッジ長密度マップ

	1.0	1.0	1.0	1.1	1.8	1.1	
	1.0		1.0	1.0	2.0	1.0	
	1.0	1.0	1.0	1.1	1.8	1.1	
		1.0	1.0	1.0	1.0	1.0	
		1.0				1.0	
		1.0	1.0	1.0	1.0	1.0	

(空白のセルは値0)

[図12]

10: 元図形パターン(ドーズ量付き)

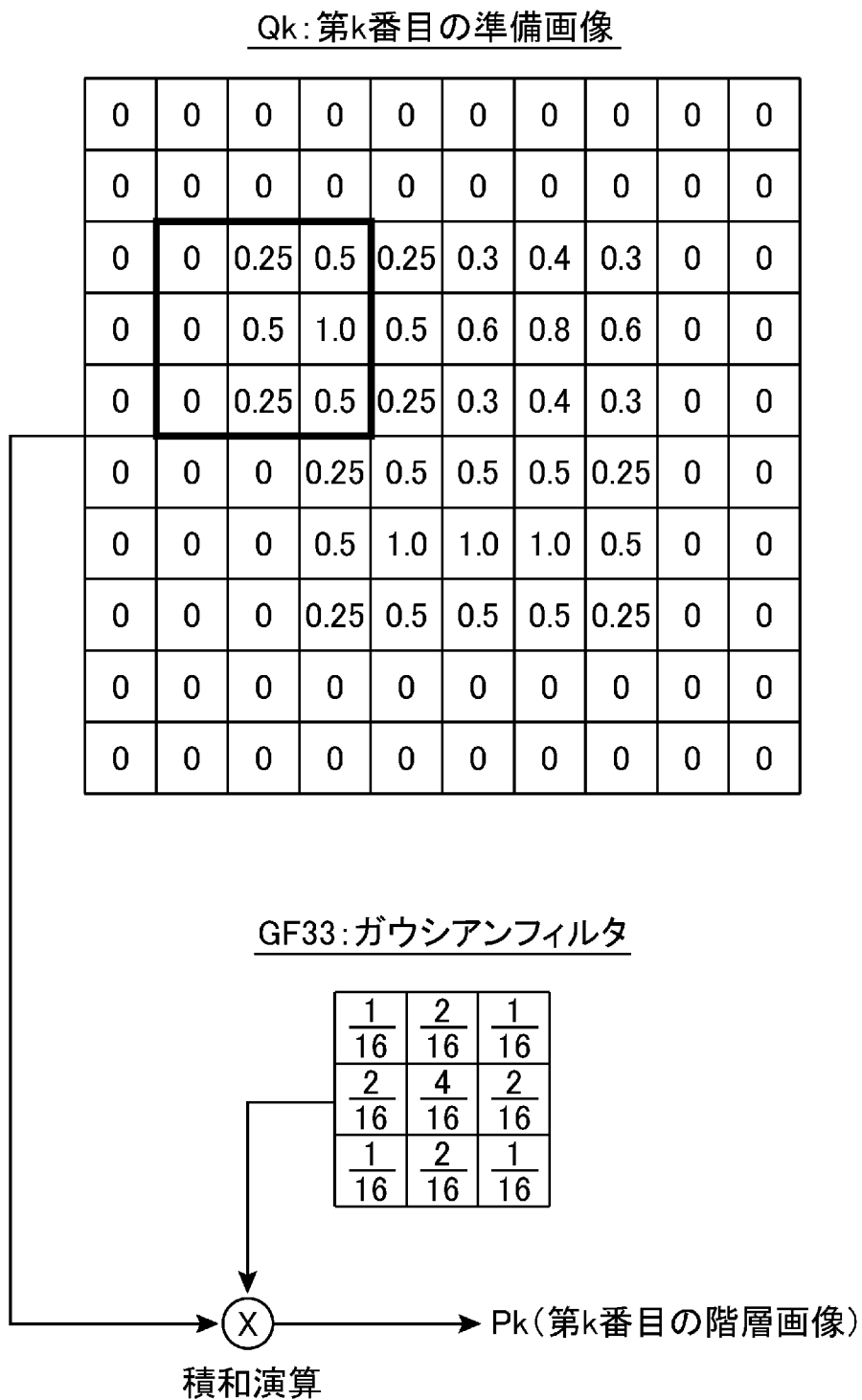
[図13]

M3: ドーズ密度マップ

	0.25	0.5	0.25	0.3	0.4	0.3	
	0.5	1.0	0.5	0.6	0.8	0.6	
	0.25	0.5	0.25	0.3	0.4	0.3	
		0.13	0.25	0.16	0.05	0.03	
		0.25	0.5	0.3	0.1	0.05	
		0.13	0.25	0.16	0.05	0.03	

(空白のセルは値0)

[図14]



[図15]

P_k: 第k番目の階層画像

0	0	0	0	0	0	0	0	0	0
0	0.016	0.063	0.094	0.081	0.078	0.088	0.063	0.019	0
0	0.063	0.25	0.375	0.325	0.313	0.35	0.25	0.075	0
0	0.094	0.375	0.563	0.488	0.469	0.525	0.375	0.113	0
0	0.063	0.258	0.407	0.374	0.351	0.368	0.257	0.077	0
0	0.016	0.094	0.22	0.277	0.231	0.158	0.089	0.026	0
0	0	0.048	0.189	0.293	0.228	0.105	0.039	0.01	0
0	0	0.032	0.126	0.196	0.153	0.071	0.026	0.007	0
0	0	0.008	0.032	0.049	0.039	0.018	0.007	0.002	0
0	0	0	0	0	0	0	0	0	0

[図16]

GF33: ガウシアンフィルタ

$\frac{1}{16}$	$\frac{2}{16}$	$\frac{1}{16}$
$\frac{2}{16}$	$\frac{4}{16}$	$\frac{2}{16}$
$\frac{1}{16}$	$\frac{2}{16}$	$\frac{1}{16}$

(a)

LF33: ラプラシアンフィルタ

-1	-1	-1
-1	8	-1
-1	-1	-1

(b)

[図17]

GF55: ガウシアンフィルタ

(a)

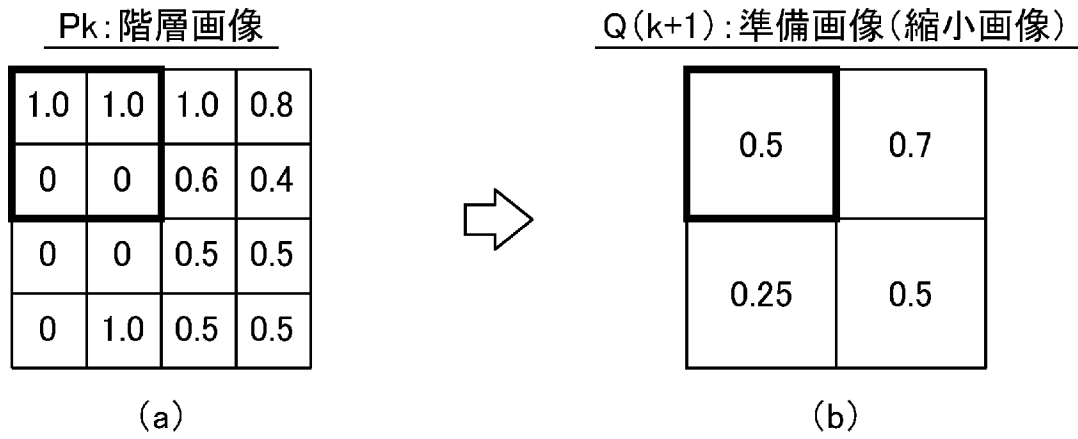
$\frac{1}{256}$	$\frac{4}{256}$	$\frac{6}{256}$	$\frac{4}{256}$	$\frac{1}{256}$
$\frac{4}{256}$	$\frac{16}{256}$	$\frac{24}{256}$	$\frac{16}{256}$	$\frac{4}{256}$
$\frac{6}{256}$	$\frac{24}{256}$	$\frac{36}{256}$	$\frac{24}{256}$	$\frac{6}{256}$
$\frac{4}{256}$	$\frac{16}{256}$	$\frac{24}{256}$	$\frac{16}{256}$	$\frac{4}{256}$
$\frac{1}{256}$	$\frac{4}{256}$	$\frac{6}{256}$	$\frac{4}{256}$	$\frac{1}{256}$

LF55: ラプラシアンフィルタ

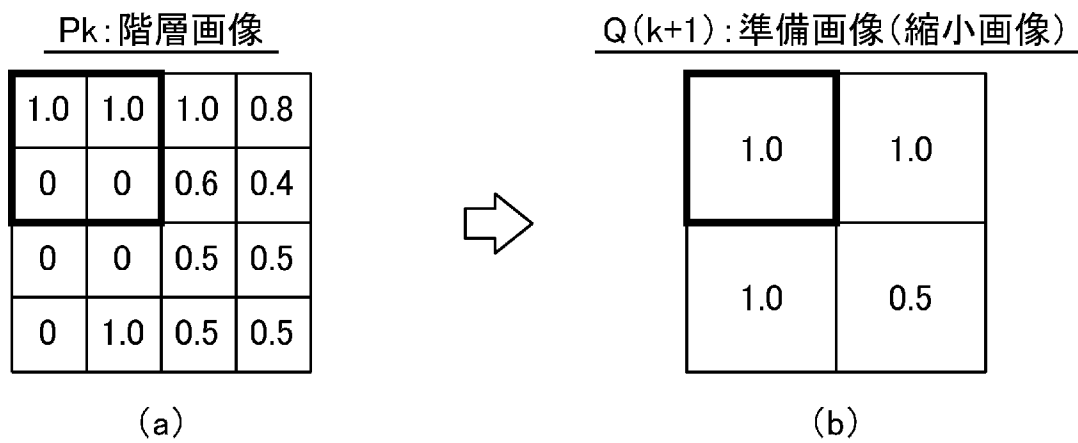
(b)

-1	-3	-4	-3	-1
-3	0	6	0	-3
-4	6	20	6	-4
-3	0	6	0	-3
-1	-3	-4	-3	-1

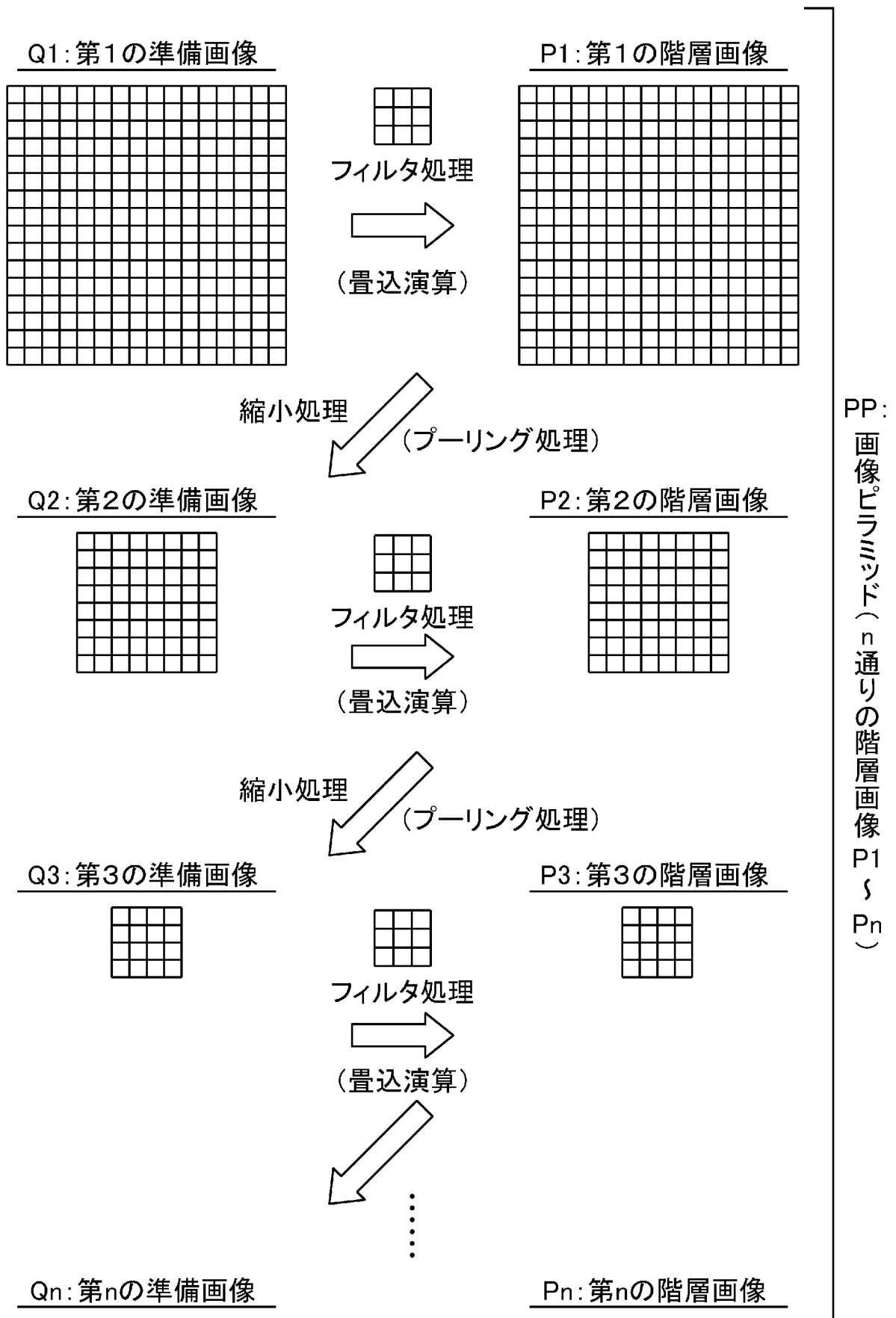
[図18]

アベレージ・プーリング処理

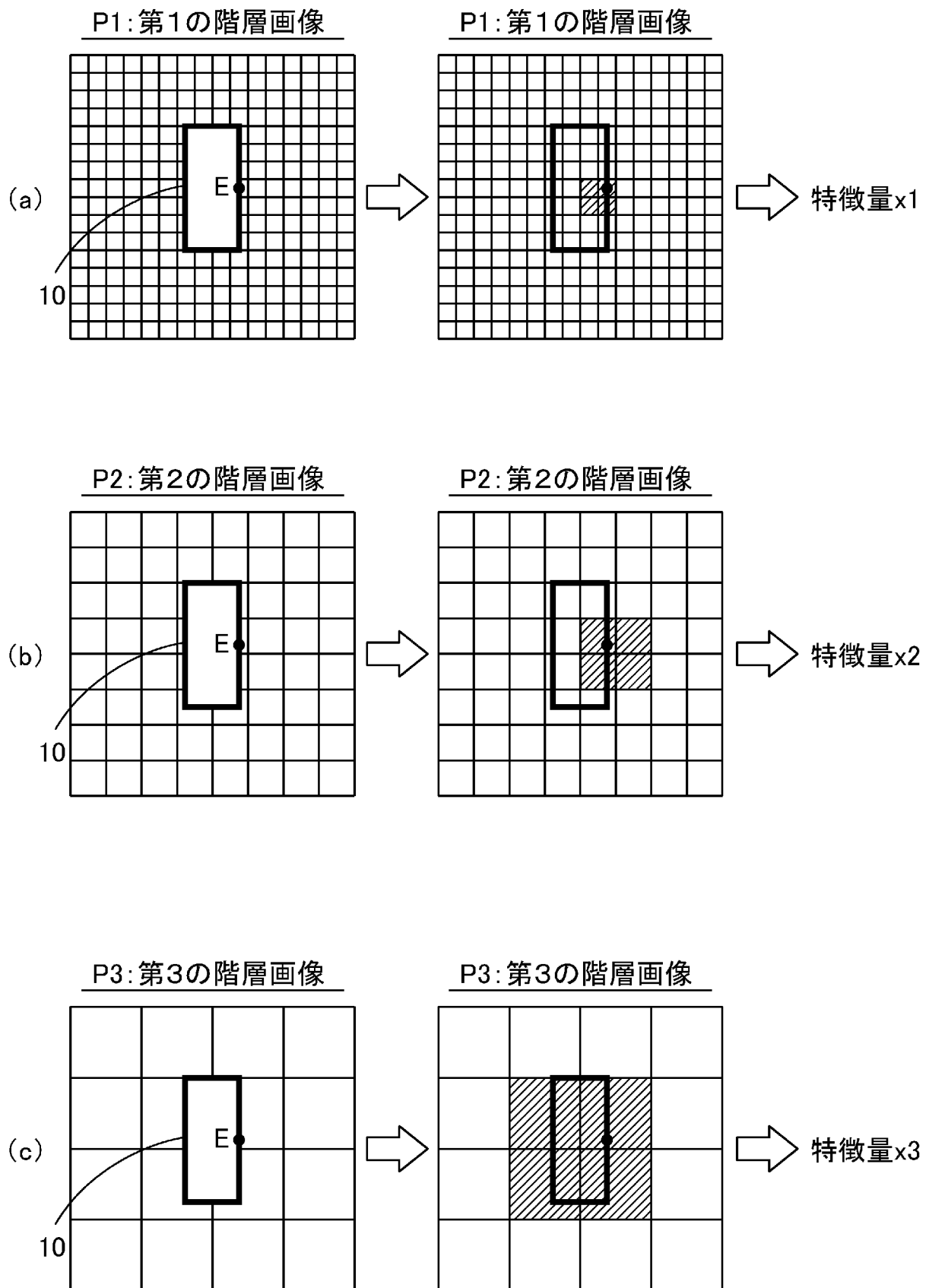
[図19]

マックス・プーリング処理

[図20]



[図21]

