



# 炎舞

雰囲気可視化し、話し合いを活発に



## Concept

会議、グループワーク、人との会話など多くの対話の場で、発言がしにくい事はないだろうか？このような場で活発な対話を目指すには、「話しやすい環境」を作ることが重要であると考えます。例えば「たき火」には、視覚及び聴覚から得られる 1/f ゆらぎによるリラックス効果があるとされ、たき火を囲み揺れる火を見ていると自然と話し始めてしまうなど、たき火には多くの「話しやすい環境」を構築する要素がある。実際に「暖炉があるだけで部屋の印象が改善され、リラックスして話しやすく、会話が促進される雰囲気が形成される」ことが研究で明らかとなっている（松波，羽生，2007）。

本構築物は、「場の雰囲気をリアルタイムにセンシングし、たき火の効果を有するの動きに反映させることで円滑な議論を促進すること」を目的として構築する。

Do you ever find it difficult to speak up in many dialogue settings such as meetings, group work, and conversations with others? In order to achieve a lively dialogue in such settings, we believe it is important to create an "environment conducive to speaking. For example, a bonfire is said to have a relaxing effect due to the visual and auditory 1/f fluctuation. For example, people naturally start talking when they sit around a bonfire and look at the flickering fire. In fact, research has shown that "the mere presence of a fireplace improves the impression of the room, creating an atmosphere that is relaxing and conducive to conversation" (Matsunami, Hanyu, 2007).

The purpose of this structure is to "promote smooth discussions by sensing the atmosphere of the place in real time and reflecting the effect of the fireplace on the movements of the participants" .

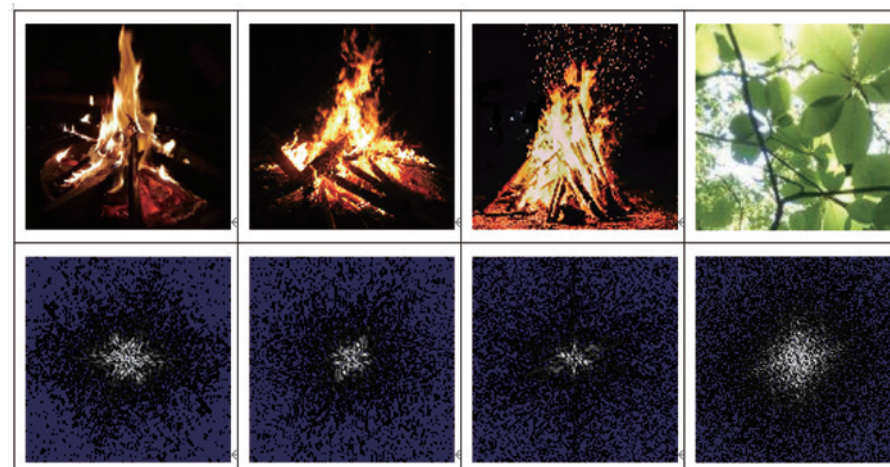
## Fire Analysis



音  
光  
熱

炎とは、熱分解で生じた可燃性ガスの燃焼、パチパチという音は木に含まれる水分が加熱され逃げ場がなくなった際に発生する水蒸気爆発の音である。このたき火の視覚的な動きや聴覚的な音は不規則であり、人間が心地よく感じる 1/f ゆらぎであると言われ、会話のためにたき火や暖炉を用いることが人類に広く普及している。これは、狩猟時代、人類が火によって、夜にたき火を囲みながら会話や宗教・文化の伝達が行われていたことに由来するとされている (Polly W. Wiessner 2014)。

