

羽虫の群れの動きのコントロール

村上 葉†

土橋 宜之‡

山本 強‡

†北海道大学大学院情報科学研究科

E-mail: †{murakai, dobashi, yamamoto}@ime.ist.hokudai.ac.jp

1 はじめに

近年、コンピュータグラフィックス (CG) 技術の発展により、様々な映像がリアルに再現できるようになった。CG 技術は映画やゲームにおける映像表現に多用されている。CG における重要な研究課題の一つとして人間や動物などの群衆の動きの再現が挙げられる。群衆の動きを再現するために、各個体の動きを一つ一つ手動で設定することは現実的ではないため、一定のルールに従って自動的に動きを計算する手法が用いられている [1]。しかし、CG において、人間や動物の群衆をリアルに再現する手法は様々なものが提案されているが、昆虫を対象としたものは少ない。

本研究では昆虫のうち、特に、羽虫の群れに着目し、その動きの制御を行う手法を提案する。羽虫の動きにはいくつかの特徴がある。例えば、各個体は、互いに適切な距離を保とうとしたり、捕食者に襲われないよう集団からはぐれないようにしたりする。このような様々な特徴を考慮して羽虫の動きを再現する手法が提案されている [2][3]。しかし、これらの手法を用いて羽虫の群れ全体をユーザが所望する位置に移動させたり、特定の範囲内に留ませたり、あるいは、群れの形状を制御したりすることは容易ではない。

本研究の目的は、羽虫の群れ特有の動きを維持したまま、群れ全体としての移動や形状の制御を行うことである。羽虫の群れの再現は Ren らの手法 [3] に基づいて実現する。この手法では、群れを構成する各個体の動きは、以下の3つの力を用いて計算されている。すなわち、個体間の位置関係から決まる力、障害物や餌などへの反応する力、羽虫特有の動きを再現するためのノイズ関数による力である。本研究では、これらの力に対して群れの形状を制御するための新たな力を導入する。そして、比例-微分制御 (PD 制御) を利用して制御力の大きさを自動的に調整する。

2 関連研究

動物などの群れをシミュレーションする研究の代表例として、文献で提案された手法 [1] がある。この手法では個々の個体に、簡単な条件を作用させ、群れ全体の動きを再現する。しかし、この方法では、羽虫特有の動きが考慮されていない。文献 [2] では、羽虫の群衆行動をシミュレーション

する手法が提案されている。この方法では、ノイズ関数とポテンシャル場を利用してより羽虫らしい動きを再現している。また、文献 [3] では、実際の羽虫の動きと比較することで、より本物に近い動きを再現することに成功している。本論文では、文献 [3] の手法を参考にしながら羽虫の群れの動きを再現する。そして、この群れの位置や形状などを制御する手法を提案する。

3 提案手法の概要

提案手法の概要を図 1 に示す。図 1(a) に示すように、ユーザは入力として、羽虫の数と各個体の初期位置 (初期分布) に加え、羽虫の群れが形成すべき目標形状を表す二値画像 (形状画像と呼ぶ) を与える。提案法は羽虫の群れの動きをシミュレーションし、群れ全体を与えられた視点方向から観察したときに、目標形状内に羽虫が存在し、かつ、目標形状内での羽虫の密度が一様となるよう群れ全体の動きを制御する。

図 1(b) に提案法による羽虫の動きの計算方法を示す。羽虫の動きのシミュレーションは文献 [3] の方法によって行う。この方法では個体ごとに羽虫の動きを再現するための力を計算する (図 1(b) の青い矢印)。この力は羽虫同士が適切な距離を保つための力、捕食者などの外的要因による力、ノイズによる力からなる。提案法では、これらに加えて、形状画像で与えられる目標形状の内部に向かう制御力を新たに加える (図 1(b) の赤い矢印)。しかし、この制御力だけでは羽虫の密度分布が一様になるとは限らず、偏りが生じてしまう場合がある。そこで、羽虫間距離を調整することにより一様な密度分布とする (図 1(b) の赤丸)。

以降、動きのシミュレーションのための力の計算方法、分布の形状制御のための力の計算方法、密度のための羽虫間距離の調整方法について順に説明する。なお、群れを形成する個体の数を N とし、簡単のため各個体の質量は 1 とする。

