

単結晶シリコン製造装置における運動モデルを用いた晶癖線の検出

1235055 上田 史織 【画像情報工学研究室】

Detection of habit lines in silicon manufacturing equipment using motion model

1235055 Shiori UETA 【Image Processing and Informatics Lab.】

1 はじめに

家電製品やコンピュータ、太陽電池などの製品には半導体を使用されている。半導体の材料で最も多く用いられているのは単結晶シリコンである。単結晶シリコンの製造では、シリコンが無転位なことが重要でその時表面に晶癖線と呼ばれる細い線が現れる。転位は熱衝撃などの環境に敏感であり転位が生じた時、晶癖線は消えて多結晶シリコンになる。製造現場では多結晶シリコンになった場合、そのシリコンの製造を中止している。

転位を回避するために、W.C.Dashらによりネックプロセスを導入した製造方法が提案された[1]。これはシリコンを細く成長させ、晶癖線確認後に必要な大きさまで拡大する製造方法である。そのため、ネックプロセスで晶癖線を発見することが重要となる。

晶癖線の出現は不規則であり、製造現場では晶癖線を定期的に確認している。現在人が監視した時の単結晶シリコンの製造成功率は約66%であるため、電気代などのコストが非常にかかる。

そこで我々はネックプロセスで晶癖線を自動検出することを目的とし、異方性を持つDifference of Gaussian画像で晶癖線検出を行った。しかし、従来の手法は製造における事前知識を十分に活用してと言えなかった。本研究では、シリコンの回転速度と90度ずつ出現するという事前知識を導入した粒子フィルタの実装を行い、従来の手法との比較を行う。

2 提案手法

2.1 Difference of Gaussianを用いた晶癖線の検出

晶癖線を検出するために画像の晶癖線とメニスカスの形状に注目した。図1から晶癖線はメニスカスより横幅が狭いことが分かる。この特徴を活用するために、横方向のみぼかしの強度を変えた画像の差分より晶癖線の検出を試みた。本手法は式(1)からガウシアンフィルタを作成することで、横軸の標準偏差 σ_x と縦軸の標準偏差 σ_y を独立して用いることが可能となった。

$$f(x, y) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left(-\frac{x^2}{2\sigma_x^2}\right) \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{y^2}{2\sigma_y^2}\right) \quad (1)$$

次にラベリング処理を行い晶癖線とそうではない画素を区別する。しかし、図2のようにメニスカス付近

の画素が分離して現れることがある。そこで膨張処理を行い、左右のメニスカス各々を一つの画素の集合として扱えるようにする。膨張処理後はラベリング処理で画素の集合を区別する。図3から晶癖線はメニスカスより画素の総数が少ないことがわかる。この特徴を利用し、3番目に画素の総数が多い画素の集合を晶癖線として検出した。

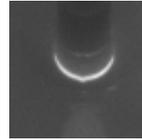


図1: 入力画像



図2: 二値画像



図3: 膨張画像

2.2 位置推定に必要な円情報の取得

粒子フィルタの状態 \mathbf{x}_t は、円の中心から見た角度情報を保持している。そのため、粒子の位置を把握するには円の中心座標と半径が必要で、ハフ変換から中心座標と半径を求める。今回は処理時間の都合により円のハフ変換を行った。

まず、画像より高周波成分を除去し、二値化を行う。晶癖線は外側に出現するので外円を検出するために、作成した二値画像の白画素より上の画素を全て白に変更する。その後、作成した画像からハフ変換で円の中心と半径を求める。

2.3 粒子フィルタを用いた晶癖線の位置推定

粒子フィルタは状態空間モデルの状態推定アルゴリズムであり、以下の状態空間モデルからなる。

$$\mathbf{x}_{t+1} = F(\mathbf{x}_t, \xi_t) \quad (2)$$

$$\mathbf{y}_t = H(\mathbf{x}_t, w_t) \quad (3)$$

式2は状態 \mathbf{x}_t の時間変化を関数 F で表しており、式3は \mathbf{x}_t から観測値 \mathbf{y}_t を求める関数 H で表している。 ξ_t はシステムノイズで、 w_t は各粒子の重みである。粒子フィルタは式(2)に基づき状態の更新を行い、その後式(3)に基づき尤度の計算をし、その結果を元に粒子をリサンプリングする。これを1フレームに対して繰り返す。これを1フレームに対して繰り返す。これを1フレームに対して繰り返す。これを1フレームに対して繰り返す。これを1フレームに対して繰り返す。