不規則性と 1 / f のゆらぎ

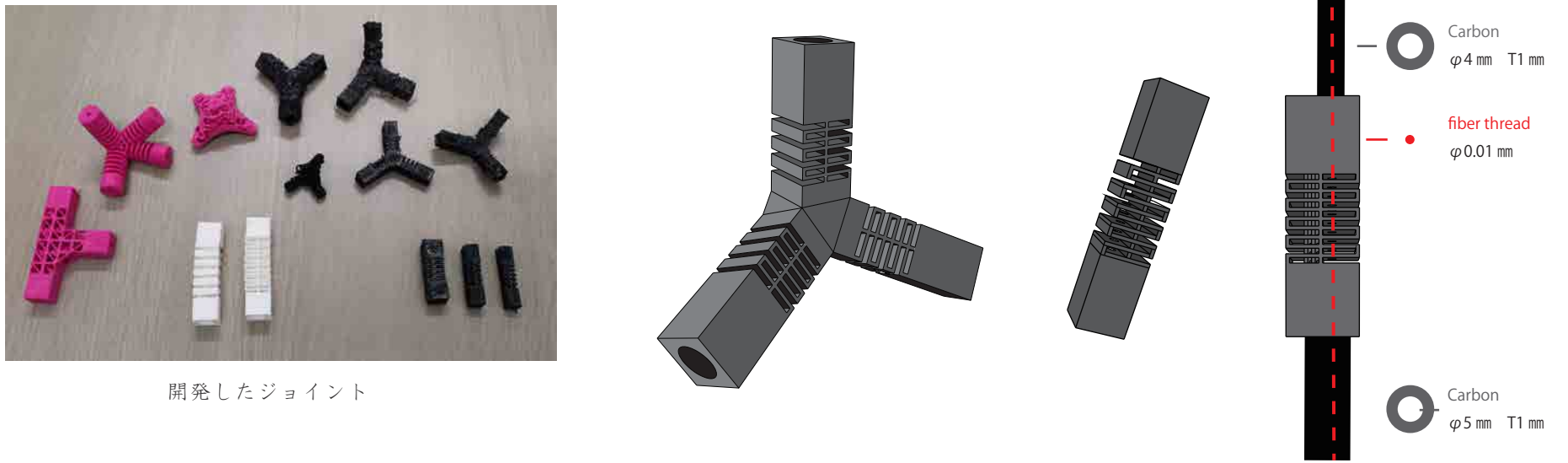


スペクトル解析

The flame is the combustion of combustible gas produced by pyrolysis, and the crackling sound is the steam explosion that occurs when the moisture in the wood is heated and has nowhere to escape. The visual movement and auditory sound of a bonfire are said to be irregular, and are said to be 1/f fluctuations that humans find pleasant, and the use of bonfires and fireplaces for conversation is widespread among humans. This is believed to be due to the fact that during the hunting era, fire was used by humans for conversation and religious and cultural transmission around bonfires at night (Polly W. Wiessner 2014).



# Joint : 3Dprint

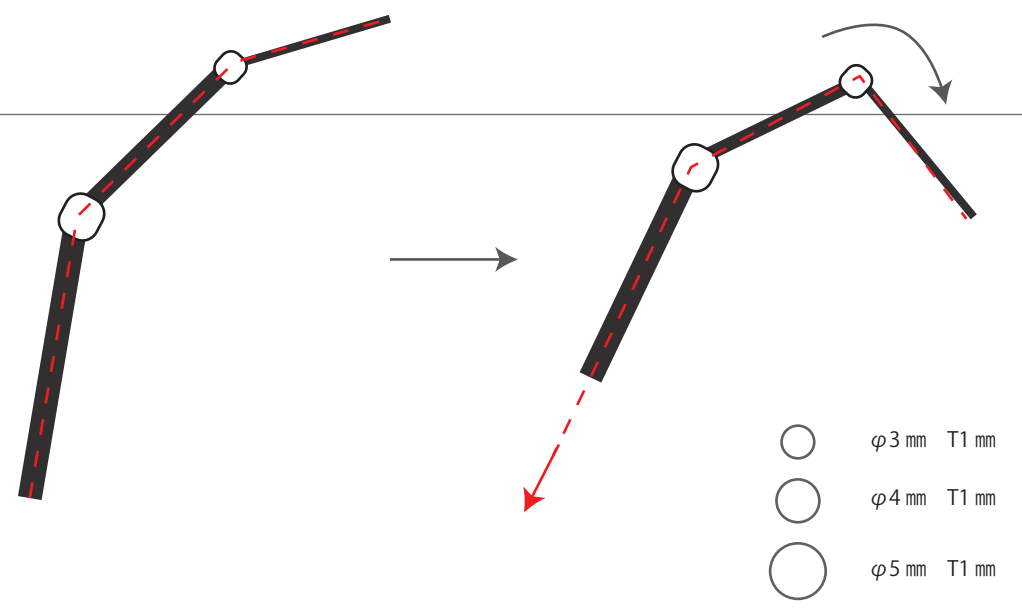


開発したジョイント

ジョイント部では、強度を確保しつつ一方方向にのみ柔らかく変形できる形状を設計した。蛇腹状にスリットが入ることで軸方向の力には変形せず横方向に曲がる変形のみをコントロールできる。  
 素材は柔軟性のあるTPUを使用し、3Dプリントによって造形した。