4 羽虫の群れのシミュレーション

本節では文献 [3] の手法を概説する。詳細については、文献 [3] を参照して頂きたい。各個体 i に力 $\mathbf{f}_{i,m}$ は以下のように表される。

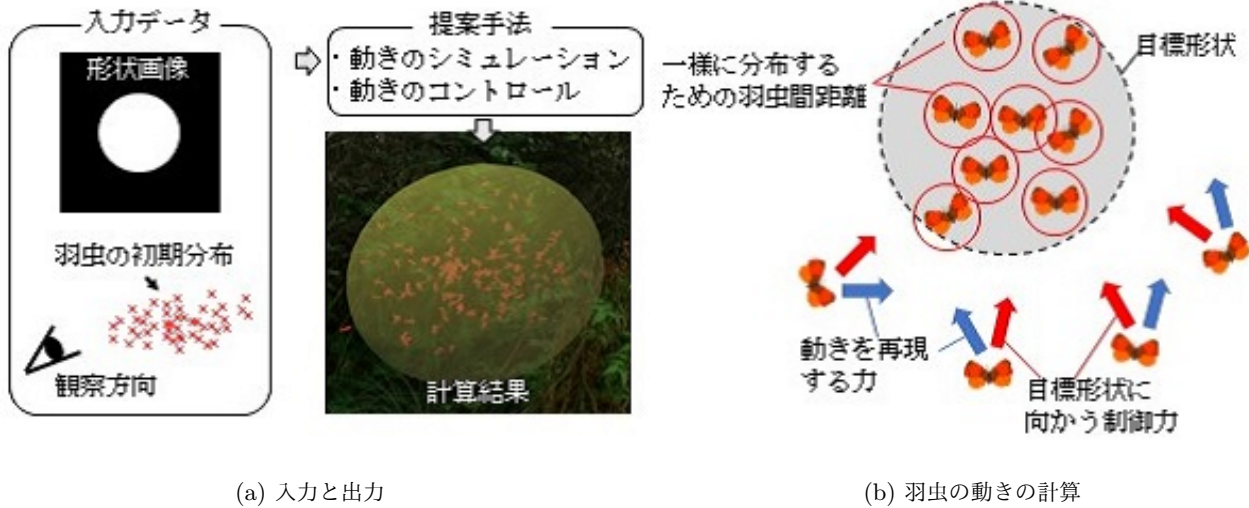


図 1: 提案手法の概要

$$\mathbf{f}_i, m = \mathbf{f}_{i, int} + \mathbf{f}_{i, pro} + \mathbf{f}_{i, \zeta} \quad (1)$$

ここで、 $\mathbf{f}_{i, int}$ は個体間で相互作用的に働く力 (interaction force) で、その距離を一定に保つように働く。 $\mathbf{f}_{i, pro}$ は空気抵抗や捕食者など外的要因により決まる力 (selfpropulsion force)、 $\mathbf{f}_{i, \zeta}$ はノイズにより決まる力を表す。以下、それぞれについて説明する。

$\mathbf{f}_{i, int}$ は次式によって計算される。

$$\mathbf{f}_{i, int} = \sum_k \mathbf{f}_{i, k} \quad (2)$$

$$\mathbf{f}_{i, k} = \frac{\chi_k}{N_k} \sum_{j=1}^{N_k} (g(r_{ji}) \hat{\mathbf{r}}_{ji} + (1 - |g(r_{ji})|) \hat{\mathbf{v}}_{ji}) \quad (3)$$

ここで、 $k = \{rep, ali, att\}$ を表し、それぞれ、個体同士が反発しあう状態、距離を保って同じ方向へ移動しようとする状態、引き付けあって距離を縮めようとする状態を表している。個体 i と j の状態は個体間の距離 $\mathbf{r}_i, \mathbf{r}_j$ によって区別され、その閾値は $r_{rep}, r_{ali}, r_{att} (0 \leq r_{rep} \leq r_{ali} \leq r_{att})$ である。 $\chi_k > 0$ は、各状態の作用を調整するための係数である。また、個体 i の位置および速度を、それぞれ \mathbf{r}_i および \mathbf{v}_i とすると、 $r_{ji} = \|\mathbf{r}_j - \mathbf{r}_i\|$ 、 $\hat{\mathbf{r}}_{ji} = (\mathbf{r}_j - \mathbf{r}_i)/r_{ji}$ 、 $\hat{\mathbf{v}}_{ji} = (\mathbf{v}_j - \mathbf{v}_i)/\|\mathbf{v}_j - \mathbf{v}_i\|$ である。関数 g は次式により与えられる。

$$g(x) = \begin{cases} -1 & (0 \leq x < r_{rep}) \\ 0 & (r_{rep} \leq x < r_{ali}) \\ 1 & (r_{ali} \leq x < r_{att}) \end{cases} \quad (4)$$

