

第 31 回 3D プリンター勉強会

ワークショップ: 「3Dプリンターで ランプシェードをつくる」

金田 泰 (Dasyn.com)

新技術: 光り輝く透明軽量 3D 印刷

▶ 従来の 3D 印刷の解決すべき課題

- 透明なフィラメントを十分にいかせなかった.
- 印刷に時間とコストがかかる.

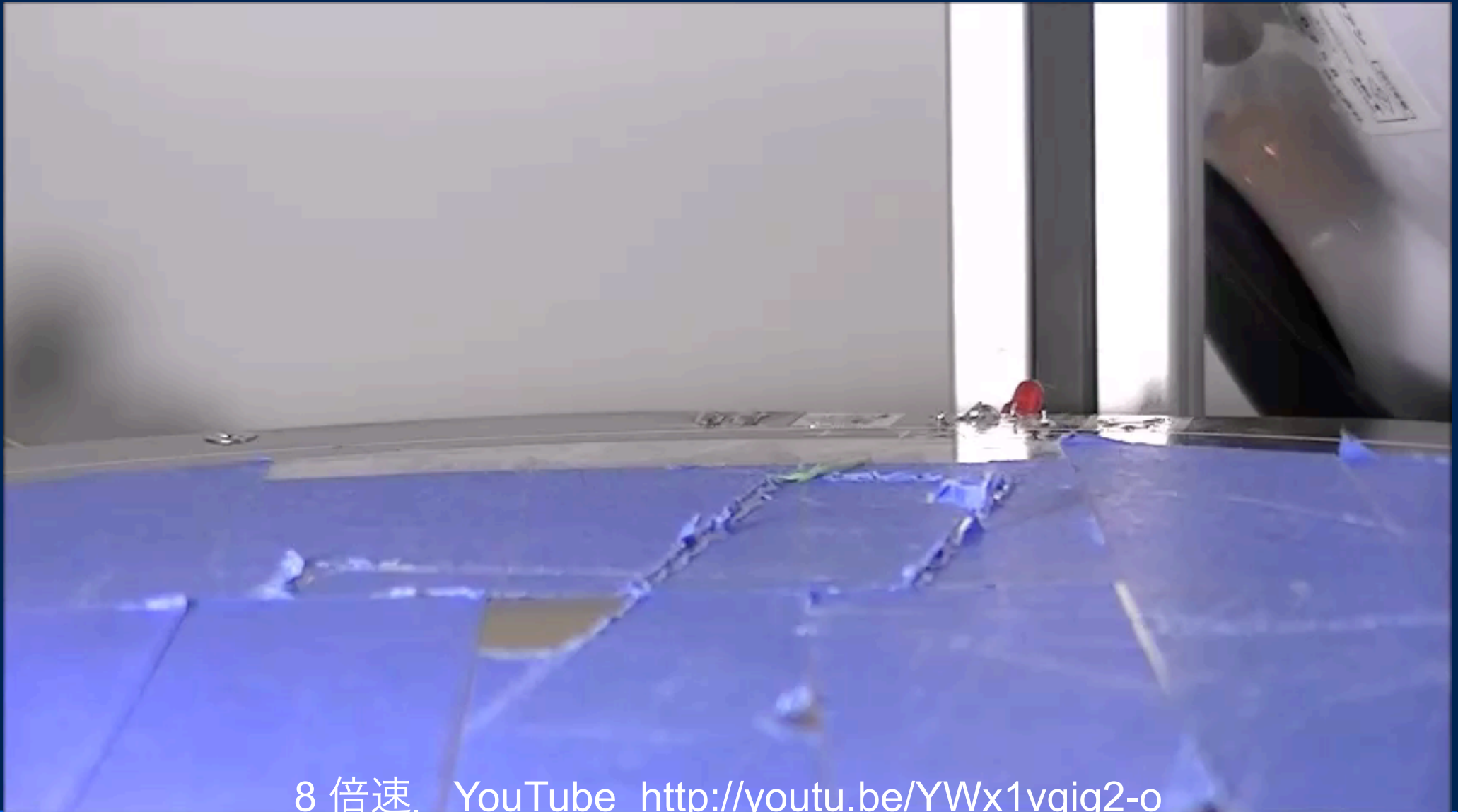


▶ 解決策の案

- フィラメント 1 層だけで高強度が実現できれば, 透明さをいかし, 印刷時間を大幅に短縮することができる.
- サポートなどで印刷物の表面をよごさないようにできれば, 透明さをいかせて, コスト増につながる後処理も不要になる.

問題解決のための新技術

- ▶ヘリカルに印刷することで「層」のつぎめもサポートもなしにきれいな印刷を実現し強度もたかめることができる。



8 倍速. YouTube <http://youtu.be/YWx1vqig2-o>

問題解決のための新技術 (つづき)

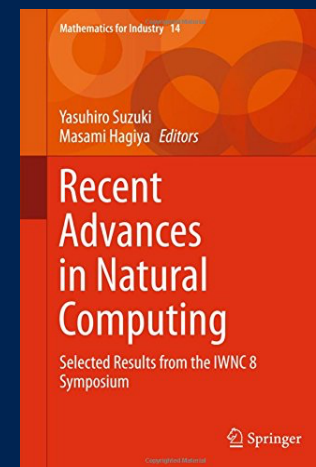
▶ 透明な PLA フィラメントをつかえば、1 重であることを長所にかえて光輝く美的な印刷が実現できる。

▶ ヘリカルな印刷には従来の 3D CAD がつかえないので、かわりに Python 言語のライブラリを開発し提案している。

- この方法では設計者がプログラミングのスキルをもつ必要がある。

▶ アイデアの権利化

- 日米において特許出願 (各 5 件)。
- 論文 (ジャーナル 2, 書籍 1, 他)。



新技術の照明器具への応用の可能性

- ▶ この方法による造形物は透明で光り輝くので、LED の光をあてればその魅力をひきだせる。



▶ LED シェードにしたときの特徴

- シェードの表面に模様や絵・文字がえがける。
- 従来のシェードにはない特性や質感が実現できる。
- 30 分程度で造形できるので、超ラピッド・プロトタイピングが実現できる。

ワークショップ: 新技術によるランプシェード製作

▶ 手順

- Python のプログラムをつかって、直径 6 cm のランプシェード印刷の G コード (印刷データ) を生成する。

```
x0 = 0; y0 = 0
obj.helix(radius, height, vpitch, x0, y0, 0, minlen=0.2)

if mode != 0:
    obj.modulate_cylinder(
        lambda theta, z: modulate(mode, Steps, theta * 180 / PI, z - 90))