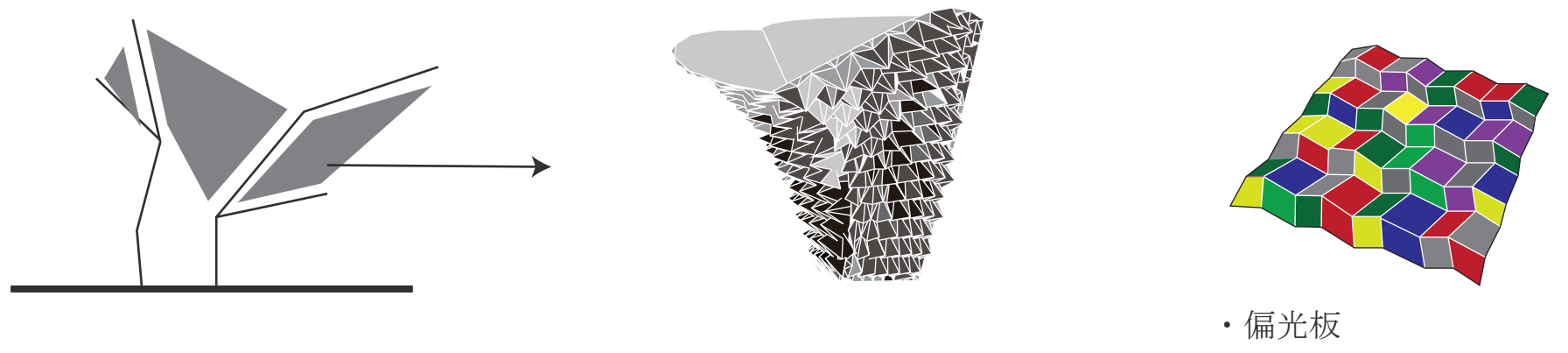
In the joint section, we designed a shape that allows soft deformation in only one direction while ensuring strength. The slit in the shape of a bellows allows only lateral bending deformation to be controlled without being deformed by axial force. The material used is flexible TPU, which was 3D printed.

# Structure Branch



糸を引っ張ると枝が曲がります。規則性がないその動きは、生き物のようでもあります。  
 When pulling the string, the branch bends. The irregularity of its movement is like that of a living creature.