次に、 $\mathbf{f}_{i, pro}$ は次式によって計算される。

$$\mathbf{f}_{i, pro} = \mathbf{f}_{i, fric} + \mathbf{f}_{i, res} \quad (5)$$

ここで、 $\mathbf{f}_{i, fric}$ は空気抵抗に該当する力 (friction force) で、 $\mathbf{f}_{i, fric} = -\gamma \mathbf{v}_i$ で与えられる。 γ はユーザにより指定する。 $\mathbf{f}_{i, res}$ は捕食者や障害物、餌などに反応するための力 (response force) で、次式により与えられる。

$$\mathbf{f}_{i, res} = \chi_{res} H(r_{res} - r_{ie}) (s_e \hat{\mathbf{r}}_{ie} \mathbf{R}(\theta) + (1 - s_e) \hat{\mathbf{r}}_{ie}) \quad (6)$$

ここで、 χ_{res} はユーザにより指定する係数、 $H(r)$ はヘヴィサイドステップ関数を表す。 r_{res} は個体に対象物を視認できる最大距離、 $\hat{\mathbf{r}}_{ie}$ は個体 i から見た対象物の単位方向ベクトル、 $\hat{\mathbf{r}}_{ie}$ は対象物と個体 i の距離である。 $\mathbf{R}(\theta)$ は個体が逃避行動を起こす際のランダムな方向転換のための回転行列を表す。 s_e は逃避行動の場合は 1、追跡行動の場合は 0 をとる。

最後に、 $\mathbf{f}_{i, \zeta}$ は昆虫の本能的な行動を表す力で、文献 [3] に従い、Curl Noise[4] を用いて計算する。

5 羽虫の動きのコントロール

形状画像で与えられる目標形状内に収まるよう各個体の動きをフィードバック制御の一つである PID 制御手法を用

いてコントロールする。ただし、我々の実験によれば、I制御（積分制御）は有効に作用しなかったため、提案法ではP制御（比例制御）とD制御（微分制御）を用いる。各個体 i にかかる制御力 $\mathbf{f}_{i,c}$ は次式により計算する。

$$\mathbf{f}_{i,co} = \begin{cases} (\chi_p(\mathbf{r}_{tar,i}) + \chi_d \frac{d(\mathbf{r}_{tar,i})}{dt}) \mathbf{g}_i & (r_{tar,i} \in \Omega) \\ (0, 0, 0) & (r_{tar,i} \notin \Omega) \end{cases} \quad (7)$$

ここで、 χ_p および χ_d は各項の効果を調整するための係数、 $r_{tar,i}$ は個体 i から最も近い目標形状の境界上の点までの距離、 $\mathbf{r}_{tar,i}$ は個体 i の位置 \mathbf{r}_i から目標形状方向へ向かうベクトル、 Ω は目標形状の内部領域を表す。 $\mathbf{r}_{tar,i}$ は、形状画像に距離変換を施した画像を用いて計算する。また、 \mathbf{g}_i は形状画像にガウスフィルタを適用し、その勾配を用いて計算する。

式 (7) の動作について説明する。式 (7) で与えられる力は各個体が目標形状の外部に存在するときのみ作用し、目標形状内に到達するとゼロとなる。第1項および第2項は、それぞれ、P制御およびD制御に対応する。P制御（第1項）により目標形状との距離に比例した目標形状方向の力が働き、各個体を目標形状へと導く。しかし、P制御のみでは、個体が目標形状に到達した後も蓄積された慣性力が働いたため、目標形状内にとどまらないことが確認された。そこで、D制御（第2項）を併用する。D制御は、各個体の急激な動きの変化を抑制する。これによって慣性が蓄積することを防ぐことができる。