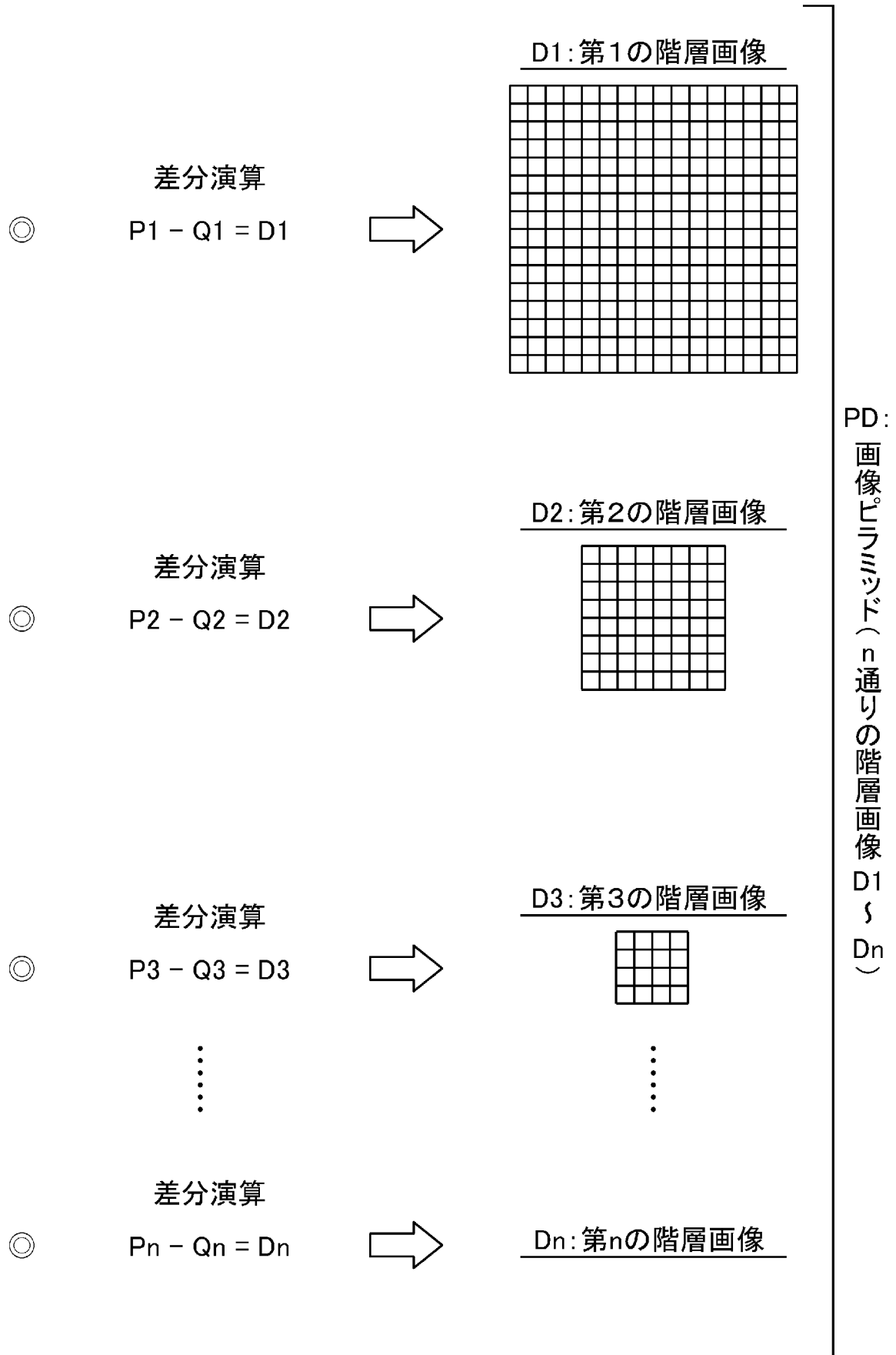


評価点Eの特徴量xは、各画素A～Dの画素値をそれぞれA～Dとし、画素ピッチuを1とすると、次の式で与えられる。

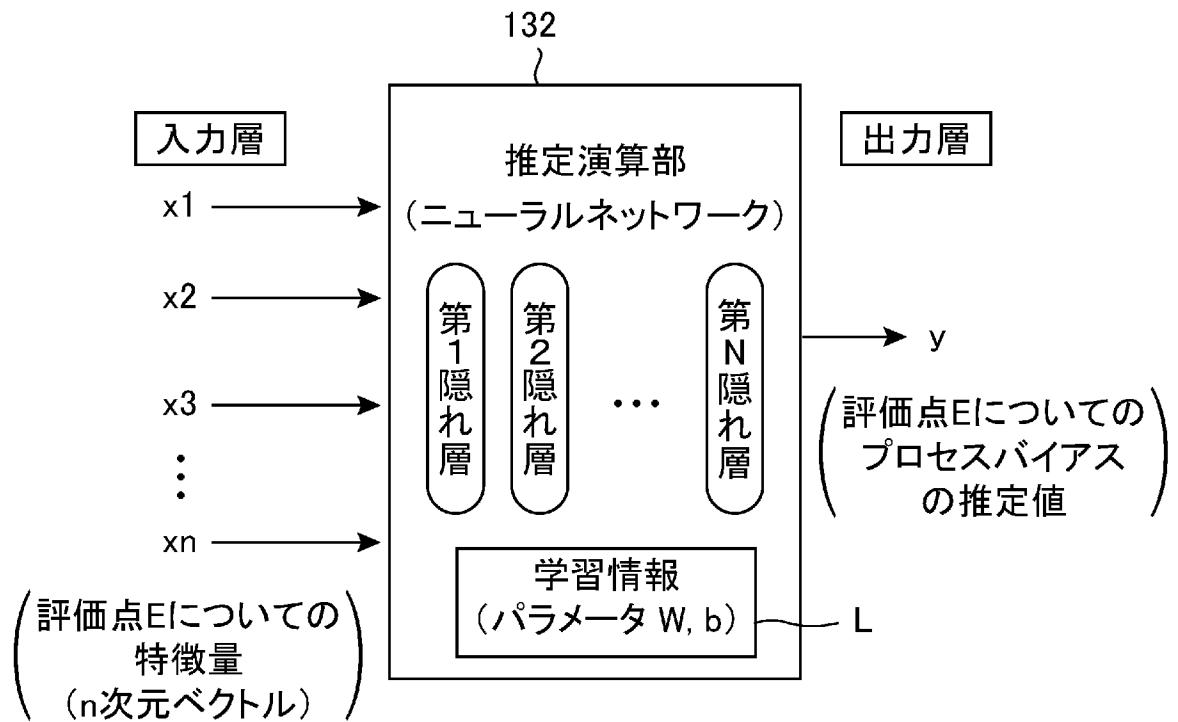
$$x = \frac{G \cdot d + H \cdot c}{2}$$

(b)

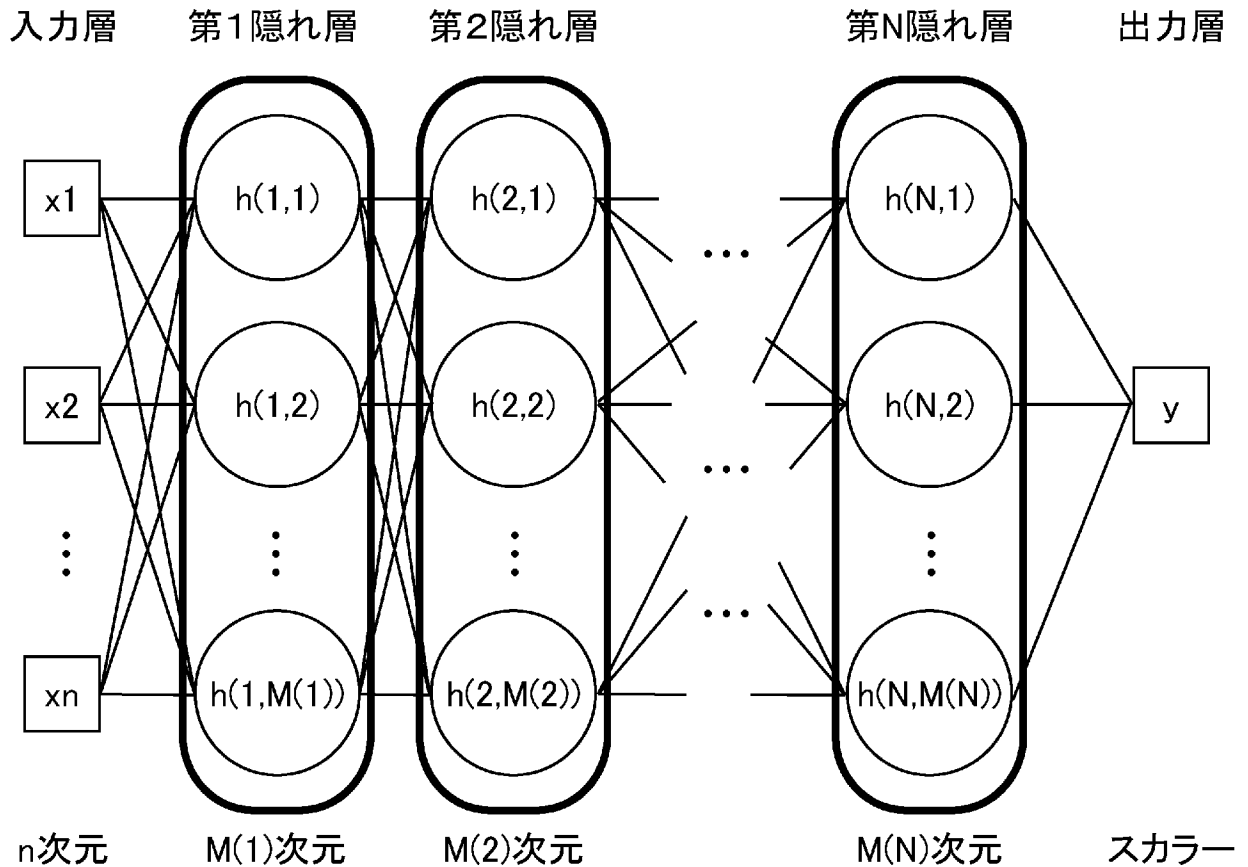
[図23]



[図24]



[図25]



[図26]

$$\begin{bmatrix} h(1, 1) \\ h(1, 2) \\ \vdots \\ h(1, M(1)) \end{bmatrix} = f(\xi)$$

但し、 $\xi = [W] \begin{bmatrix} x1 \\ x2 \\ \vdots \\ xn \end{bmatrix} + [b]$

ここで

$$[W] = \begin{bmatrix} W(1, 11) & W(1, 12) & \dots & W(1, 1n) \\ W(1, 21) & W(1, 22) & \dots & W(1, 2n) \\ \vdots & \vdots & & \vdots \\ W(1, M(1)1) & W(1, M(1)2) & \dots & W(1, M(1)n) \end{bmatrix}$$

$$[b] = \begin{bmatrix} b(1, 1) \\ b(1, 2) \\ \vdots \\ b(1, M(1)) \end{bmatrix}$$

[図27]

(a) シグモイド関数

$$f(\xi) = \frac{1}{1 + e^{-\xi}} = \frac{\tanh\left(\frac{\xi}{2}\right) + 1}{2}$$

(b) 正規化線形関数

(ReLU: Rectified Linear Unit, Rectifier)

$$f(\xi) = \max(0, \xi)$$

(c) 正規化線形関数

(Leaky ReLU)

$$f(\xi) = \max(k\xi, \xi) \quad k < 1$$

[図28]

$$\begin{bmatrix} h(i+1, 1) \\ h(i+1, 2) \\ \vdots \\ h(i+1, M(i+1)) \end{bmatrix} = f(\xi)$$

$$\text{但し、} \quad \xi = [W] \begin{bmatrix} h(i, 1) \\ h(i, 2) \\ \vdots \\ h(i, M(i)) \end{bmatrix} + [b]$$

ここで

$$[W] = \begin{bmatrix} W(i+1, 11) & W(i+1, 12) & \cdots & W(i+1, 1M(i)) \\ W(i+1, 21) & W(i+1, 22) & \cdots & W(i+1, 2M(i)) \\ \vdots & \vdots & & \vdots \\ W(i+1, M(i+1)1) & W(i+1, M(i+1)2) & \cdots & W(i+1, M(i+1)M(i)) \end{bmatrix}$$

$$[b] = \begin{bmatrix} b(i+1, 1) \\ b(i+1, 2) \\ \vdots \\ b(i+1, M(i+1)) \end{bmatrix}$$

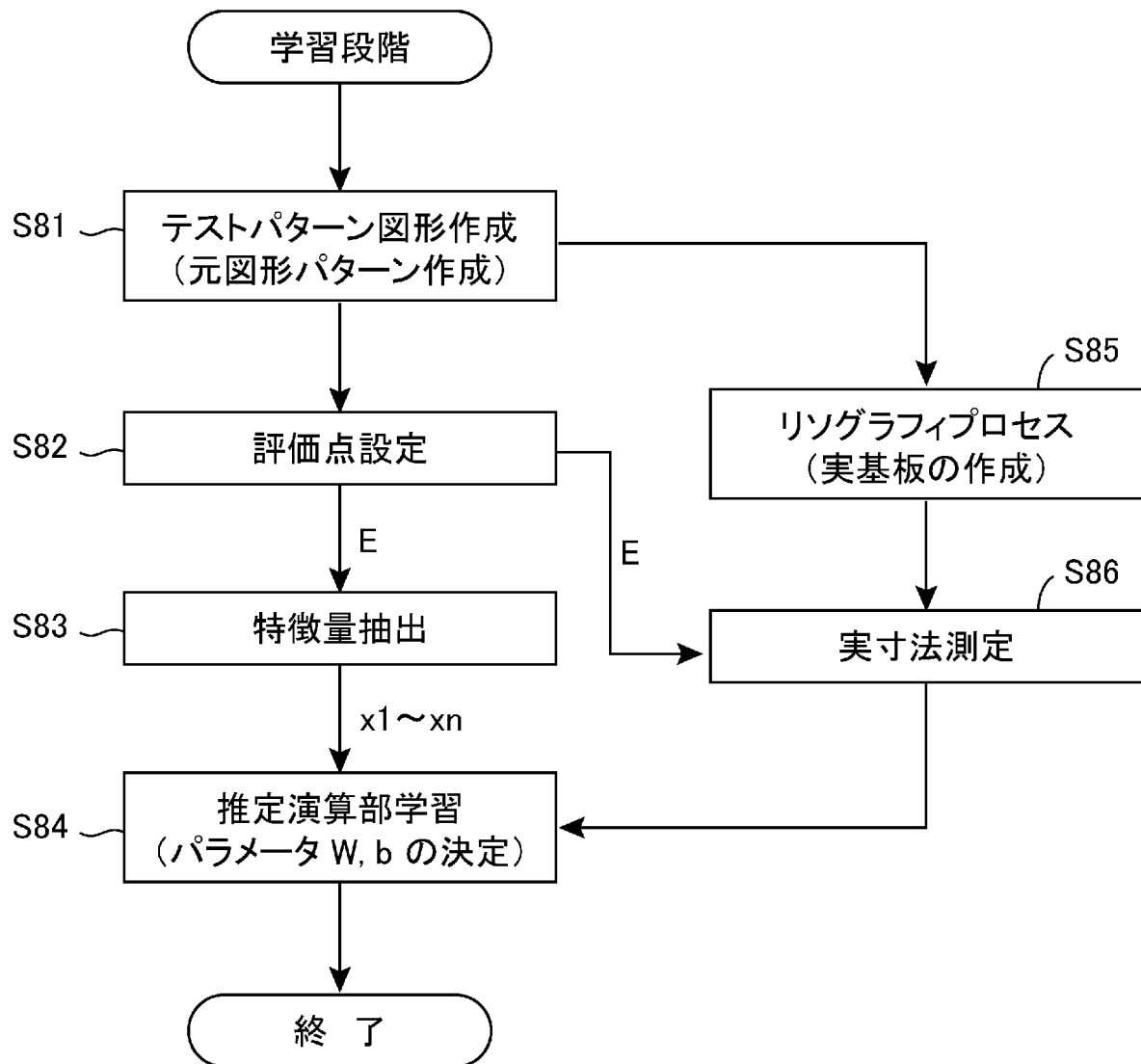
[図29]

$$y = [W] \begin{bmatrix} h(N, 1) \\ h(N, 2) \\ \vdots \\ h(N, M(N)) \end{bmatrix} + b(N+1)$$

ここで

$$[W] = \begin{bmatrix} W(N+1, 11) & W(N+1, 12) & \cdots & W(N+1, 1M(N)) \end{bmatrix}$$

