

光タンパク質の開発によって、生きた細胞で目的の分子を見ることが簡単にできるようになった。遺伝子組換え技術により観察したい生物に蛍光タンパク質を導入すれば、生きたまま観察したい分子だけを光らせて観察できる。蛍光顕微鏡技術を利用し、更に発展させることが、生物の優れた能力のからくりを理解していくうえで必要である。

2 分解能の壁を越える顕微鏡技術

二つの近接した点を二つと見分けられる最小の距離のことを分解能という。顕微鏡の分解能には制限がある。顕微鏡では倍率も重要であるが、レンズを組み合わせれば無限に倍率を上げることができる。しかし、ある倍率以上になると2点を見分けることができなくなる。これが理論的な光の分解能の限界である。光を使用する光学顕微鏡の場合、分解能は光の波が重なり合う回折という現象のため、光の波長の約半分の大きさが限界となる。

この分解能の限界を打ち破ったのが、超解像顕微鏡の開発である。蛍光顕微鏡の分解能の限界は約250 nmだが、超解像顕微鏡法を用いれば15～100 nmという高分解能が得られる。この技術を開発した研究者は、2014年のノーベル化学賞を受賞した[1]-[3]。

2.1 超解像顕微鏡

超解像顕微鏡は、光の物理法則を変えたのではない。光の物理法則の制約の中で、分解能を制約する原因を回避する戦略を用いたのである。例えば、構造化照明法は、サンプル中の顕微鏡の分解能よりも微細な模様、励起光で作った別の模様を重ね合わせることで干渉による模様を作り出す[4] (図1)。この干渉模様はモアレと呼ばれ、元の模様よりも大きな模様になる。そのため、通常の顕微鏡の分解能でも、モアレを観察できる。この観察したモアレと、重ね合わせた既知の模様から、サンプル中の微細な模様を計算によって求め

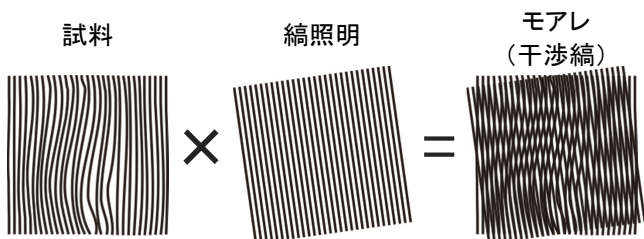


図1 構造化照明法の原理

試料の模様が微細であれば、通常の顕微鏡では観察できない。しかし、既知の模様(縞)を持つ照明を重ね合わせると、模様同士が干渉し合いモアレ(干渉縞)が発生する。モアレは元の模様より大きいので、通常の顕微鏡で観察できる。観察したモアレと既知の縞照明の情報を基に、試料の姿を計算により解くことができる。

ることができる。このように、直接見ている画像は通常の分解能の制約に縛られているが、コンピュータ上での計算によって初めて分解能の壁を越える画像が得られるのである。

2.2 分解能低下を防ぐ技術

超解像顕微鏡法は原理実証されたが、生きた生物を用いて実際に計測を行うのは極めて困難である。超解像顕微鏡は、極限の分解能を追求するために、理想的な条件からの僅かな逸脱によっても理論が破綻し、分解能が大幅に低下したり、アーティファクト(画像処理過程で発生するデータの誤りや信号のゆがみにより本来存在しないような構造が生じること)が生じたりしてしまう。我々は超解像顕微鏡法を実際の生物に適用するうえで必要となる技術の開発に取り組んできた。例えば、立体的な細胞では非焦点面の蛍光ノイズが強くなり、そのために分解能が大きく低下する。このような非焦点面の蛍光を除くデコンボリューションや蛍光ノイズを低減させるデノイジングなどの画像処理を超解像顕微鏡法に適用し、蛍光ノイズの存在下でも高分解能を得ることに成功した[5]-[7]。また、従来は、超解像顕微鏡法により得られた画像に生じるアーティファクトの原因の特定は困難であった。この問題に対し、我々は、得られた画像を空間周波数領域で解析することにより、光軸のズレや迷光のような光学調整の問題を簡単に特定できる解決法を提案してきた[8][9]。

しかし、現在の蛍光顕微鏡法における避けがたい問題が、生体試料自体の持つ屈折率の問題である。細胞自体がビー玉のような屈折率物体として働くため、観察対象によっては細胞という“レンズ”を通して観察することになり、観察が困難・不可能になる(図2 A)。例えば、動物の初期胚を従来の顕微鏡で観察すると、対物レンズに近い部分はよく見えるが、その反対側の深部の領域の像劣化は激しく、ほとんど見ることができない。生物試料の屈折率は極めて複雑に変化しているため、現在の顕微鏡では、組織深部の幹細胞や神経シナプスなどを高分解能で観察することは極めて困難である。

生体試料が持つ屈折率の問題を解決するためには、屈折率をリアルタイムで測定し補正できる補償光学(Adaptive Optics: AO)の開発が不可欠である。AOは、天体望遠鏡に使用されてきた技術であり、地表の大気のゆらぎを光学的に取り除き、地表の天体望遠鏡からでも宇宙の姿を高分解能で観察できる技術である。AOは日本ではNICTで研究されてきた技術であり[10]、衛星との宇宙光通信にも使用されている[11][12]。AOでは、まず波面センサーにより光の波面のゆがみを計測する(図2 B)。次に、形を自在に変