obj.deform_cylinder(
    lambda r, theta, z: (r, theta, z + (height - z) * z * cos(3 * theta) * 0.0015),
    lambda v, r, theta, z: v,
    lambda c, r, theta, z: c)
z0 = radius * sin(0.5*PI*z1/height1) + 0.1

obj.deform_cylinder(
    lambda r, theta, z: (r if r > radius1 else r * cos(0.5*PI*(z-z1)/height1) * fr(theta, z),
        theta,
        z if r > radius1 else r * sin(0.5*PI*(z-z1)/height1) + z0),
    lambda v, r, theta, z: v,
    lambda c, r, theta, z: c if z > 0.05 * radius else 1.5 * c)
obj.deform_cylinder(
    lambda r, theta, z: (r, theta, z),
    lambda v, r, theta, z: (vfacout * (r/radius)**2 + vfacin * (1 - (r/radius)**2)) * v,
    lambda c, r, theta, z: (cfacout * (r/radius) + cfacin * (1 - (r/radius))) * c)

obj.draw(minMotion = 0.4)
```

- きょうは時間がないので、記述済みのプログラムにパラメタをあたえて実行し、G コードを生成する。
- 生成した G コードを使用してランプシェードを 3D 印刷する。
 - きょうは Delta 型 3D プリンタを使用するが、他の種類のプリンタでも印刷可能。
- ランプシェードをあらかじめ用意した USB LED 光源とくみあわせて照明器具をつくる。
- つくった照明器具で照明をテストする。



手順 1: G コードの生成

- ▶ 本来はつぎのように Python のプログラムにパラメタをあたえて G コード・ファイルを生成する.
`python shadesWS.py パラメタ ... >ファイル.gcode`
- ▶ わかりやすくするため, Web インタフェースを用意した.