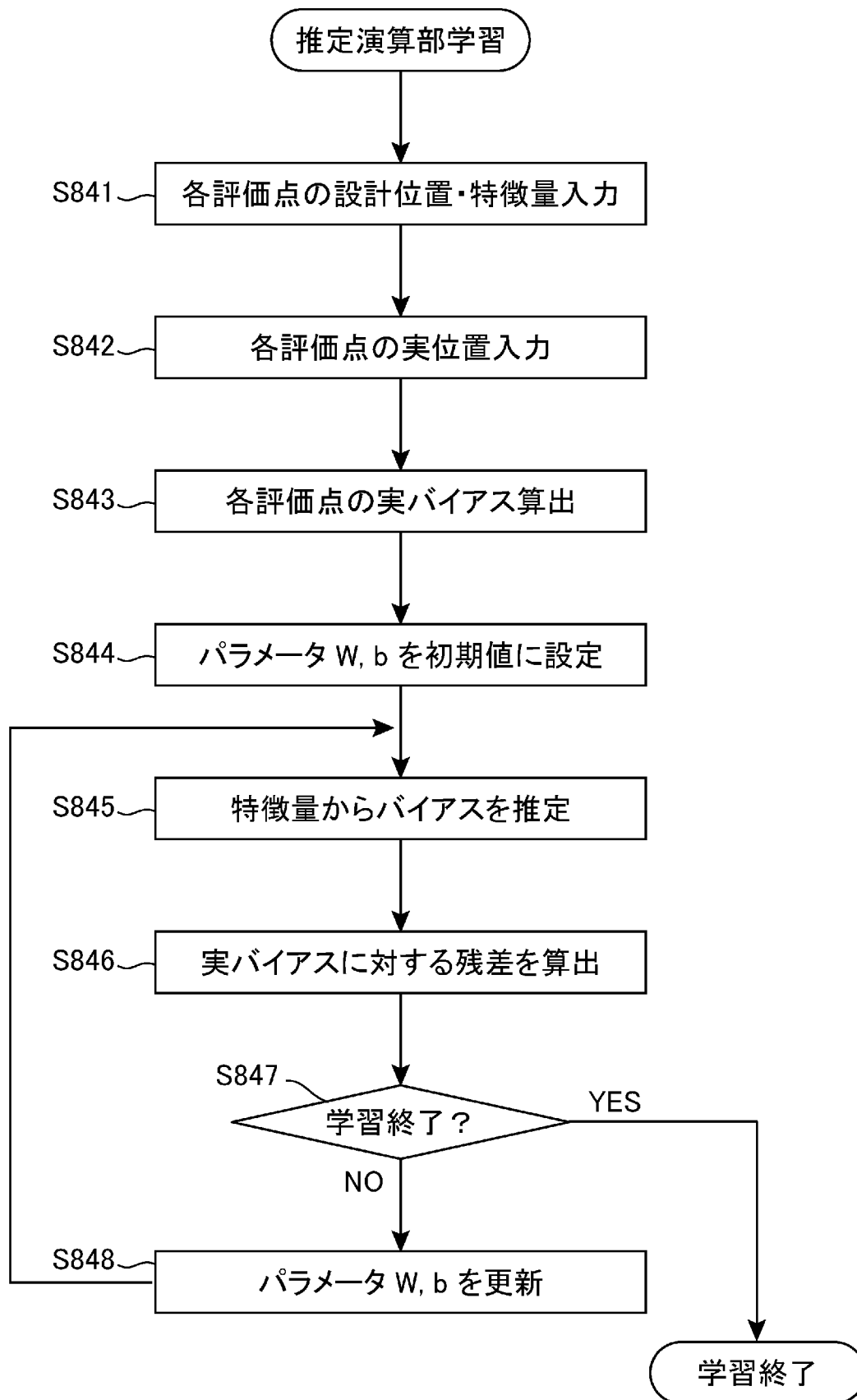
[図30]



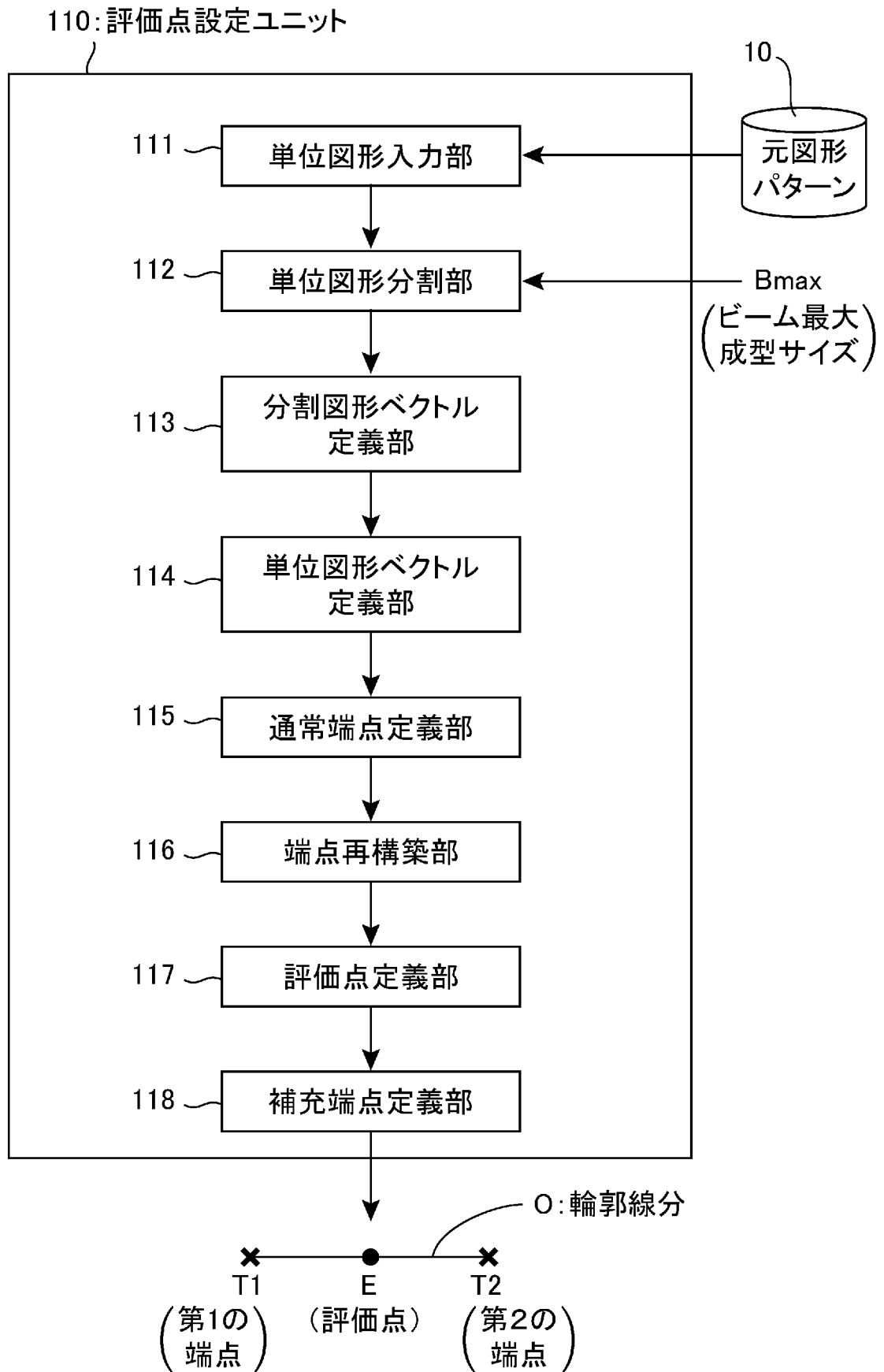
計算機上で実行
するプロセス

実基板上で実行
するプロセス

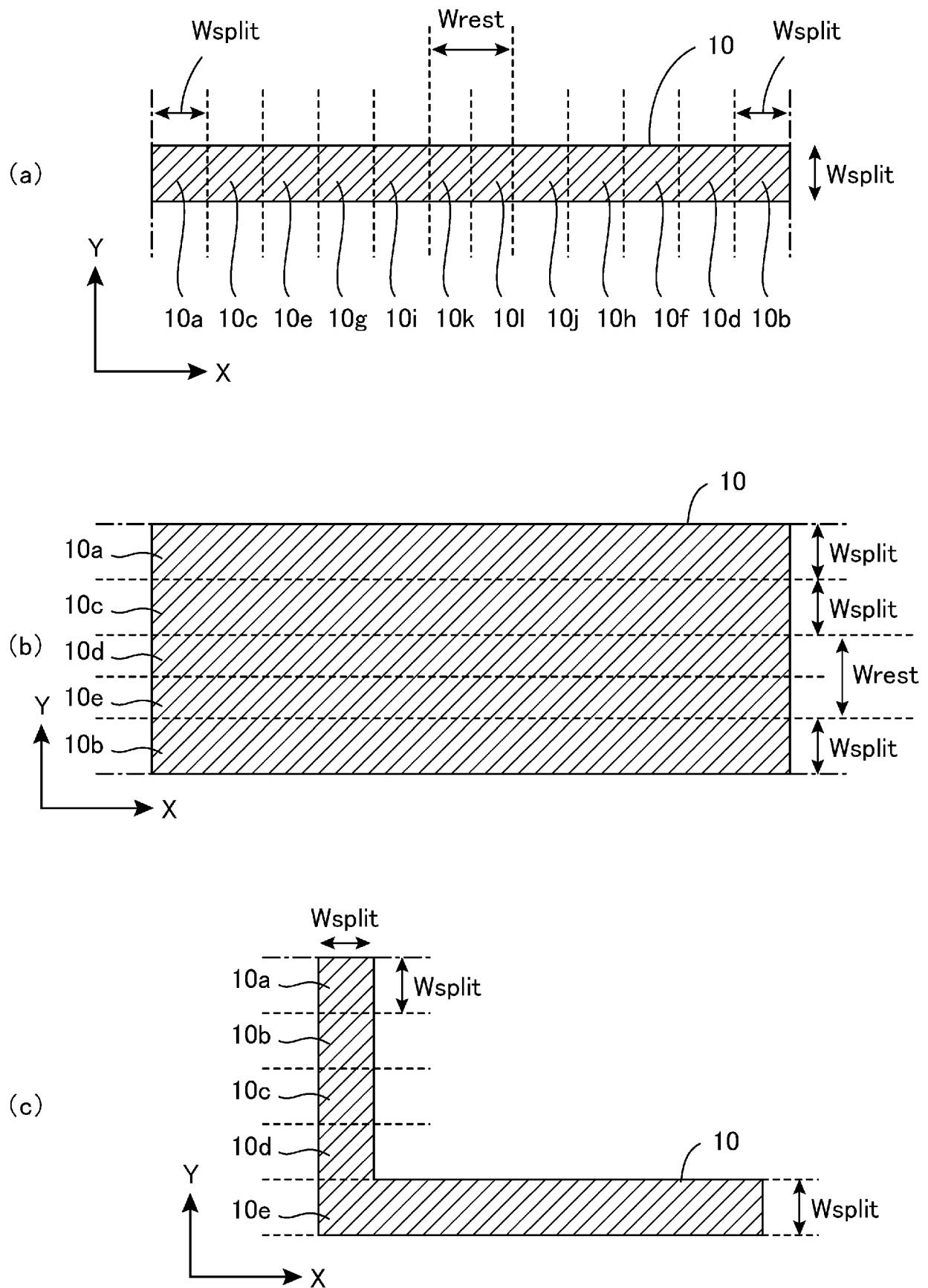
[図31]



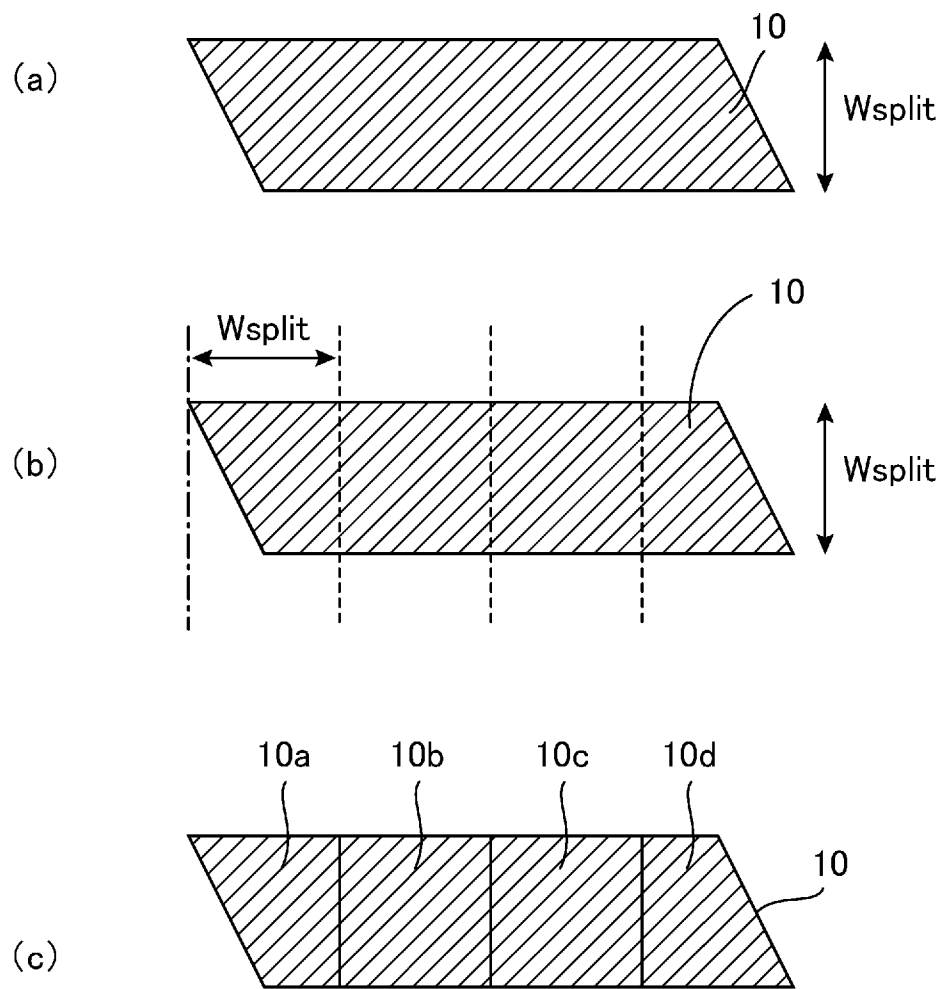
[図32]



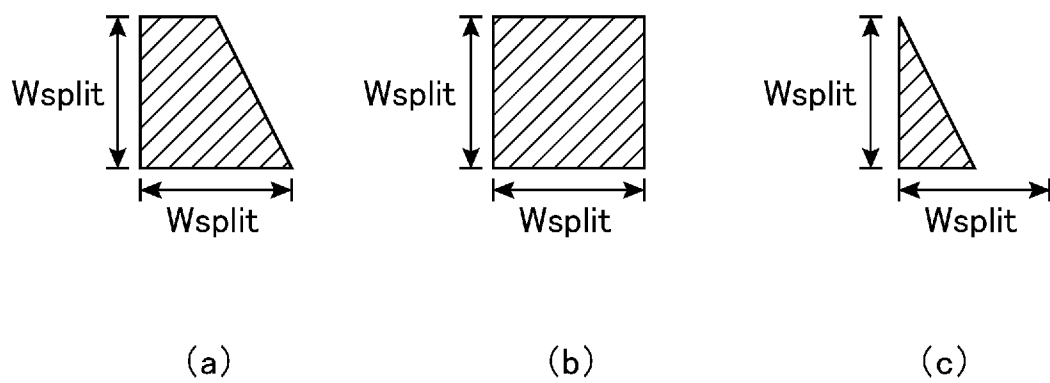
[図33]



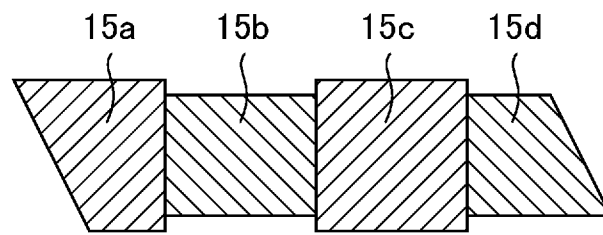
[図34]



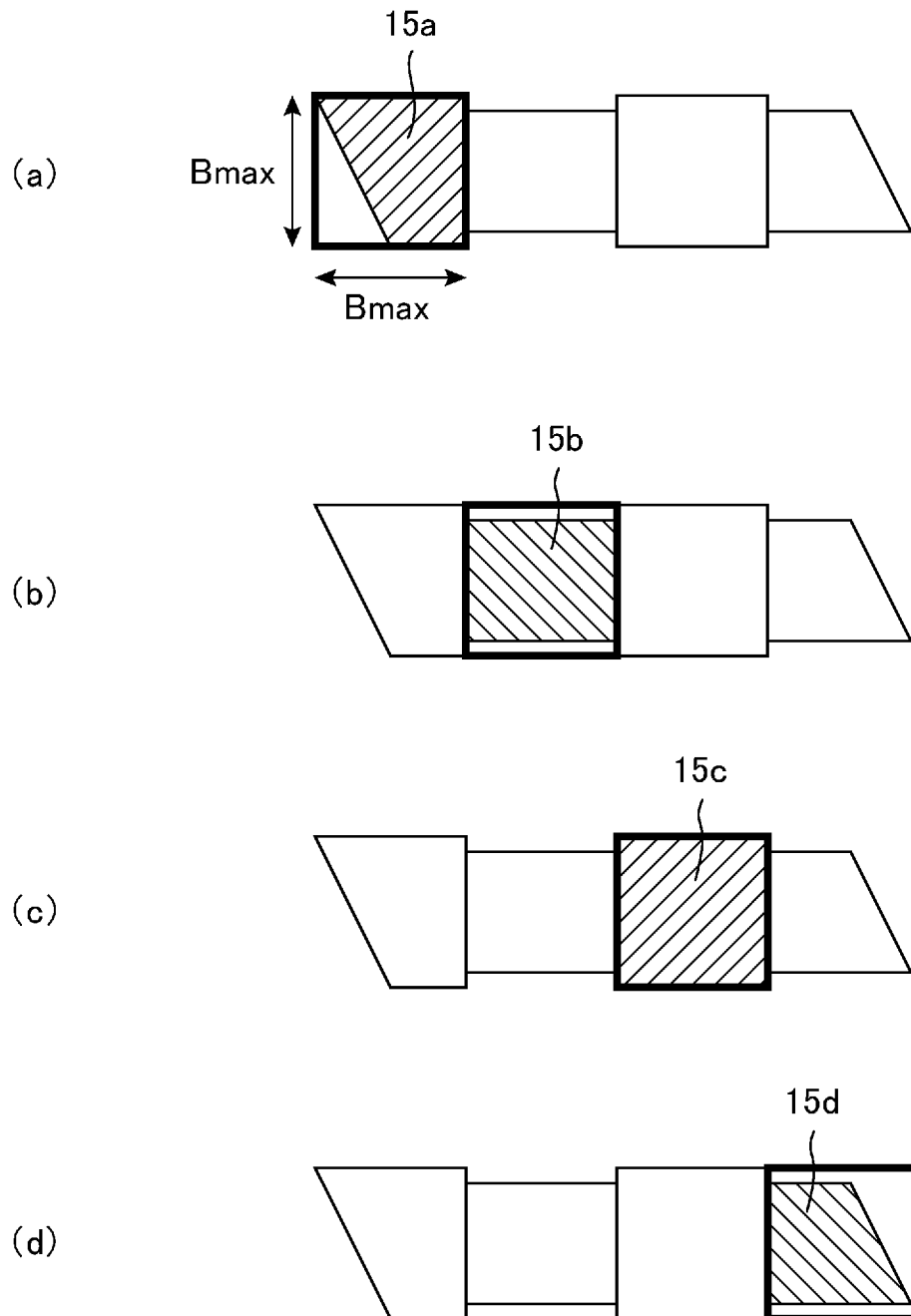
[図35]