6 羽虫の密度のコントロール

前節の方法により、各個体が目標形状内に収まるようにコントロールすることができる。本節では、目標形状内での羽虫の密度の制御方法について述べる。密度が一様となるよう制御する方法に加えて、密度分布が一様でない場合についても説明する。ただし、密度が一様でない場合は実装には至っておらず、今後の課題とする。参考として理論の概要のみ記述する。

6.1 密度が一様な場合

形状領域内において密度を一様とする場合について説明する。この場合は、密度画像は形状画像と同じ二値画像となる。個体数が N 、目標形状の内部領域の面積を S としたとき、個体間の平均的な距離 R が次式を満たせば目標形状内に一様に個体が分布すると期待できる。

$$\pi R^2 = \frac{S}{N} \quad (8)$$

このことから、羽虫間の距離が $R = \sqrt{\pi S/N}$ となるよう各個体の動きを制御すればよい。一つの単純な方法として、各タイムステップにおいて、各個体の周辺の個体との距離を計算し、その距離が R となるように仮想的な外力を与える方法が考えられる。しかし、この方法では、外力の大きさを適切に調整しなければ、自然な動きが得られない。そもそも、自然な動きを実現するためには、できるだけ外力によらずに目的の密度へと収束する方が望ましい。

そこで、本研究では、4節で述べた群れのシミュレーションにおけるパラメータを適切に設定することで、自然な動きを実現する。具体的には、式 (4) における r_{rep} 、 r_{ali} および r_{att} を以下のように設定する。

$$\begin{cases} r_{rep} = aR \\ r_{ali} = bR \\ r_{att} = cR \end{cases} \quad (9)$$

ここで、 a, b, c はユーザにより指定する係数である。本論文で示す実験例では全て $(a, b, c) = (0.8, 1.2, 2.0)$ としている。

6.2 密度が一様でない場合

密度が一様でない場合への拡張について概要を述べる。この場合は、ユーザは、目標形状内の各点での目標密度を示す濃淡画像（密度画像と呼ぶ）を与える。与えられた視点方向から群れ全体を観察したとき、羽虫の密度が密度画像の対応する画素の輝度に比例するよう制御する。いま、観察方向に垂直な平面を考え、この平面を密度画像の画素数と同じ数の格子状の一様なセルに分割する。密度画像の画素数を n_x, n_y 、セル (i, j) に対応する密度画像の輝度を $I_\rho(i, j)$ とする。個体数が N のとき、セル (i, j) の個体数 N_{ij} が次式を満たせばよい。

$$N_{ij} = \frac{I_\rho(i, j)}{\sum_{k=0}^{n_x-1} \sum_{l=0}^{n_y-1} I_\rho(k, l)} \quad (10)$$

そこで、セルごとに式 (9) および (10) を用いて、 r_{rep} 、 r_{ali} 、 r_{att} を決定することで与えられた密度画像に比例するよう羽虫の群れを制御することができる。

7 実験結果

羽虫の動きと密度のコントロール法を実装し、提案法の有効性について検討した。ただし、前述したように、密度のコントロールについては、一様な場合についてのみ実装している。

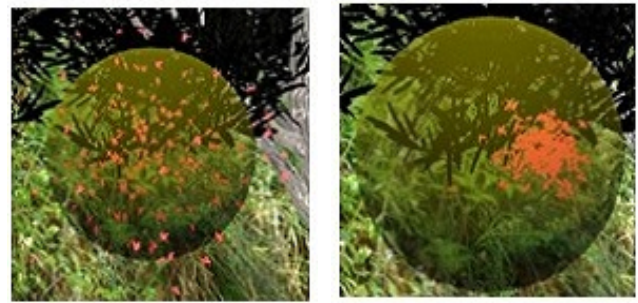
まず、5節で述べた動きの制御に関する実験を行った。図2および図3にその結果を示す。これらの例では、黄色で示す球体の内部に群れが移動するよう制御した。まず、式(7)に示すP制御とD制御の各項の効果を調べるため、それぞれの係数を変更して実験を行った。図2(a)はP制御の影響を大きくした場合を示している。この場合は、球体内部にはとどまらず、その周辺で振動し続ける個体が多く見られた。次に、図2(b)はD制御の影響を大きくした場合である。この場合は、球体内にとどまるが、球体に到達するまでに時間がかかってしまう。また、球体内に移動した後も不自然にかたまった移動の様子が観察された。図3(a)から(d)は各パラメータを適切な値に調整した場合の群れの移動の様子を示しており、全ての個体が球体内部にとどまるよう適切に移動の様子が確認された。