本手法では、粒子フィルタの状態 \mathbf{x}_t には前節で述べたように円の中心から見た角度のみとして定義するこ

とにした。つまり、円の中心から見た角度を θ とすると状態 x_t は式 (4) の通りである。

$$x_t = \begin{bmatrix} \theta_t \end{bmatrix} \quad (4)$$

これより、等速円運動による回転は定数加算となり、回転速度の変動を雑音として記述することができる。したがって状態方程式は、以下の通りである。

$$x_{t+1} = x_t + v_r + \xi_t \quad (5)$$

v_r は 1 フレーム間隔の晶癖線移動角度であり、 ξ_t は時刻 t の雑音である。尤度計算は異方性 DoG により検出された箇所を中心とした正規分布に従って分布するモデルを定義することにした。予備実験より尤度計算に使用した正規分布の標準偏差 $\sigma = 20$ とした。また製造の事前知識として晶癖線が 90 度ずつ出現することを考慮し、尤度計算は 90 度ずつを同様に扱った。

3 検証

粒子フィルタの有効性を確認するために、従来の手法との比較を行う。

従来手法は各フレームより前の 1 分間の DoG による検出結果の移動平均値を結果として使用する。粒子フィルタによる手法はまず 360 度の中で晶癖線尤度が最大ものをフレームごとに抽出する。雑音による結果の変動も考慮してその結果を 5 回取得し平均値を結果として使用する。また実際のシステムでは同じ晶癖線を検出しないことを考えているため、DoG による検出後 7 フレームは検出処理を行わないように設定した。

図 4 はあるフレームの各角度の尤度を算出した結果である。この結果より、尤度全体から見たらピークは低いですがピーク付近には近い値の尤度が収束していることが分かる。この特徴を利用して本手法では尤度を算出する角度間隔を 10 度に設定し、晶癖線最大尤度を求めた。また、晶癖線最大尤度は前後フレームで値のバラつきがあるため、結果には最大尤度とその尤度を 5 フレームごとに平均を取ったグラフの二つを重ねて描画する。

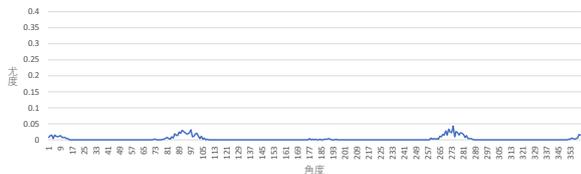


図 4: あるフレームの各角度の尤度の合計値

4 結果

今回は 3 つの画像データを使用して検証を行った。その中の二つのデータについての結果を記述する。図 5 と図 6 は同様の画像データに対する従来手法と本手法の結果である。このデータは 1 分が 214 フレームであり、従来手法は 214 フレームから計測を開始している。またどちらのグラフにもある赤い線はシリコンの引き上げ時刻を表している。グラフより移動平均の結果は 1 回目と 2 回目のピークで差があるのに対して、本手法では 2 つのピークではあまり差が見られなかった。



図 5: DoG 画像による移動平均の結果 (1)



図 6: 晶癖線最大尤度の平均値の結果 (1)

図 7 と図 8 は先ほどの二つとは異なる画像データに対する従来手法と本手法の結果である。このデータも 214 フレームから計測を開始している。従来手法は最後、検出頻度が上昇しているのに対して、晶癖線最大尤度は上昇していない。この理由については、図 9 のようにピークが 4 つある状態が最後まで続いていたため、1 箇所に収束しなかったからである。

以上から、晶癖線検出に対して本手法より従来手法の精度が良いことが明らかになった。しかし、移動平均は前のフレームの結果を参考に行っているため、検出頻度の上昇はリアルタイムで処理を行う粒子フィルタより遅い。



図 7: DoG 画像による移動平均の結果 (2)

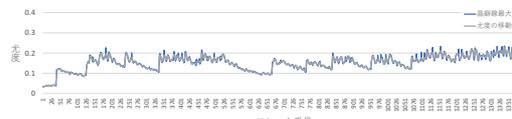


図 8: 晶癖線最大尤度の平均値の結果 (2)



図 9: 図 8 の結果のあるフレームの各角度の尤度

5 まとめ

本研究では従来の手法で活用していなかった事前知識を導入した粒子フィルタを実装し、DoG の検出結果の移動平均と比較した。その結果、本手法より従来手法の精度が良いことが確認できた。しかし、従来手法は前のフレームを参考に行っているため、リアルタイムでの実行には本手法の方が強いことが確認できた。

参考文献

[1] William.C.DASH : Growth of Silicon Crystals Free from Dislocations, Journal of Applied Physics, Vol.30, No.4, pp. 459-474, 1959.