[図36]

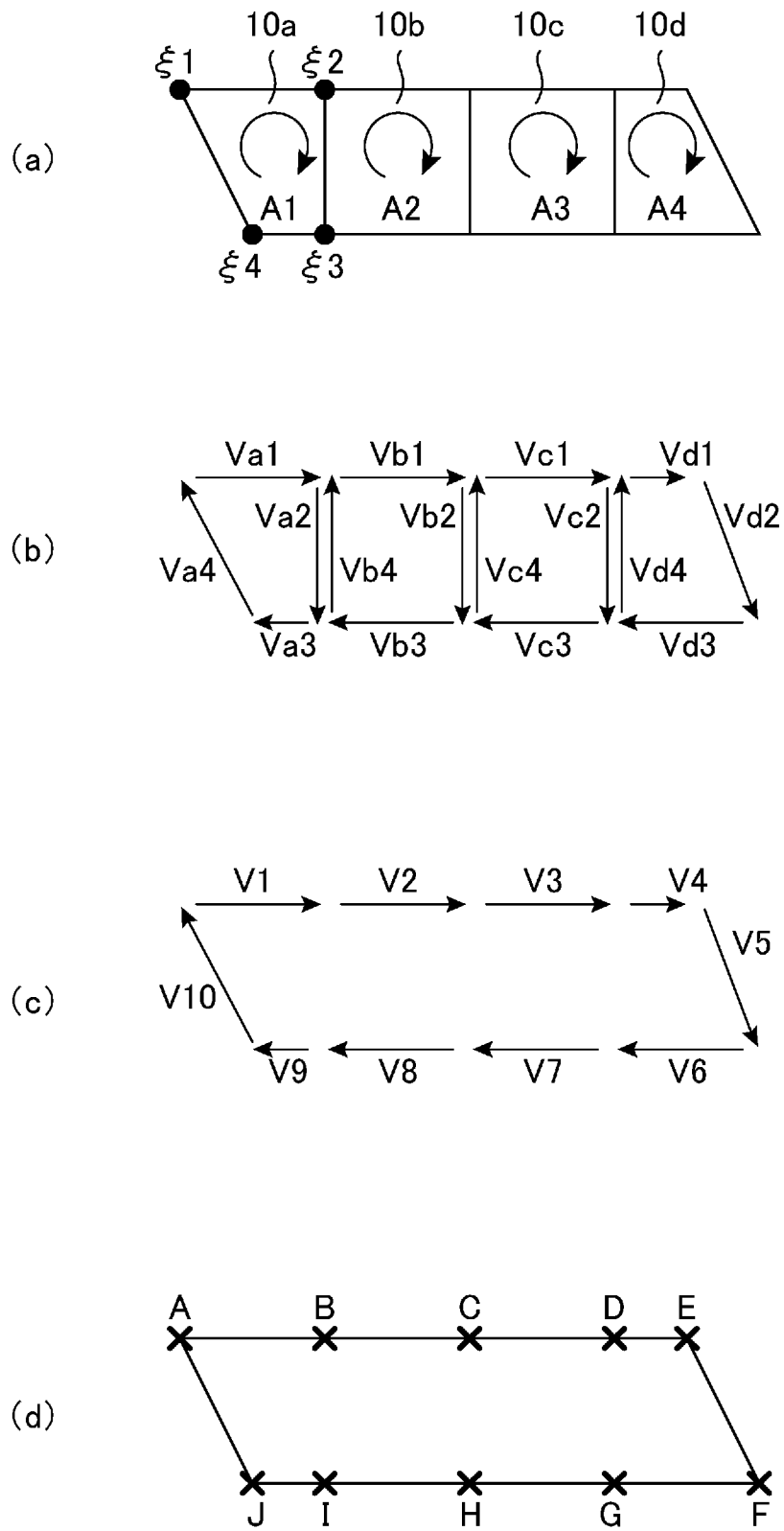
15: 補正後の単位図形

[図37]



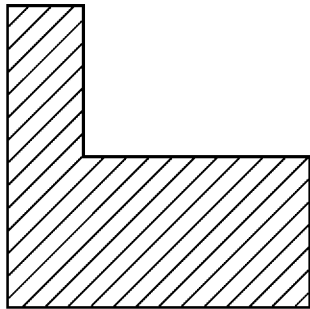
[図38]

10: 単位図形

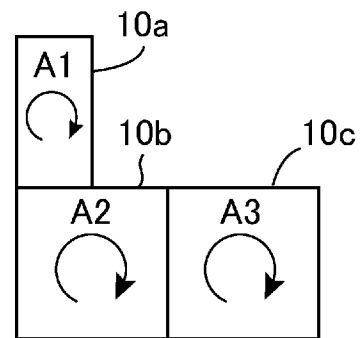


[図39]

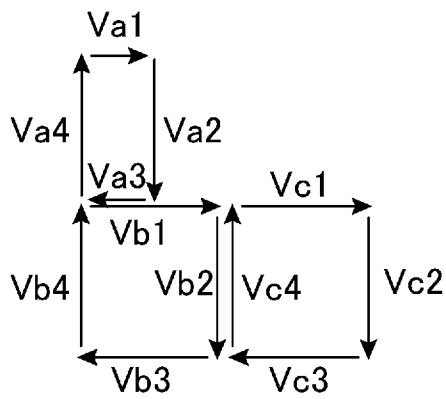
10: 単位図形



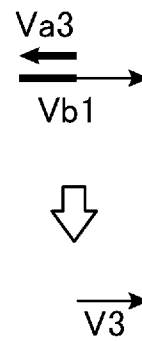
(a)



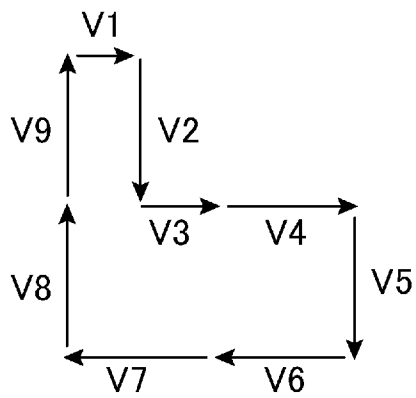
(b)



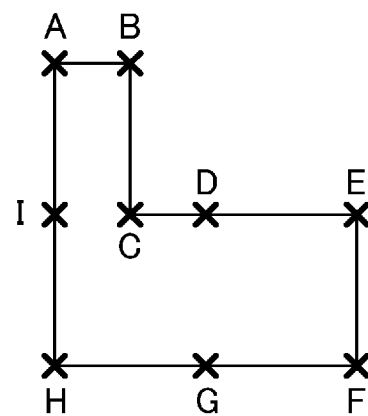
(c)



(d)



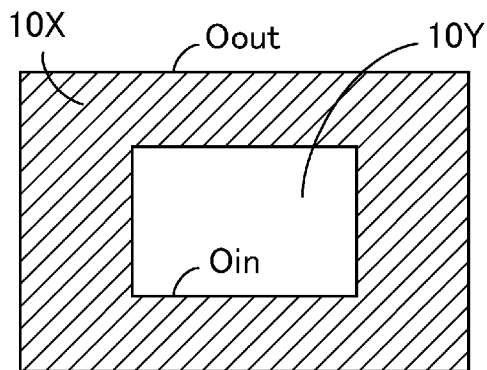
(e)



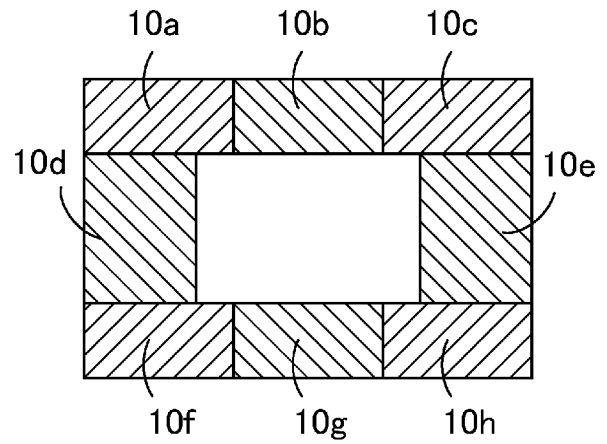
(f)

[図40]

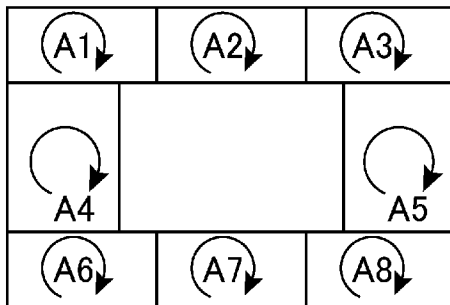
10: 単位図形



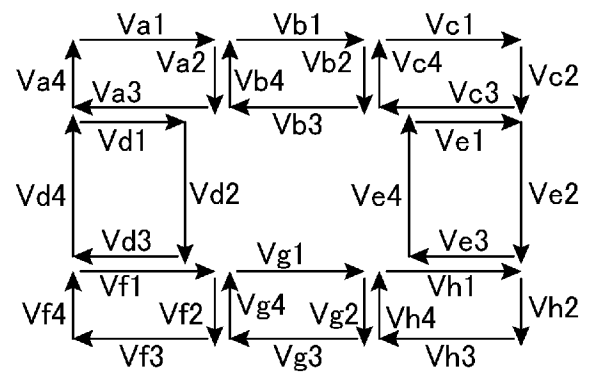
(a)



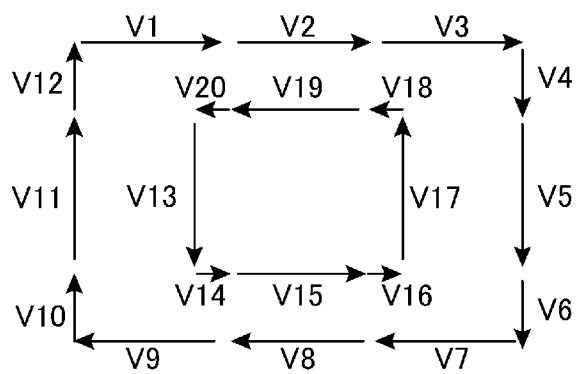
(b)



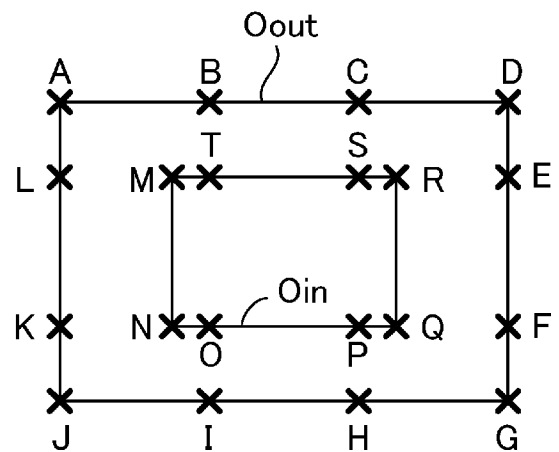
(c)



(d)

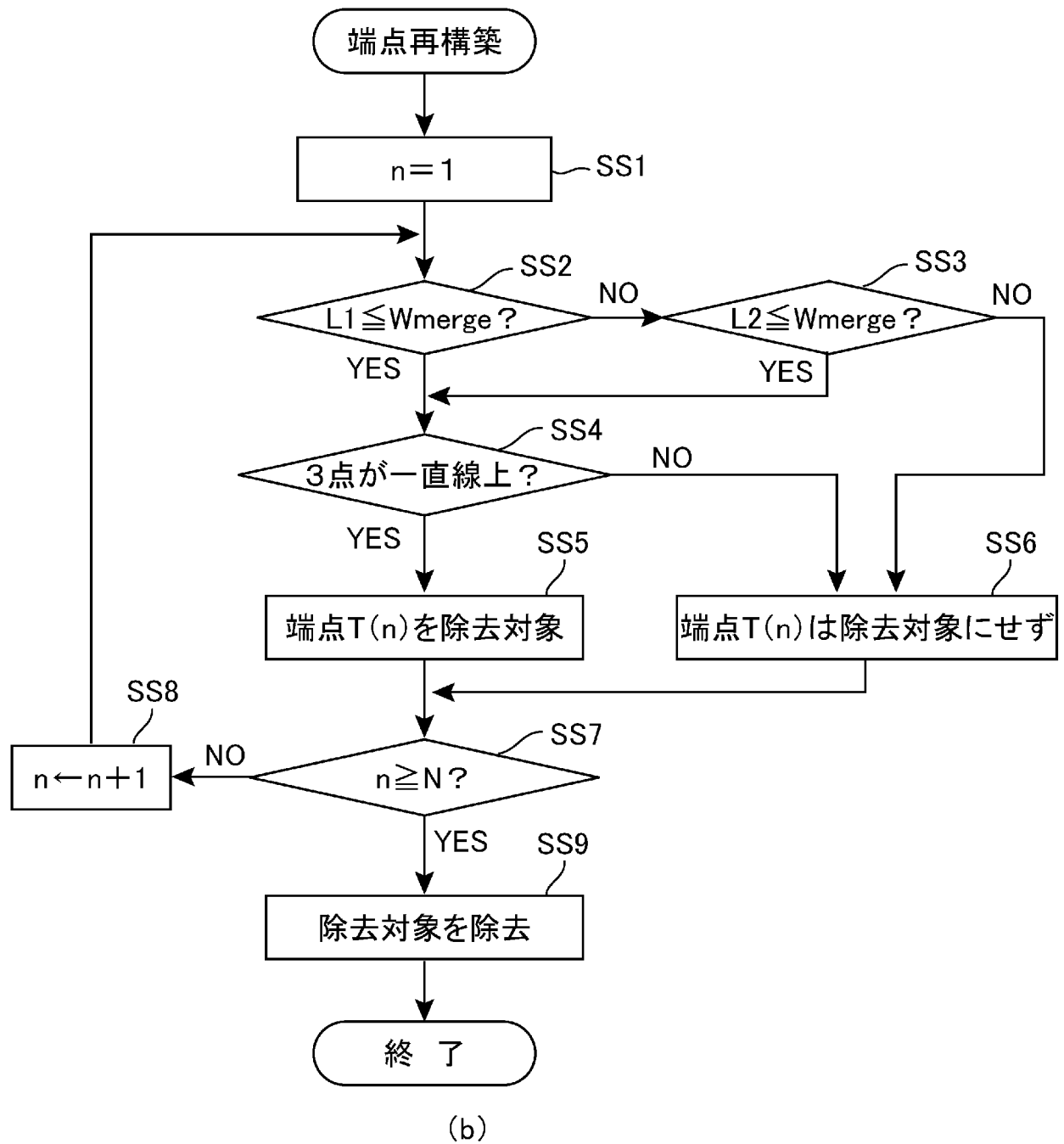
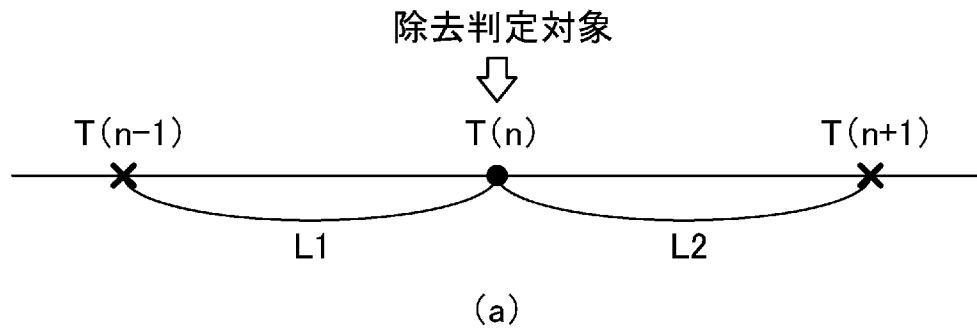


(e)

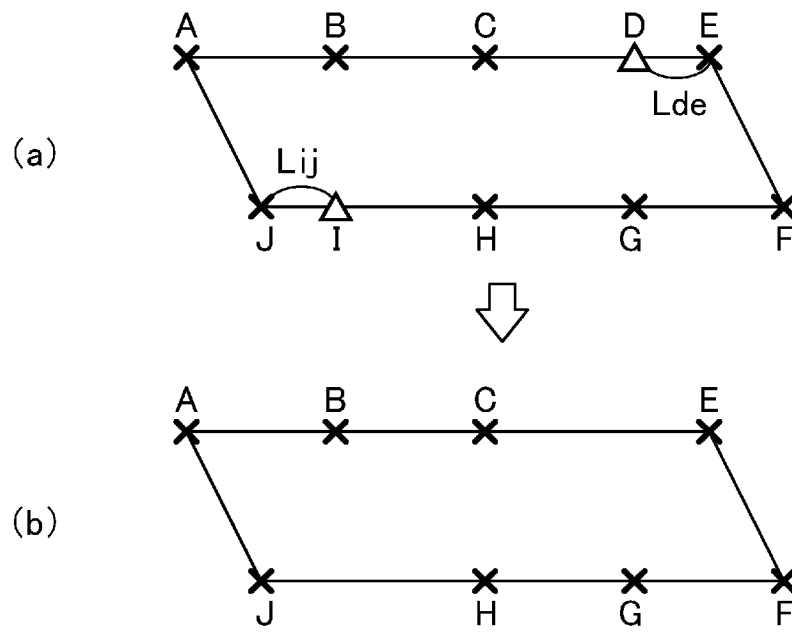


(f)