次に、6節で述べた密度が一様となるようコントロールする手法を用いた実験を行った。その結果を図4および図5に示す。図4では、形状画像を二つ指定し、'VC'という文字が形成された後、'2018'という文字が形成されるようにした。図4(a)から(d)は一つ目の目標形状から二つ目の目標形状へと変化する様子を示している。図5は、羽虫の群れが図5(d)の左上に示すドクロ形状を形成するようコントロールした例を示している。

8 まとめと今後の課題

本稿では、羽虫の動きをコントロールする方法について提案した。羽虫の動きは既存手法を用いて計算し、目標形状の位置に移動するよう新たな外力を加えた。外力の大きさはPD制御により自動調整する。また、目標形状内で羽虫の密度が一様となる方法について提案した。できるだけ自然な動きを保つよう羽虫間距離を自動調整することで実現した。提案法により、効果的なアニメーションが生成できることを確認した。

今後の課題として、まず、6.2節で述べた密度が一様でない場合の実装があげられる。これにより、濃淡のある画像を目標形状として指定でき、より興味深いアニメーションが生成できることが期待できる。また、目標形状が連続的な動画により与えられる場合についても検討したい。



(a) P 制御

(b) D 制御

図2: P制御とD制御の効果

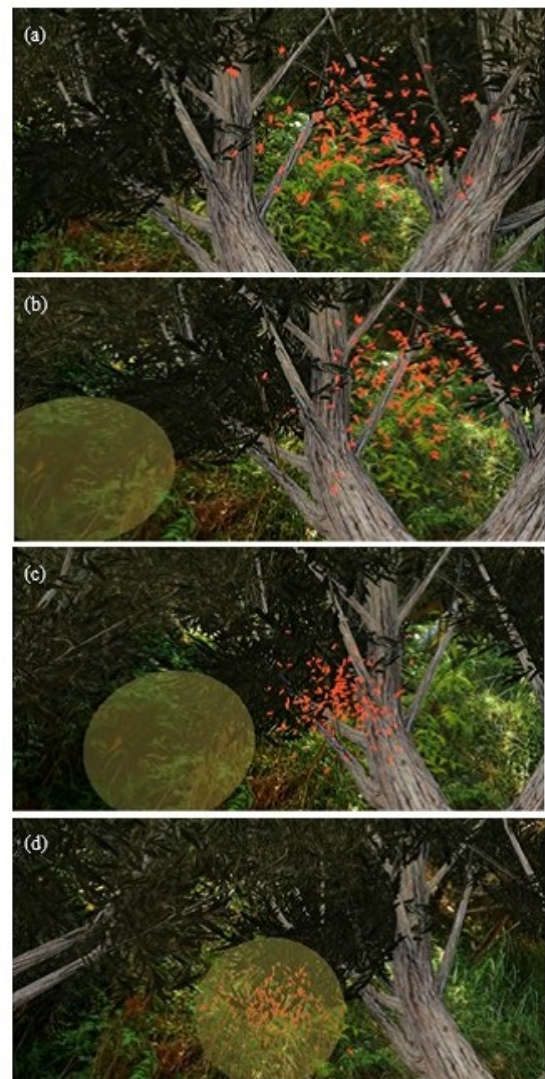


図3: 動きの制御結果

参考文献

- [1] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *SIGGRAPH Computer Graphics*, no. 2, 1987.

- [2] J. X. D. Z. Wang, X. and L. Zhou, “Inherent noise-aware insect swarm simulation.” in *Computer Graphics Forum*, no. 3, 2014.
- [3] R. J. J. X. M. D. Wang, X, “Bswarm: Biologically-plausible dynamics model of insect swarms,” in *SCA '15 Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, no. 1, 2015.
- [4] H. J. BRIDSON, R. and M. NORDENSTAM, “Curl-noise for procedural fluid flow,” in *ACM Trans*, no. 6, 2007.
- [5] R. A. MORSE, “Swarm orientation in honeybees,” no. 4, 1963.
- [6] L. Y.-X. LUKEMAN, R. and L. EDELSTEIN-KESHET, “Inferring individual rules from collective behavior,” in *Proceedings of the National Academy of Sciences*, no. 5, 2010.
- [7] B. J. M. DOMENICI, P. and J. P. BACON, “Animal escapology ii: escape trajectory case studies,” in *The Journal of experimental biology* 214, 15, 2474-2494, 2011.
- [8] G. S. L. M. VAN DEN BERG, J. and D. MANOCHA, “Reciprocal n-body collision avoidance,” in *In Robotics Research*, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds., vol. 70 of *Springer Tracts in Advanced Robotics.*, no. 7, 2011.