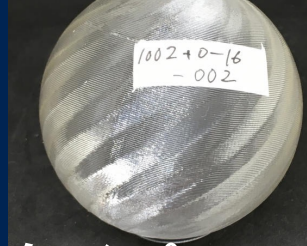
```
; Shade for the workshop 2016-12-3  
; shade2007-0-12-3-0.001.gcode  
G92 E0  
G90  
G21  
M105  
M140 S32  
M141 S30.0  
M142 S0.0  
M113 S1.0  
M108 S16.0
```

生成された G-code

手順 1: G コードの生成 (つづき)

▶ 1.1 シェード・タイプの選択

タイプ 1002



タイプ 1004



タイプ 1005



タイプ 1006



タイプ 2002



タイプ 2004



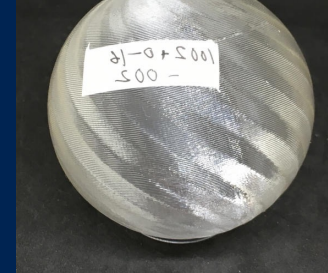
タイプ 2005



タイプ 2006



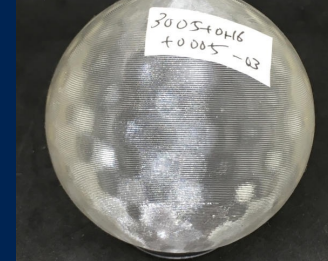
タイプ 1003



タイプ 2003



タイプ 3005



タイプ 1007



タイプ 2007



手順 1: G コードの生成 (つづき)

▶ 1.2 反復回数の選択

4 回



...

16 回



...

96 回



8 回



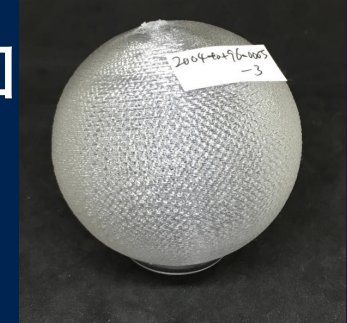
...

24 回



...

96 回



16 回



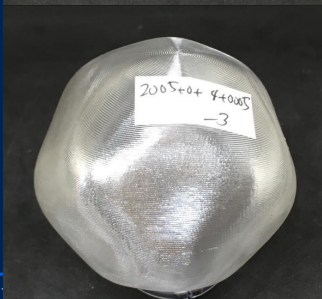
...

64 回



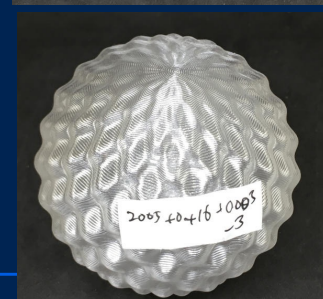
実演

4 回



...

16 回



...