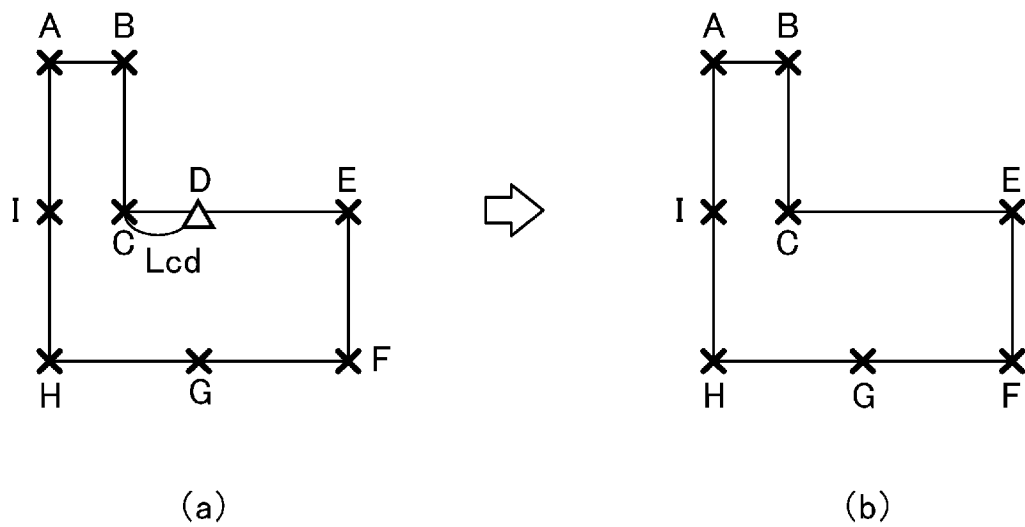
[図41]



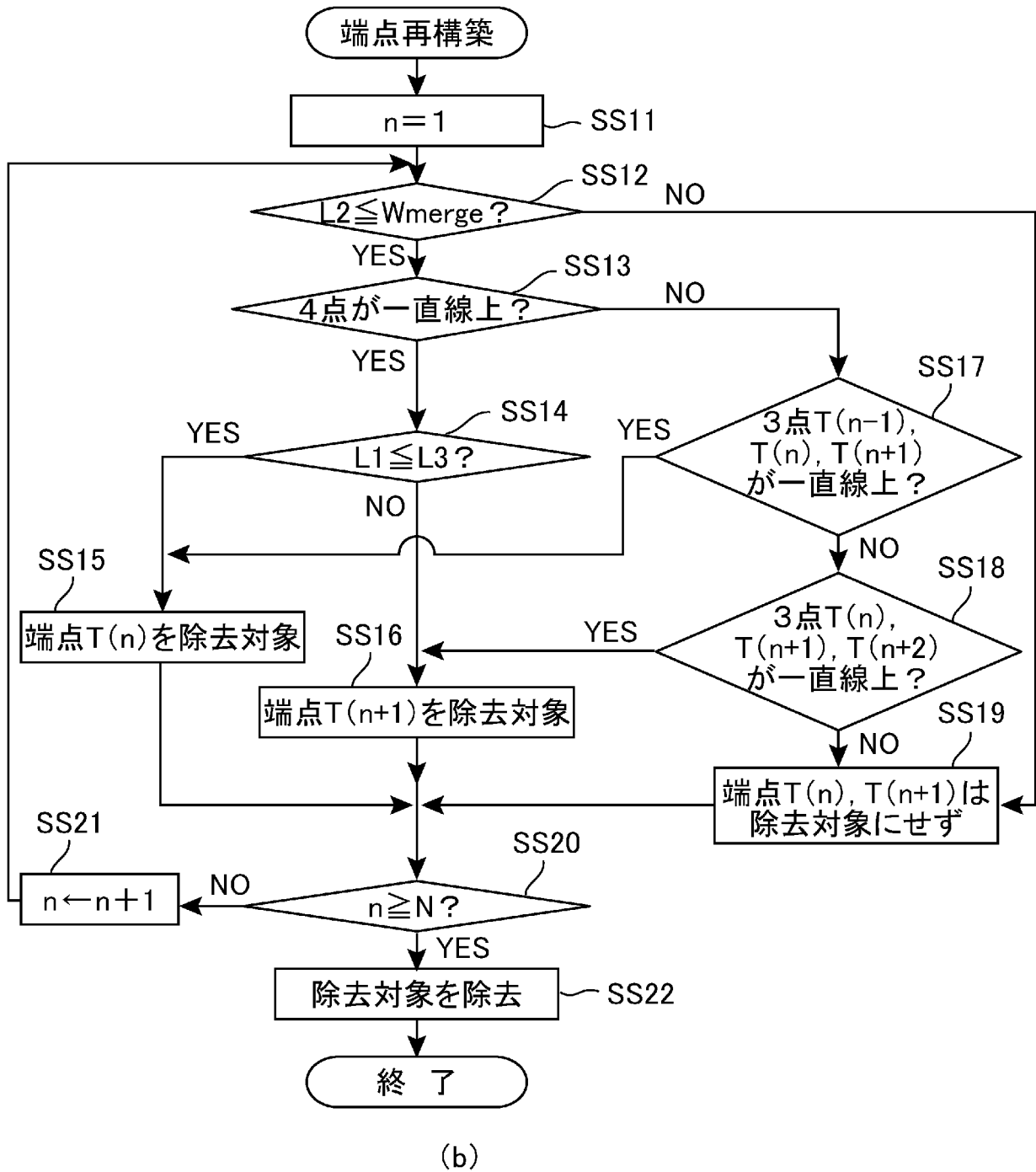
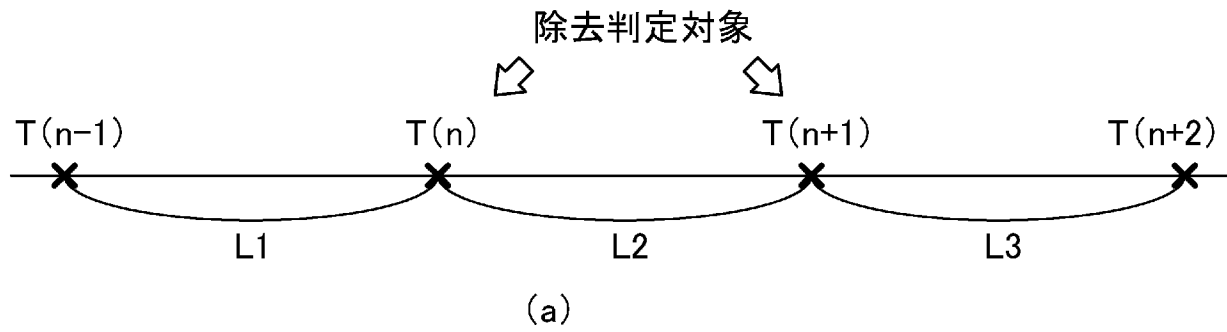
[図42]



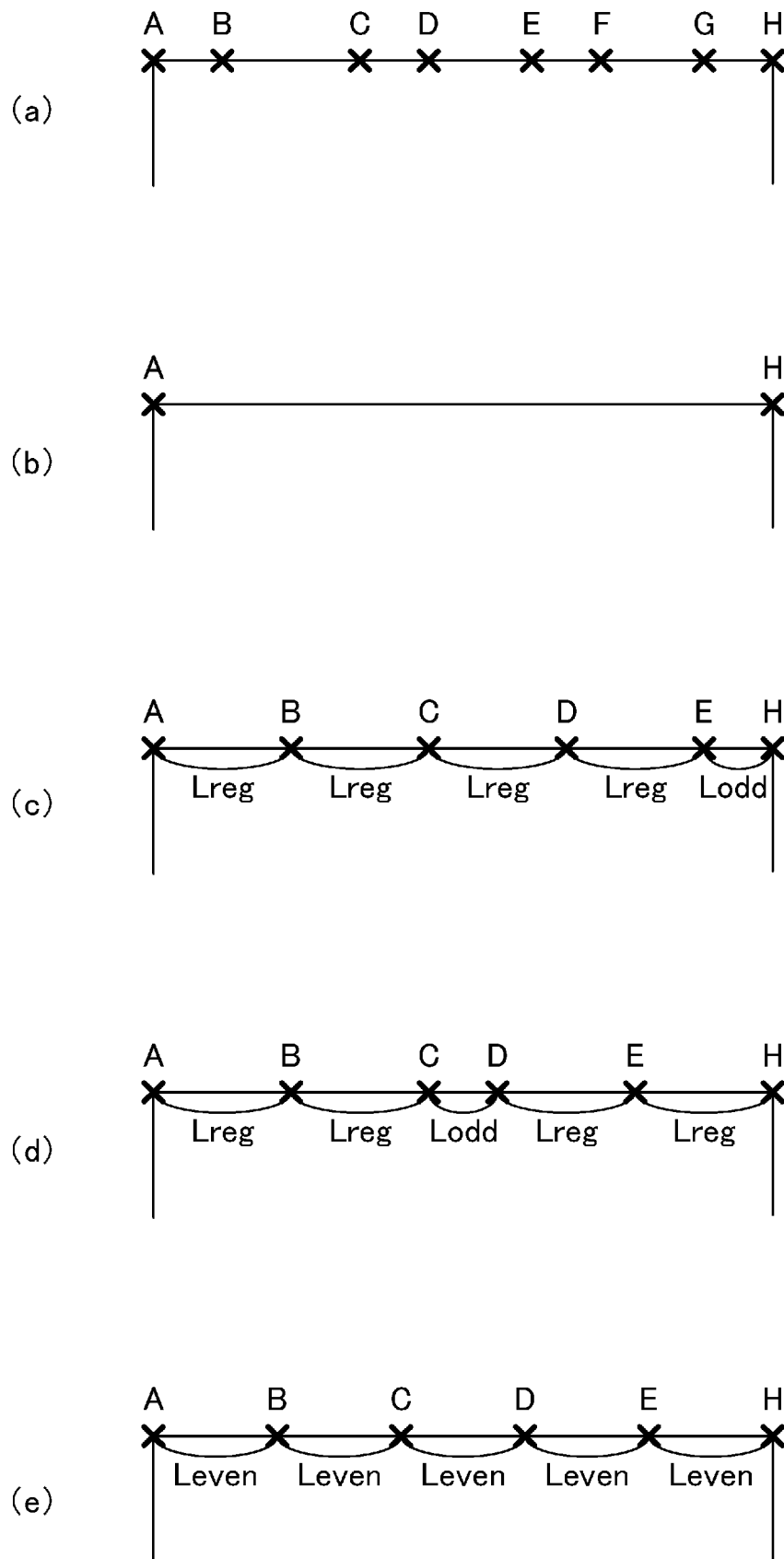
[図43]



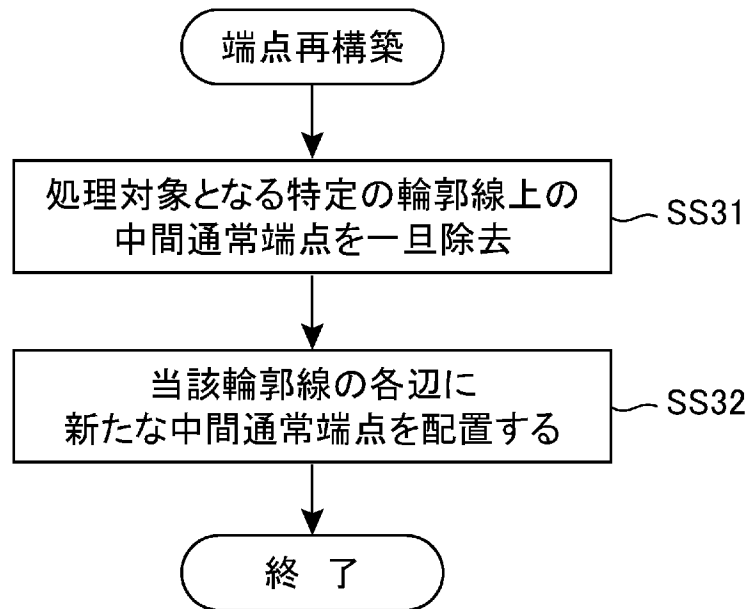
[図44]



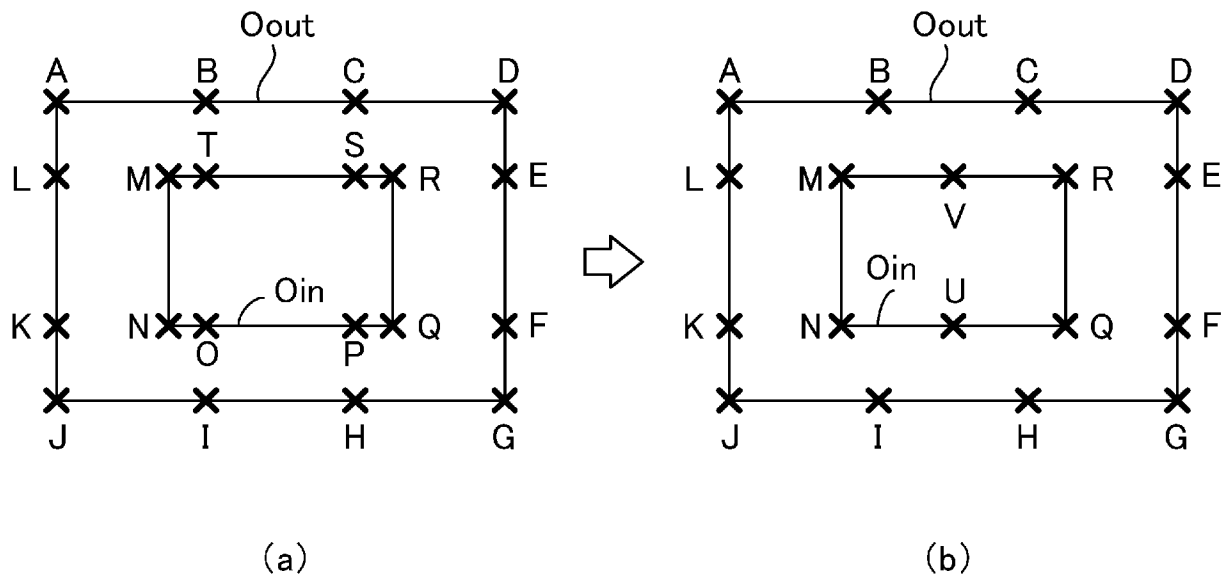
[図45]



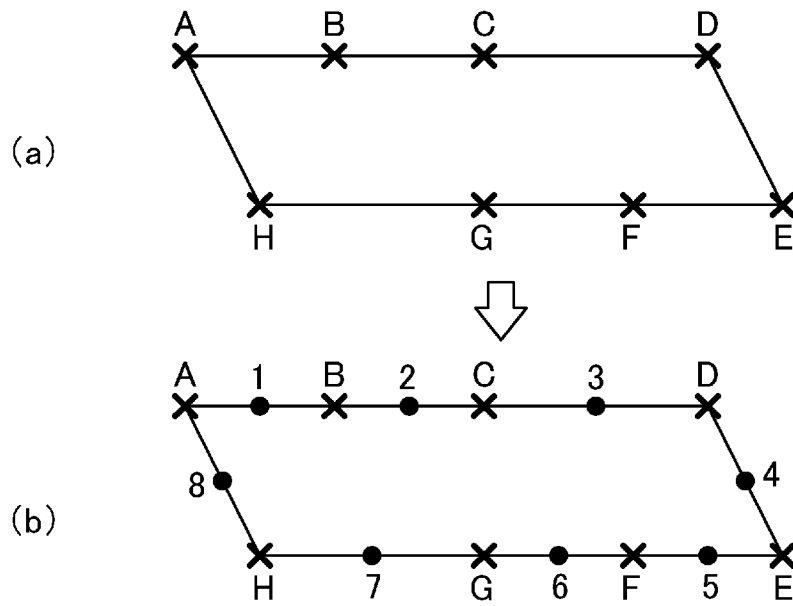
[図46]



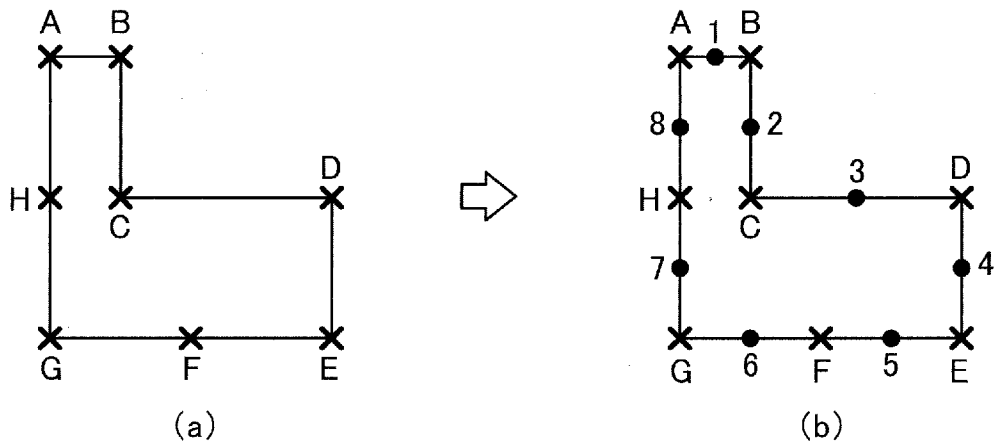
[図47]