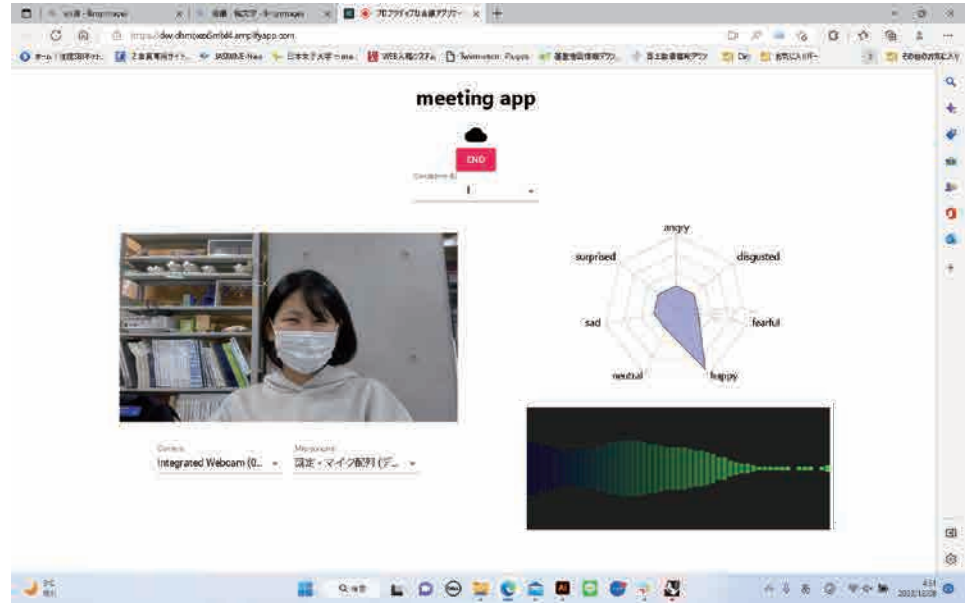
# Spark



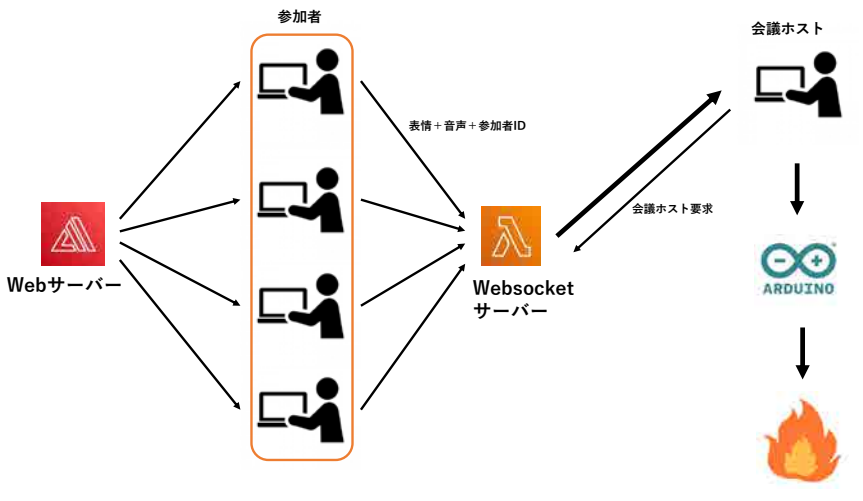
セロファンは光の振動方向により屈折率が異なる性質(複屈性)を持ちます。1枚目の偏光フィルムで光の一方方向の振動が取り出され、それが複屈折により分光した後、2枚目のフィルムにより特定の色の方向のみが取り出されるため、色付いて見えます。

Cellophane has a refractive index that differs depending on the direction of vibration of light (birefringence); the first polarizing film extracts the vibration of light in one direction, which is then spectrally birefringed, and the second film extracts only the specific color direction, resulting in a colored appearance.

# Sensing Concept



表情読み取りの様子



ネットワークストラクチャー

- 感情スコア : 個々の会話への姿勢
- 「neutral以外の総和」 - 「neutral」 = 感情スコア
- 音量 : 場の盛り上がり度

と定義した。この指標が大きいときは様々な感情が飛び交う「良い雰囲気」が形成されていると考える。話を弾ませている人の構造物の動きが大きくなる。これは、ファシリテーションにおいて、発言者にボールを持たせ、ボールを持つ発言者に注目を集める手法を再現したものである。

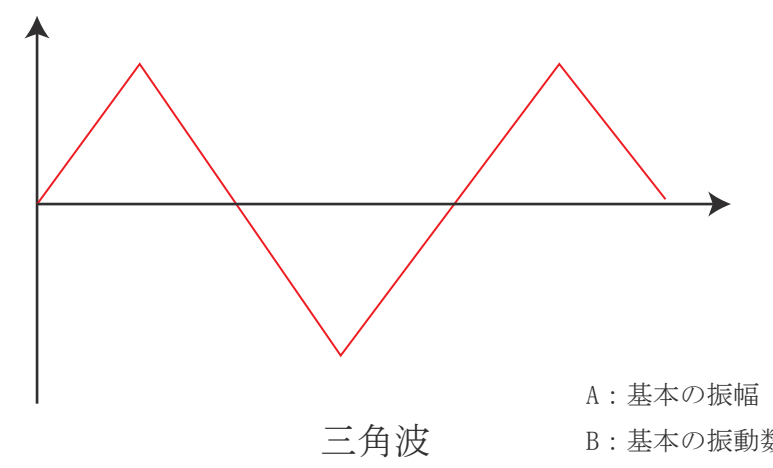


Emotion Score: Attitude toward individual conversation  
 Sum of all but neutral - Neutral = Emotion Score  
 Volume: Degree of excitement of the event.  
 When this index is high, we consider that a "good atmosphere" is formed where various feelings are exchanged. The structure leans toward the person who is speaking. This replicates a facilitation technique in which the speaker is given a ball to hold and the attention is drawn to the speaker holding the ball.



キーパーソンに向かって動く

# How to move



- 基本となる振動を1周期分作り、その振動をセンシング結果のスコアを元に以下の動きの変化
1. 振幅を大きく / 小さく → 音
  2. 振動を速く / 遅く → 個人の感情
  3. 盛り上がっている方向に寄る