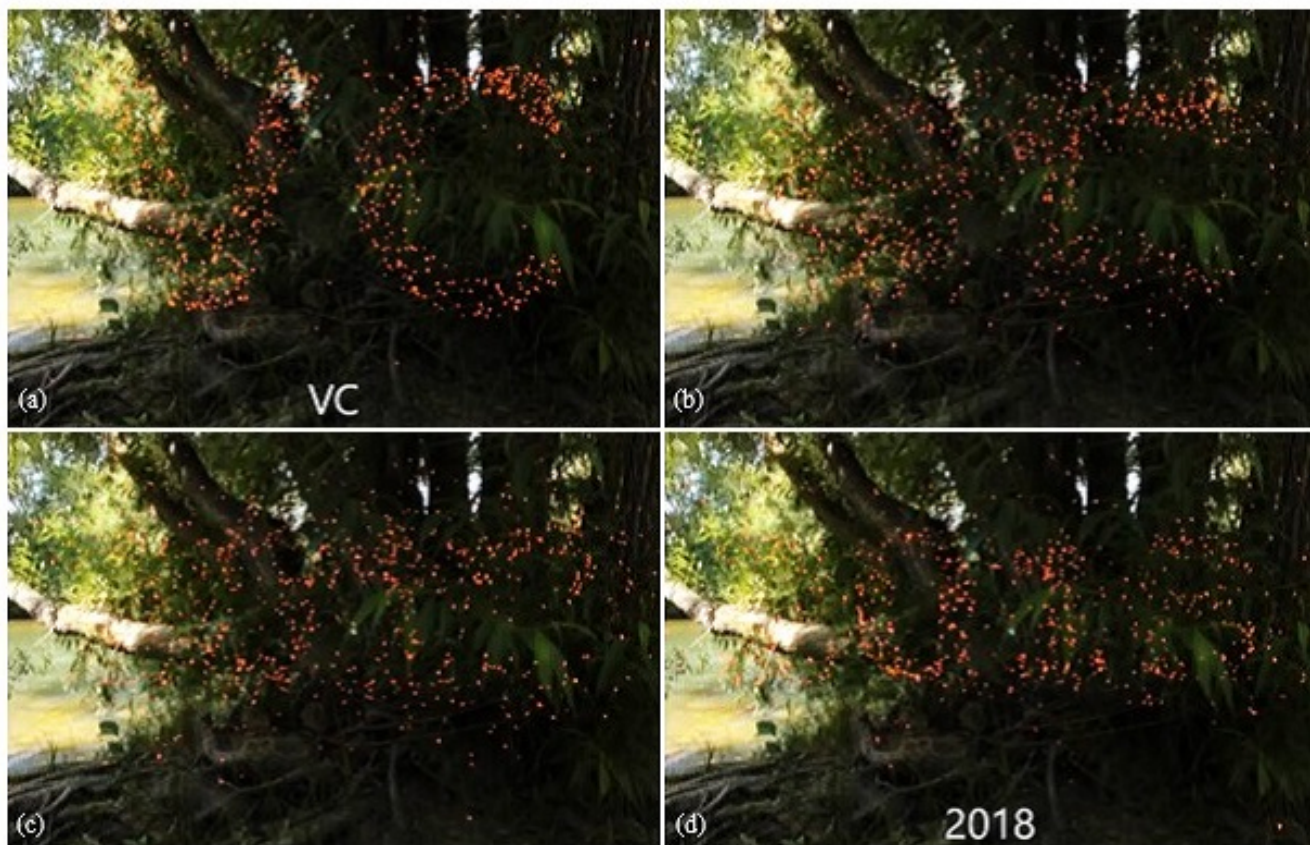


図 4: 文字を形成するように制御した例