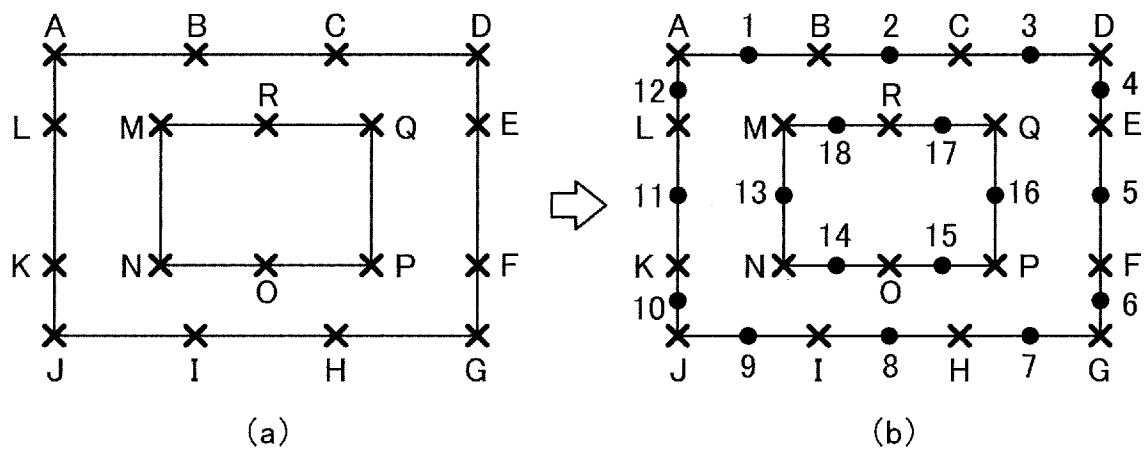
[図48]



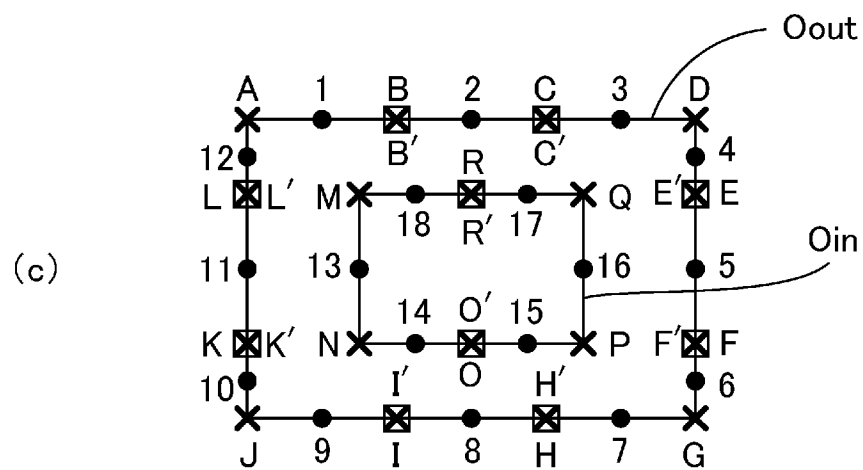
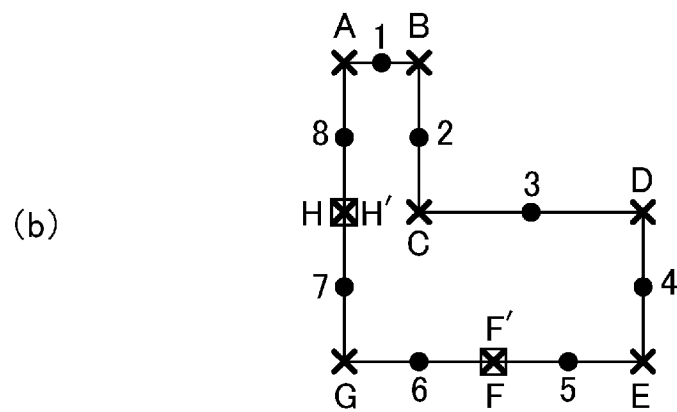
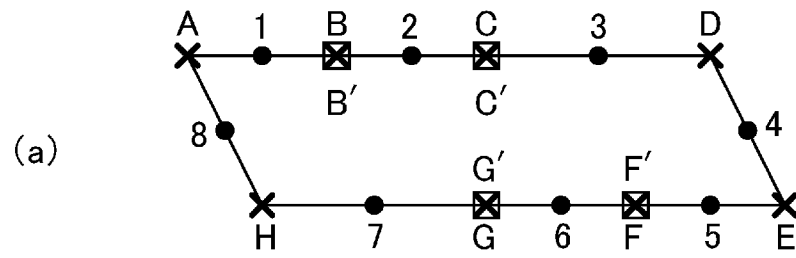
[図49]



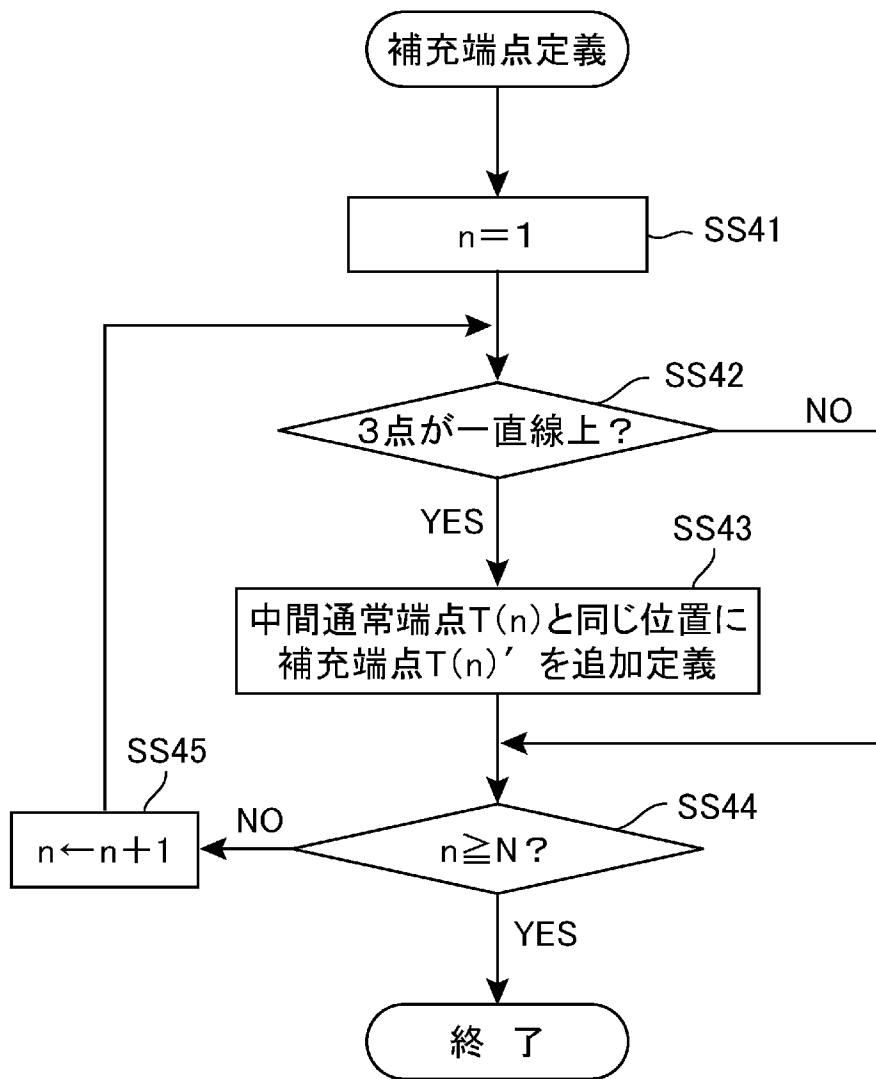
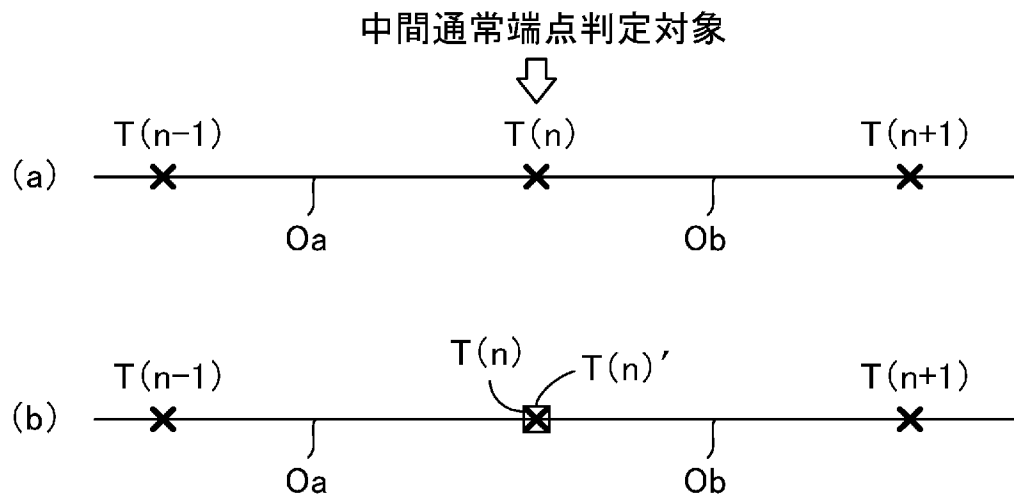
[図50]



[図51]

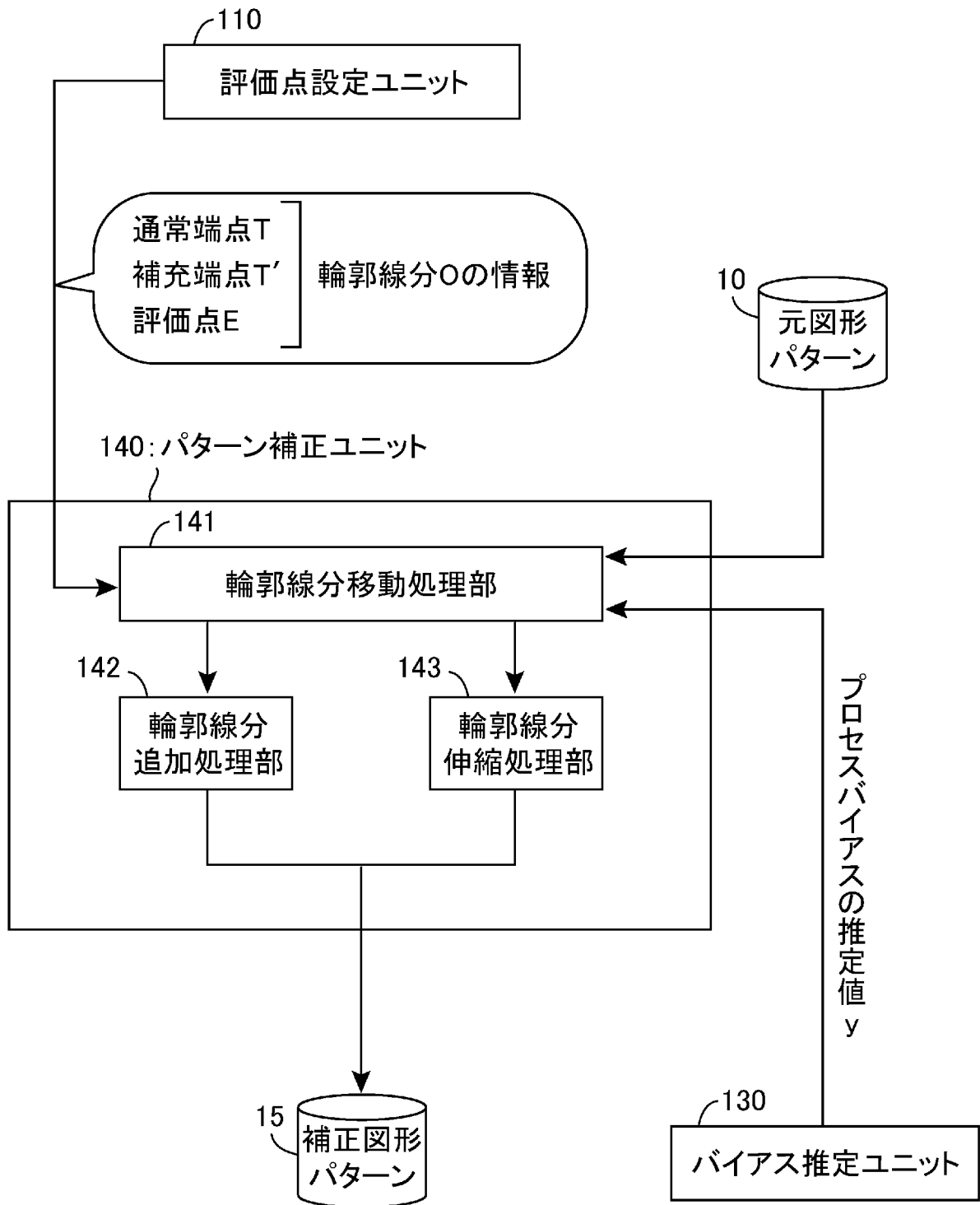


[図52]



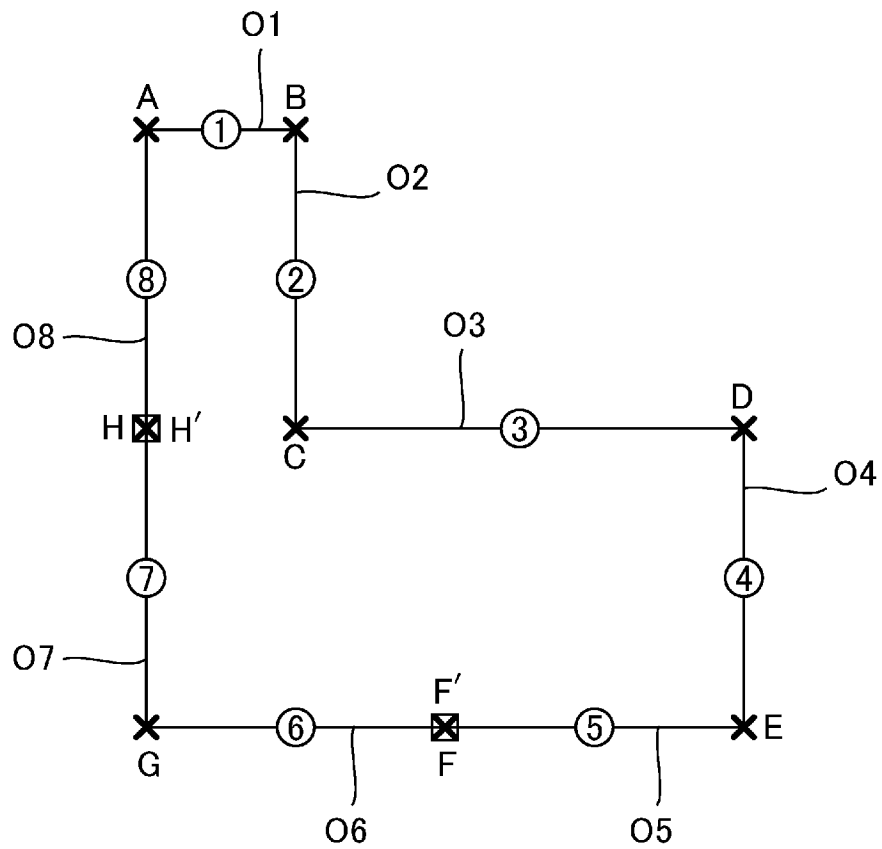
(c)

[図53]



[図54]