- The basic vibration is created for one cycle, and the following movement changes based on the score of the sensing results
1. increase/reduce the amplitude → sound
  2. faster/slower vibration → personal emotion
  3. move closer to the direction of the upsurge

```
void setup() {
  pinMode(PIN1, OUTPUT);
  pinMode(PIN2, OUTPUT);
  pinMode(PIN3, OUTPUT);
  pinMode(PIN4, OUTPUT);
  pinMode(PIN5, OUTPUT);
  pinMode(PIN6, OUTPUT);
  pinMode(PIN7, OUTPUT);
  pinMode(PIN8, OUTPUT);
  pinMode(PIN9, OUTPUT);
  pinMode(PIN10, OUTPUT);
  pinMode(PIN11, OUTPUT);
  pinMode(PIN12, OUTPUT);
  pinMode(PIN13, OUTPUT);
  pinMode(PIN14, OUTPUT);
  pinMode(PIN15, OUTPUT);
  pinMode(PIN16, OUTPUT);
  pinMode(PIN17, OUTPUT);
  pinMode(PIN18, OUTPUT);
  pinMode(PIN19, OUTPUT);
  pinMode(PIN20, OUTPUT);
  pinMode(PIN21, OUTPUT);
  pinMode(PIN22, OUTPUT);
  pinMode(PIN23, OUTPUT);
  pinMode(PIN24, OUTPUT);
  pinMode(PIN25, OUTPUT);
  pinMode(PIN26, OUTPUT);
  pinMode(PIN27, OUTPUT);
  pinMode(PIN28, OUTPUT);
  pinMode(PIN29, OUTPUT);
  pinMode(PIN30, OUTPUT);
  pinMode(PIN31, OUTPUT);
  pinMode(PIN32, OUTPUT);
  pinMode(PIN33, OUTPUT);
  pinMode(PIN34, OUTPUT);
  pinMode(PIN35, OUTPUT);
  pinMode(PIN36, OUTPUT);
  pinMode(PIN37, OUTPUT);
  pinMode(PIN38, OUTPUT);
  pinMode(PIN39, OUTPUT);
  pinMode(PIN40, OUTPUT);
  pinMode(PIN41, OUTPUT);
  pinMode(PIN42, OUTPUT);
  pinMode(PIN43, OUTPUT);
  pinMode(PIN44, OUTPUT);
  pinMode(PIN45, OUTPUT);
  pinMode(PIN46, OUTPUT);
  pinMode(PIN47, OUTPUT);
  pinMode(PIN48, OUTPUT);
  pinMode(PIN49, OUTPUT);
  pinMode(PIN50, OUTPUT);
  pinMode(PIN51, OUTPUT);
  pinMode(PIN52, OUTPUT);
  pinMode(PIN53, OUTPUT);
  pinMode(PIN54, OUTPUT);
  pinMode(PIN55, OUTPUT);
  pinMode(PIN56, OUTPUT);
  pinMode(PIN57, OUTPUT);
  pinMode(PIN58, OUTPUT);
  pinMode(PIN59, OUTPUT);
  pinMode(PIN60, OUTPUT);
  pinMode(PIN61, OUTPUT);
  pinMode(PIN62, OUTPUT);
  pinMode(PIN63, OUTPUT);
  pinMode(PIN64, OUTPUT);
  pinMode(PIN65, OUTPUT);
  pinMode(PIN66, OUTPUT);
  pinMode(PIN67, OUTPUT);
  pinMode(PIN68, OUTPUT);
  pinMode(PIN69, OUTPUT);
  pinMode(PIN70, OUTPUT);
  pinMode(PIN71, OUTPUT);
  pinMode(PIN72, OUTPUT);
  pinMode(PIN73, OUTPUT);
  pinMode(PIN74, OUTPUT);
  pinMode(PIN75, OUTPUT);
  pinMode(PIN76, OUTPUT);
  pinMode(PIN77, OUTPUT);
  pinMode(PIN78, OUTPUT);
  pinMode(PIN79, OUTPUT);
  pinMode(PIN80, OUTPUT);
  pinMode(PIN81, OUTPUT);
  pinMode(PIN82, OUTPUT);
  pinMode(PIN83, OUTPUT);
  pinMode(PIN84, OUTPUT);
  pinMode(PIN85, OUTPUT);
  pinMode(PIN86, OUTPUT);
  pinMode(PIN87, OUTPUT);
  pinMode(PIN88, OUTPUT);
  pinMode(PIN89, OUTPUT);
  pinMode(PIN90, OUTPUT);
  pinMode(PIN91, OUTPUT);
  pinMode(PIN92, OUTPUT);
  pinMode(PIN93, OUTPUT);
  pinMode(PIN94, OUTPUT);
  pinMode(PIN95, OUTPUT);
  pinMode(PIN96, OUTPUT);
  pinMode(PIN97, OUTPUT);
  pinMode(PIN98, OUTPUT);
  pinMode(PIN99, OUTPUT);
  pinMode(PIN100, OUTPUT);
}

void loop() {
  // ... (code continues) ...
}
```