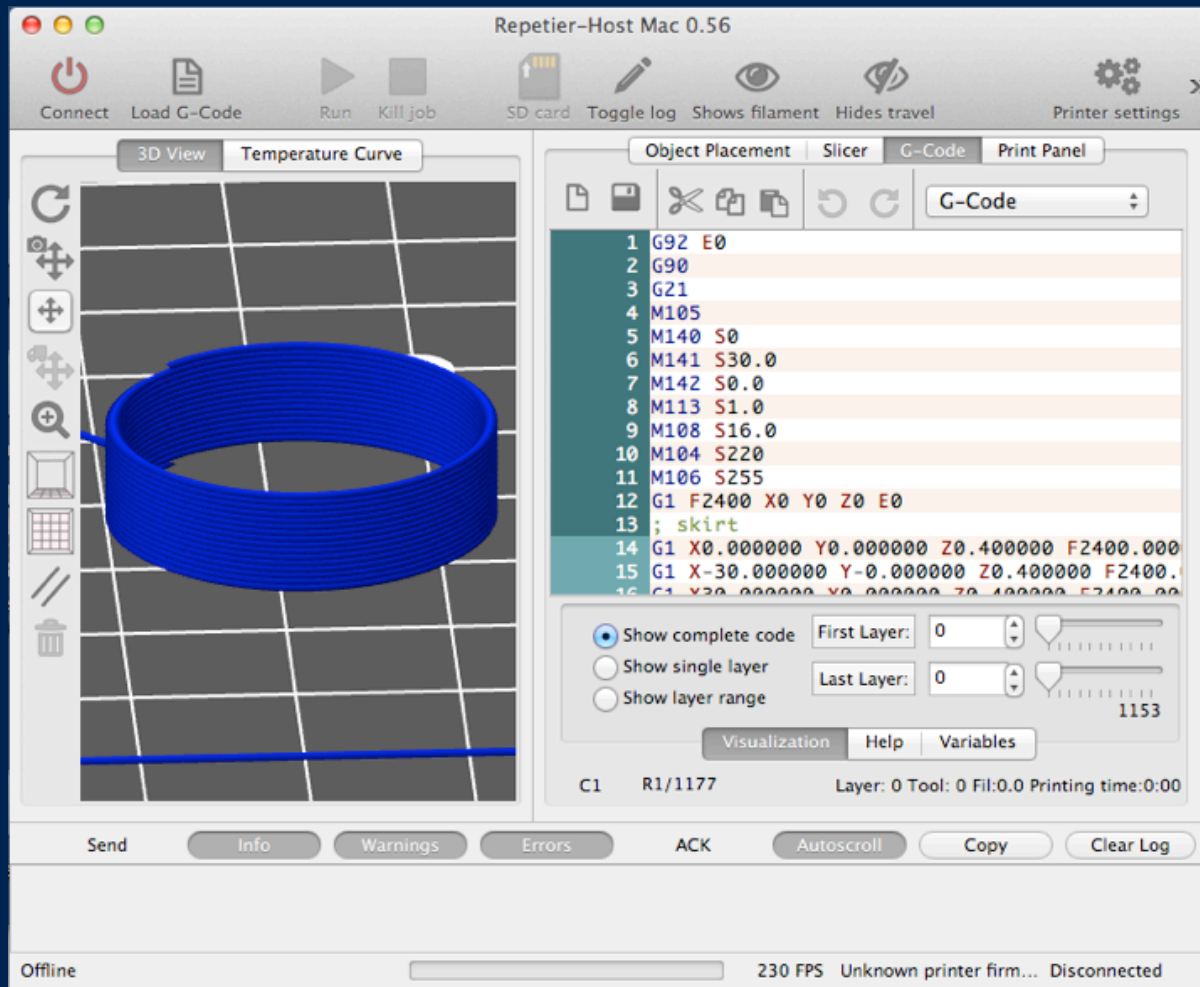
64 回



手順 2: 3D 印刷

▶ PC (またはタブレット) でデルタ型 3D プリンタを制御して印刷する.

■ Repetier Host というソフトウェアを使用する.



手順 3: 照明器具の製作

▶ くみたてずみの USB LED ライトを使用する.

- ただし, シェードをはりつけるための粘着テープは各自でつける.
- ライトはシェードの任意の位置にはりつけられる.



実物

手順 4: 照明器具のテスト

- ▶ AC-USB アダプタや USB ハブを使用して製作した器具をテストする.



実物