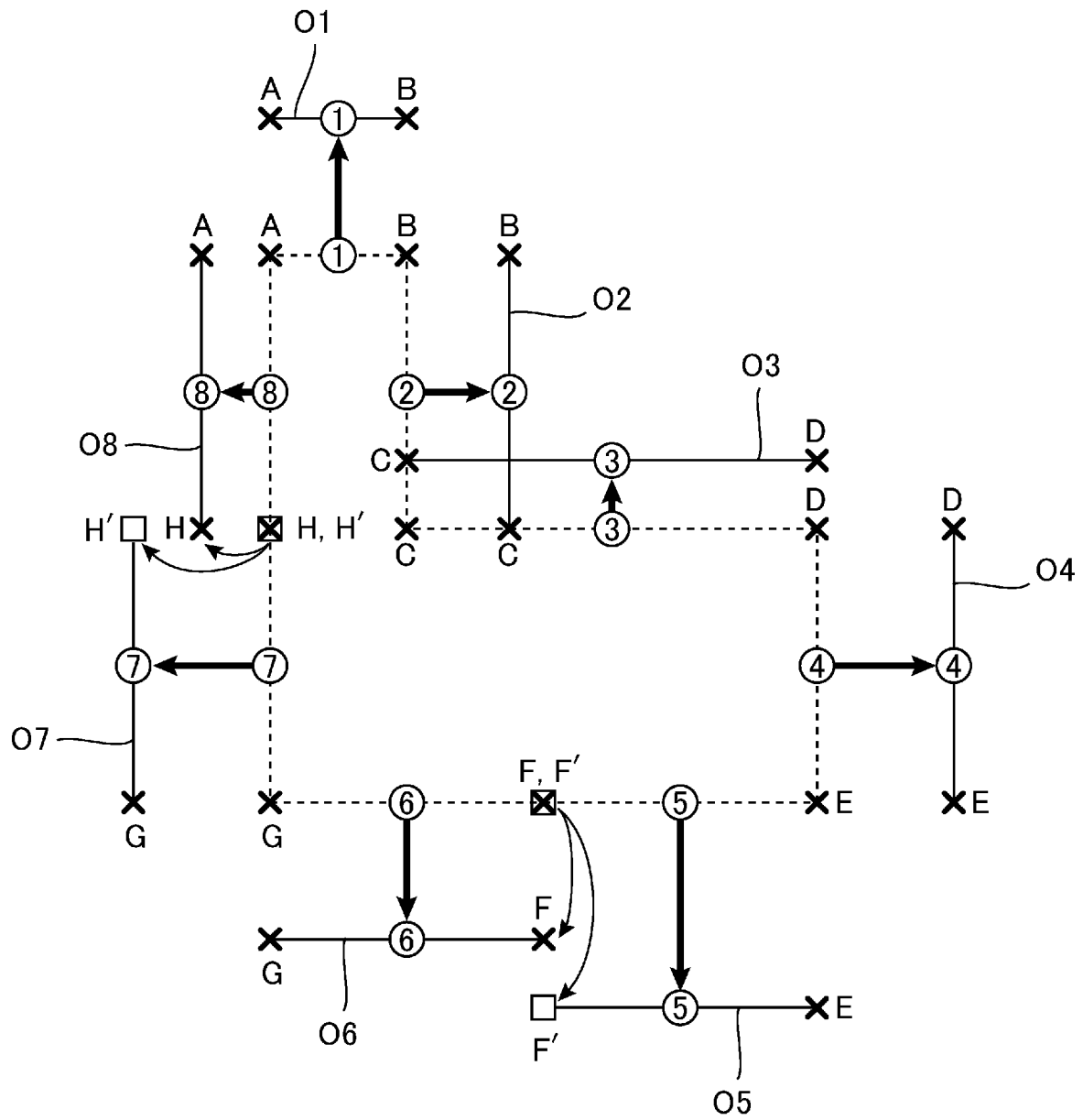
10: 元図形パターン(補正前の単位図形)



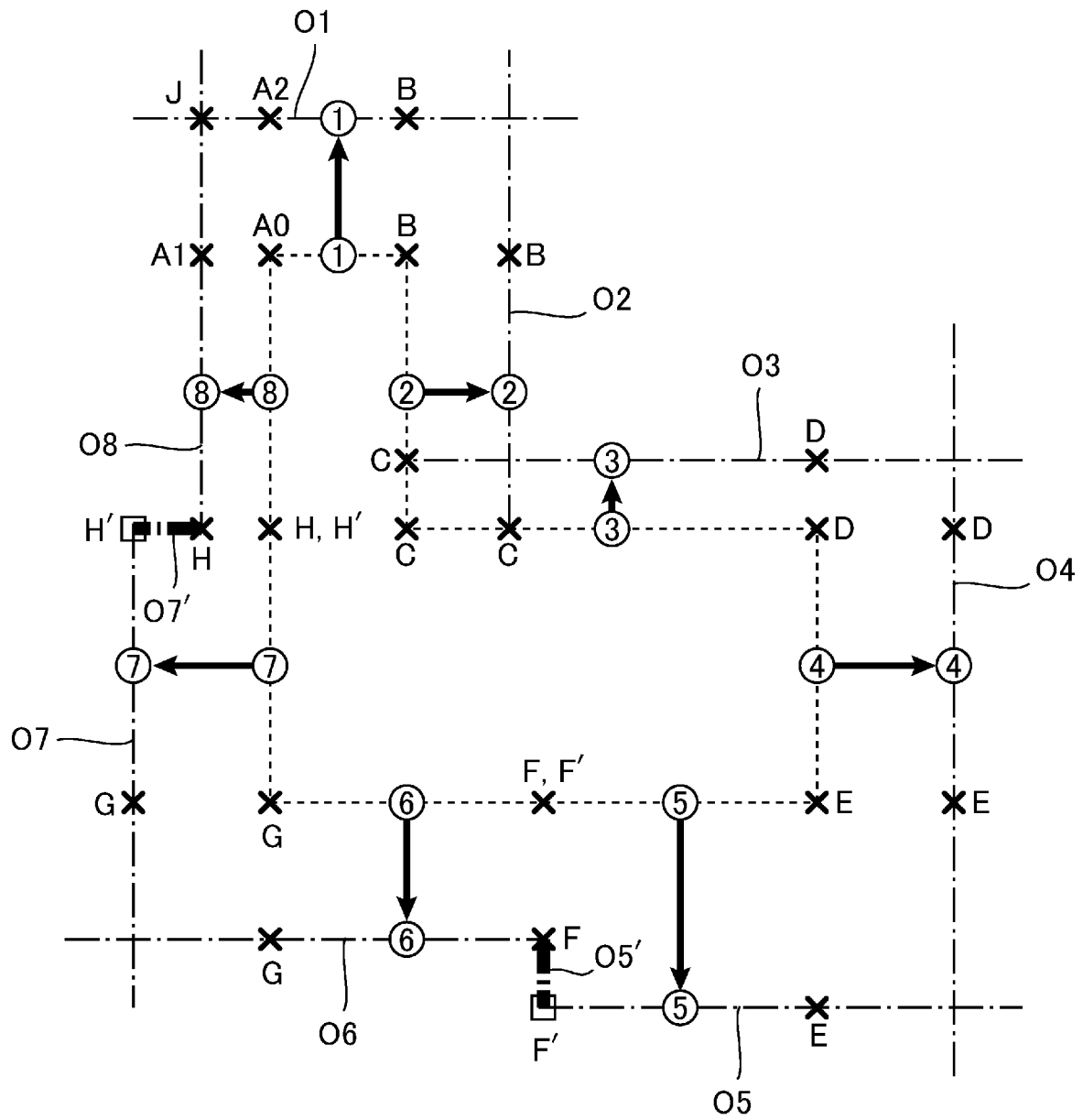
[図55]

(A, 1), (B, 2), (C, 3), (D, 4), (E, 5),
 (F', Φ), (F, 6), (G, 7), (H', Φ), (H, 8)

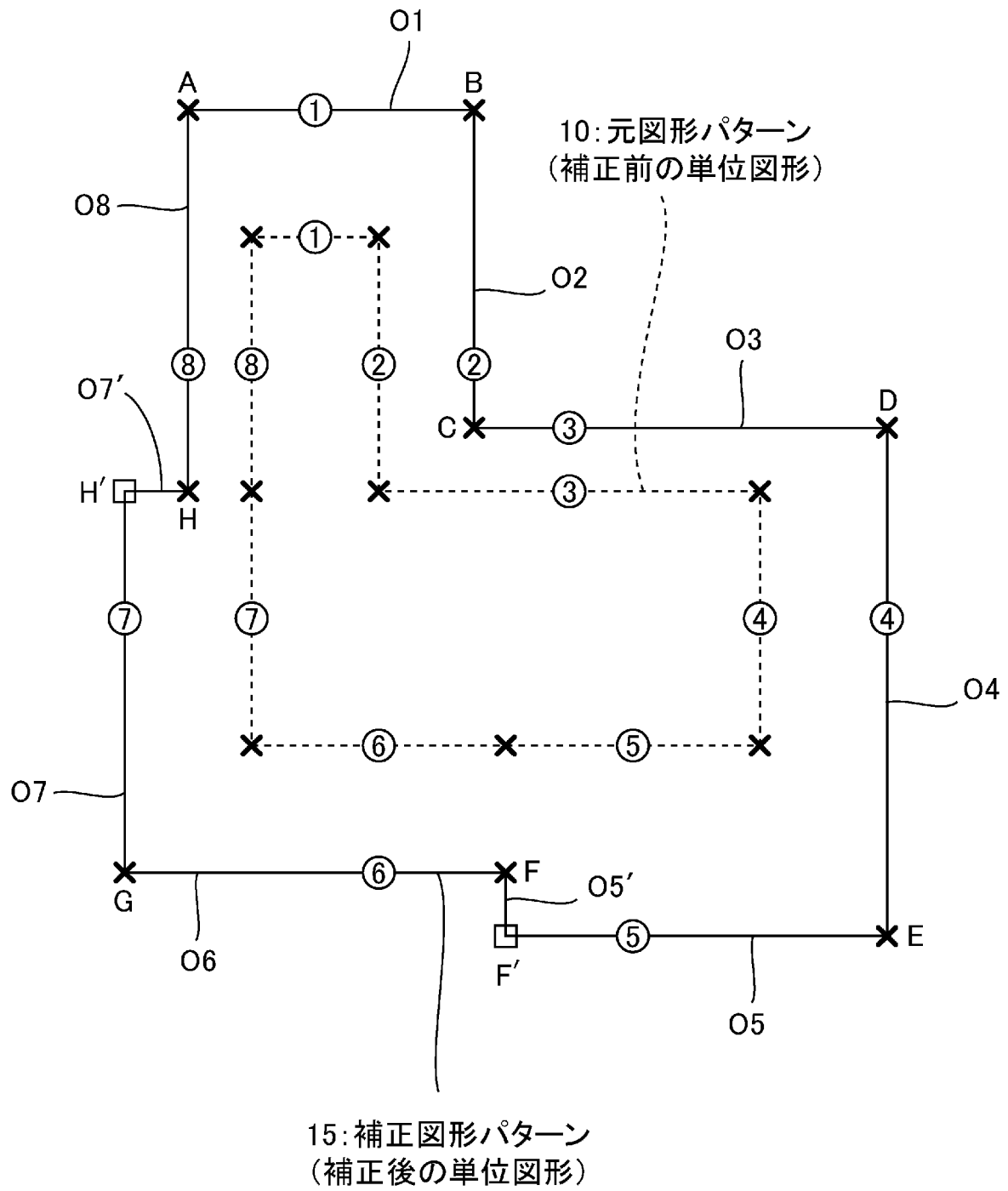
[図56]



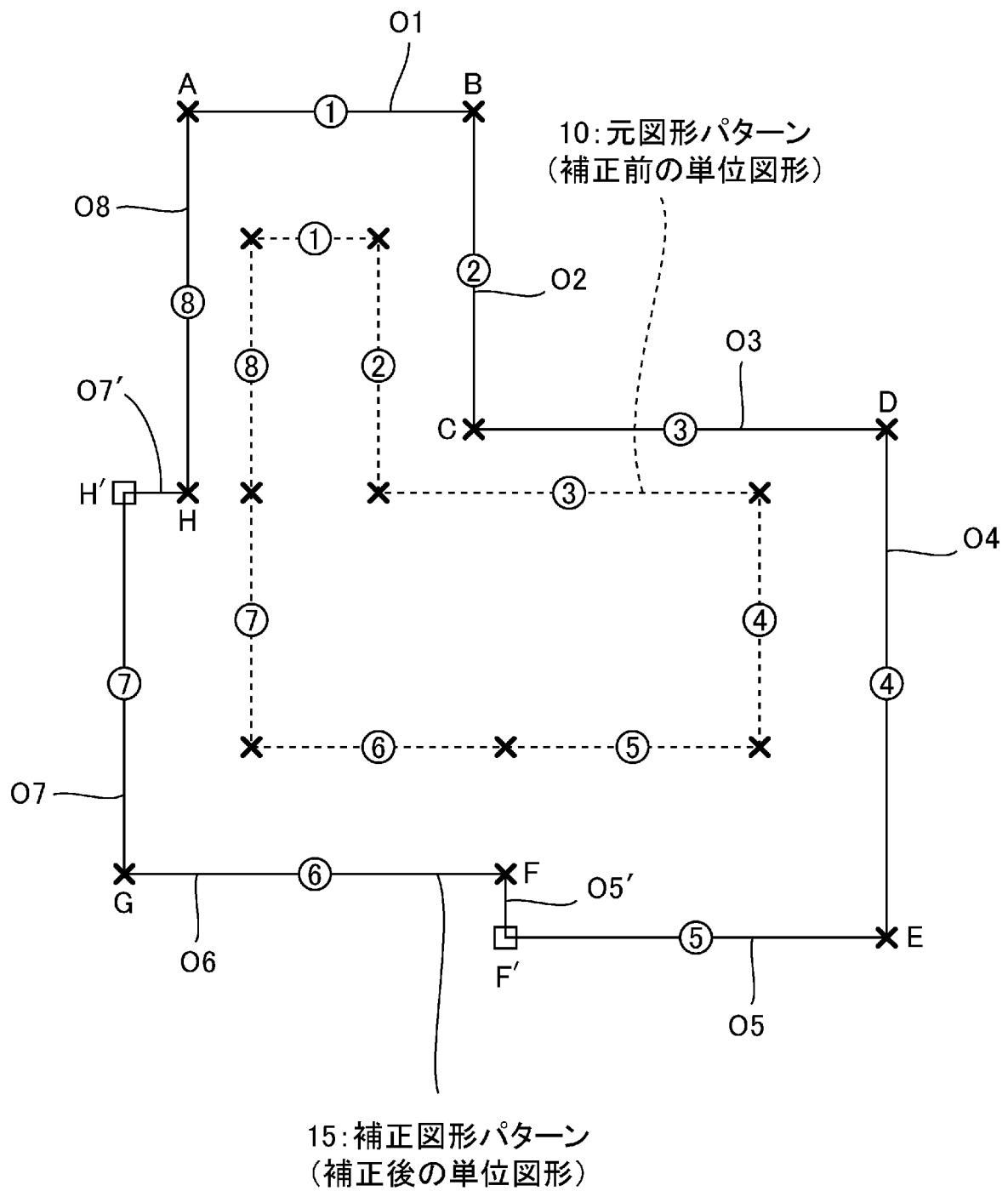
[図57]



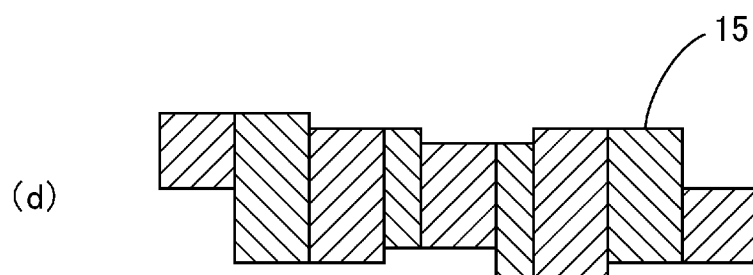
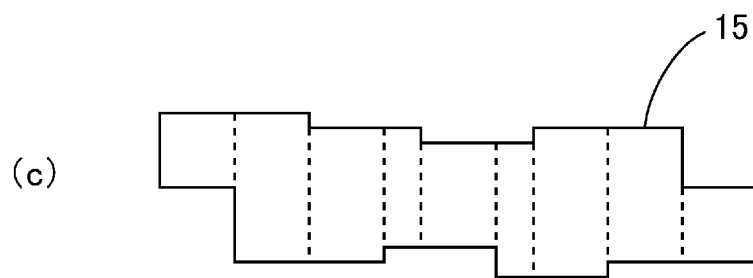
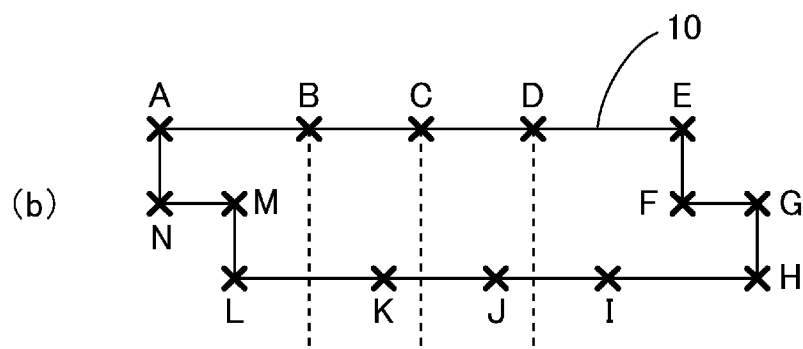
[図58]



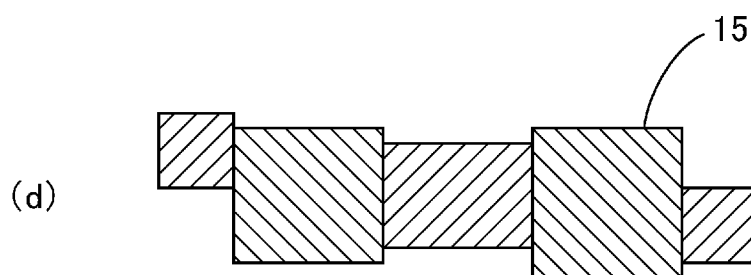
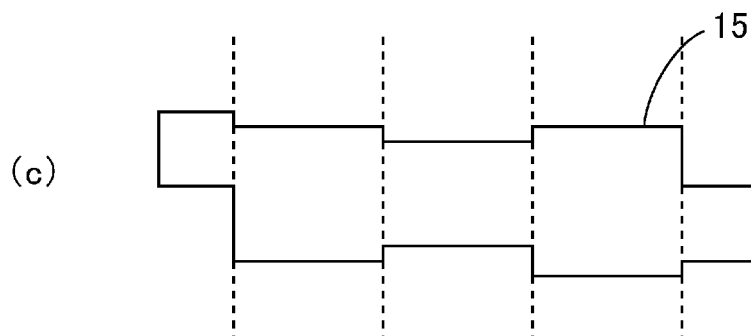
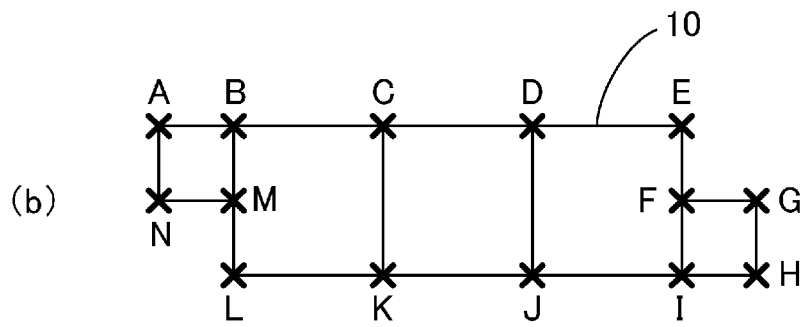
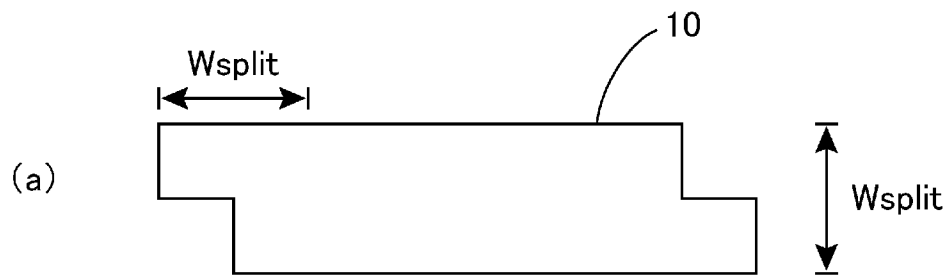
[図59]



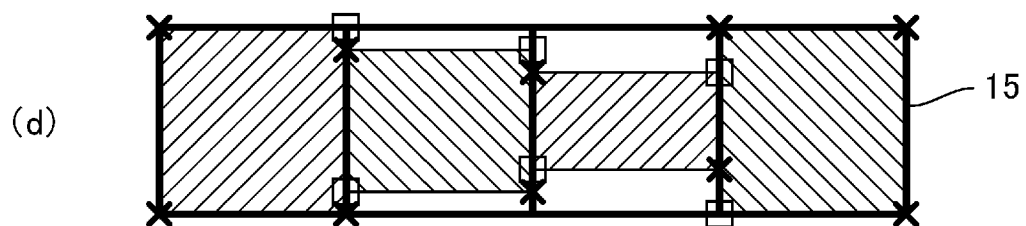
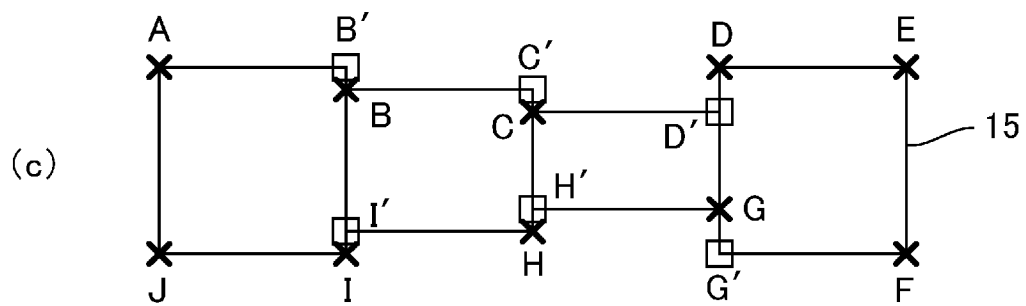
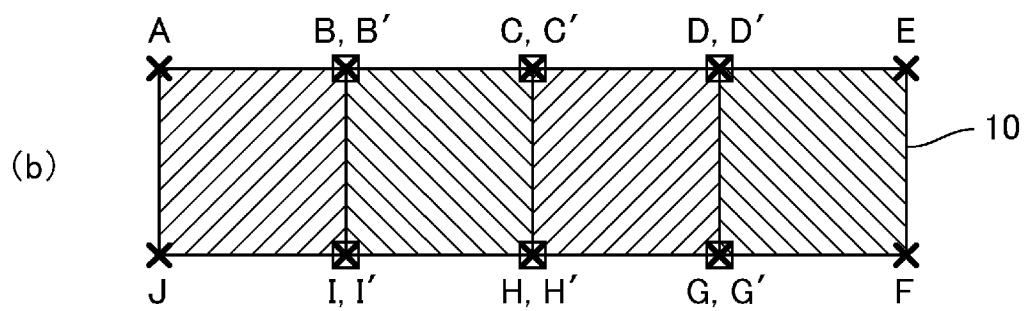
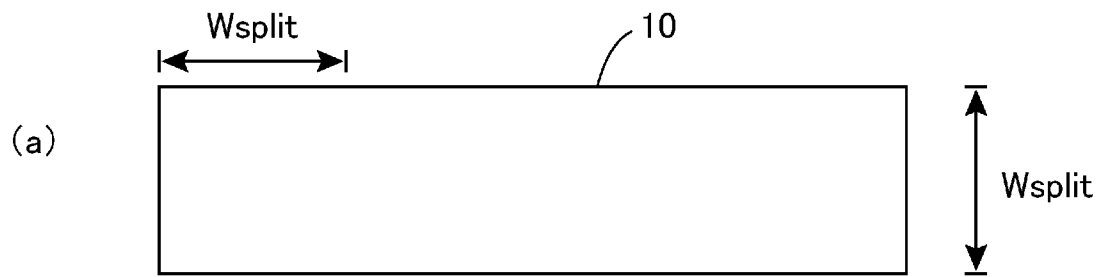
[図60]



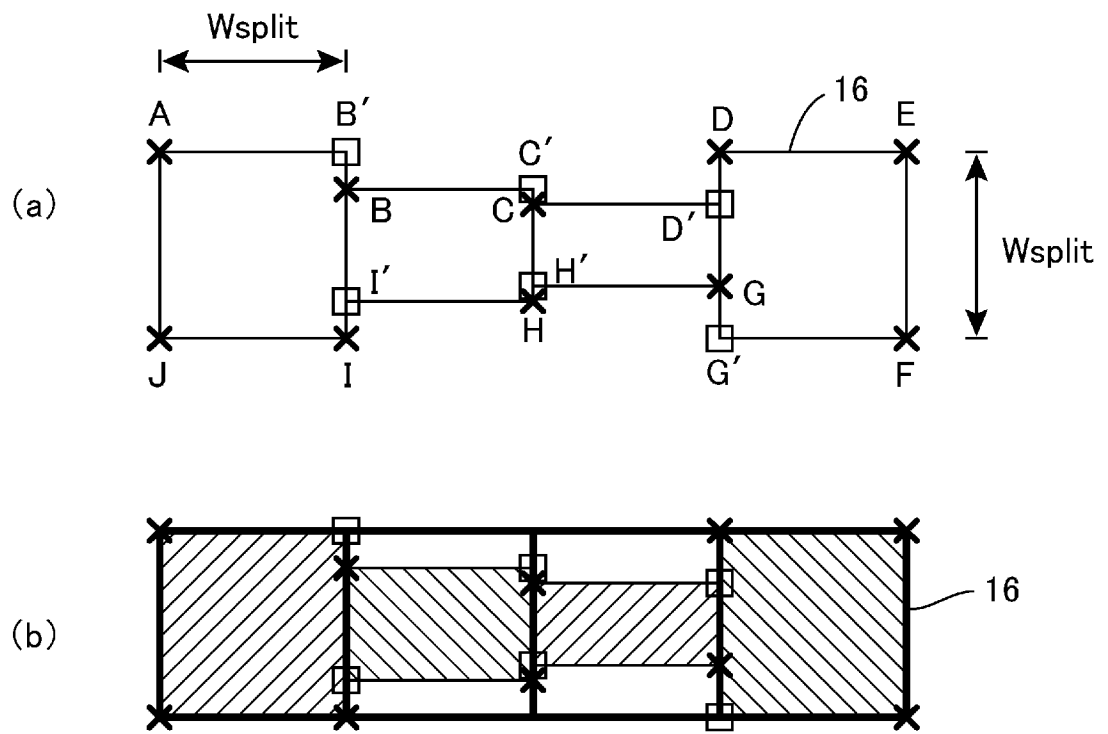
[図61]



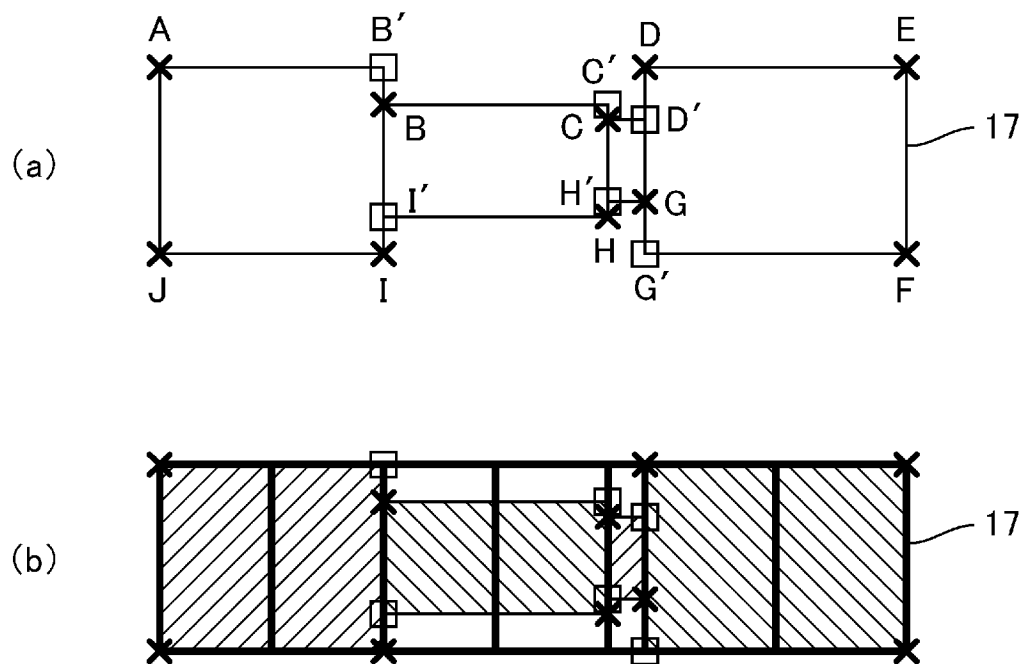
[図62]



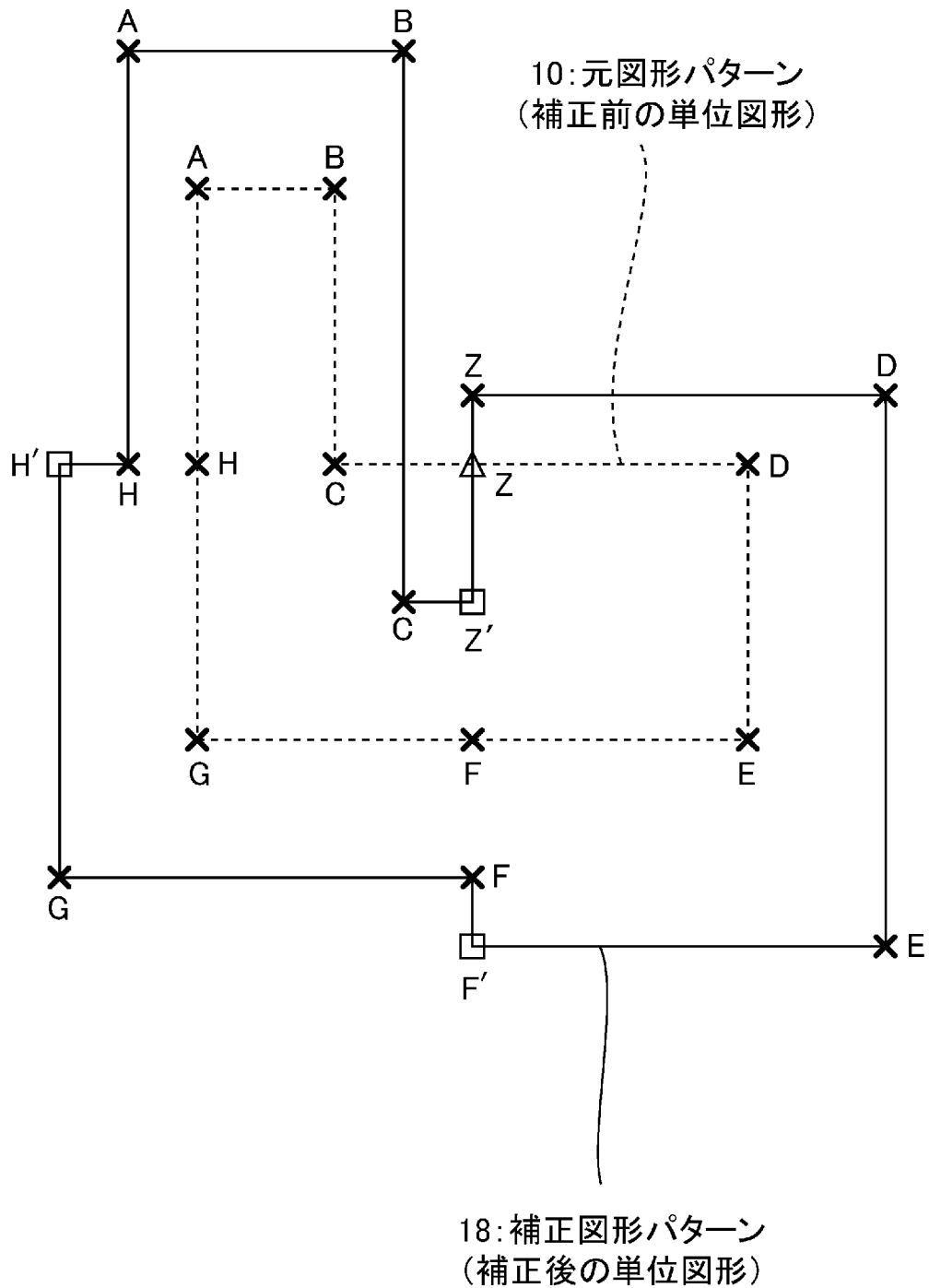
[図63]



[図64]



[図65]



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2018/022296

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl. G03F1/36 (2012.01) i, G03F7/20 (2006.01) i, G06F17/50 (2006.01) i,
H01L21/027 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl. G03F1/36, G03F7/20, G06F17/50, H01L21/027

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Published examined utility model applications of Japan	1922-1996
Published unexamined utility model applications of Japan	1971-2018
Registered utility model specifications of Japan	1996-2018
Published registered utility model applications of Japan	1994-2018

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2016-134567 A (NUFLARE TECHNOLOGY, INC.) 25 July 2016, claim 1, paragraphs [0015]-[0118], all drawings & US 2016/0211118 A1 & KR 10-2016-0090250 A & TW 201638997 A	1-21
A	WO 2016/102607 A1 (ASELTA NANOGRAPHICS) 30 June 2016, paragraphs [0026]-[0068], all drawings & JP 2018-500771 A & EP 3037878 A1 & KR 10-2017-0137695 A & CN 107636535 A	1-21

☒ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"I" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

31 July 2018 (31.07.2018)

Date of mailing of the international search report

07 August 2018 (07.08.2018)

Name and mailing address of the ISA/

Japan Patent Office
3-4-3, Kasumigaseki, Chiyoda-ku,
Tokyo 100-8915, Japan

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2018/022296

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2008-65246 A (NEC ELECTRONICS CORPORATION) 21 March 2008, paragraphs [0017]-[0057], all drawings (Family: none)	1-21
A	WO 2003/067331 A1 (SONY CORP.) 14 August 2003, page 6, line 6 to page 9, line 21, fig. 1-4, 10 & JP 4218528 B2 & US 2005/0147893 A1 & TW 200401163 A & CN 1646988 A	1-21
A	JP 11-194480 A (SONY CORP.) 21 July 1999, paragraphs [0015]-[0034], fig. 1-7 & US 6314554 B1	1-21

A. 発明の属する分野の分類（国際特許分類（IPC））

Int.Cl. G03F1/36(2012.01)i, G03F7/20(2006.01)i, G06F17/50(2006.01)i, H01L21/027(2006.01)i

B. 調査を行った分野

調査を行った最小限資料（国際特許分類（IPC））

Int.Cl. G03F1/36, G03F7/20, G06F17/50, H01L21/027

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	1922-1996年
日本国公開実用新案公報	1971-2018年
日本国実用新案登録公報	1996-2018年
日本国登録実用新案公報	1994-2018年

国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
A	JP 2016-134567 A (株式会社ニューフレアテクノロジー)2016.07.25, [請求項1], 段落[0015] - [0118], 全図 & US 2016/0211118 A1 & KR 10-2016-0090250 A & TW 201638997 A	1-21
A	WO 2016/102607 A1 (アセルタ ナノグラフィクス)2016.06.30, 段落[0026] - [0068], 全図 & JP 2018-500771 A & EP 3037878 A1 & KR 10-2017-0137695 A & CN 107636535 A	1-21

☒ C欄の続きにも文献が列挙されている。

☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの

「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの

「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）

「O」口頭による開示、使用、展示等に言及する文献

「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」同一パテントファミリー文献

国際調査を完了した日

31.07.2018

国際調査報告の発送日

07.08.2018

国際調査機関の名称及びあて先

日本国特許庁（ISA/J P）

郵便番号100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官（権限のある職員）

植木 隆和

2G

3706

電話番号 03-3581-1101 内線 3226

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
A	JP 2008-65246 A (NECエレクトロニクス株式会社)2008.03.21, 段落[0017] - [0057], 全図 (ファミリーなし)	1-21
A	WO 2003/067331 A1 (ソニー株式会社)2003.08.14, 第6頁第6行-第9頁第21行, 図1-4, 10 & JP 4218528 B2 & US 2005/0147893 A1 & TW 200401163 A & CN 1646988 A	1-21
A	JP 11-194480 A (ソニー株式会社)1999.07.21, 段落[0015] - [0034], 図1-7 & US 6314554 B1	1-21