家紋—画像・テキストの新たなチャレンジ

スプロート・リチャード
Sakana.ai
rws@sakana.ai

概要

家紋は日本文化の大切な一部である. 本研究では, 7,408 個の家紋のデータセットについて報告する. データセットは, 画像, 家紋用語を使った説明, 依存関係と英語翻訳を含む. データに加えて, ベースライン画像・テキストシステムと合成家紋を作成するための文法ベースの生成システムを提供する. データはすべてオープンソースである.[1)] 一般的なテキスト・画像問題に比べて家紋はより制約のある問題である. 同時に, データのスパース性の問題がある. これらのデータをコミュニティに提供することで, 制約を活用してデータのスパース性に対処できるアプローチの開発への関心が高まることを期待している.

1　はじめに

家紋は鎌倉時代からの日本文化の一部である [1, 2, 3, 4, 5, 6, 7, 8, 9]. 13 世紀までに, 貴族たちは, 荷車に印をつける手段として紋を使っていた. 高澤等 [8] によると (3 頁)「一説に, 当時内裏に参内する公家が用いる牛車が大層混雑し, 退出してきた公家が自分の牛車を素早く識別するために, おのおの独自の紋章を車に施したといわれる.」 ほぼ同時期に, 武士たちは戦いの際に氏族を区別するために家紋を使い始めた. 特に後者の用途は, ヨーロッパの紋章と機能的に同一である.

ヨーロッパの紋章はまだエリートのものである. 例えば, 英国では紋章を受け取りたい場合は,「College of Arms」に申し込まなければならず, 紋章の権利がある (英語「armigerous」) ことを証明すべきだ [10]. 対照的に, 日本家紋は民主化された. ほとんどの各家族は自分の家紋がある.

ヨーロッパ紋章の形は「blazon」という言語に正式に説明される. 例えば, 図 1 は簡単な紋章は英国の

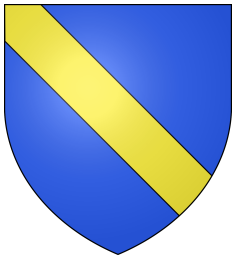

図 1 ヨーロッパ紋章の例,**azure a bend or**. Bear17(https://commons.wikimedia.org/wiki/File:Azure,_a_bend_Or.svg), CC BY-SA 3.0

「**azure a bend or**」(**azure**=青い, **bend**=ベンド, **or**=黄金) という blazon を示す. 英国の blazon は英語とフランス語が少し混ざったもののようであるが, 実際にフォーマルな言語である. Blazon は紋章の中の色々なモチーフ, モチーフの空間関係, 色などの情報を説明する. 家紋も blazon のような「家紋用語」というフォーマルな言語がある.

家紋には数百のモチーフがあり, それらモチーフには色々な組み合わせ方法がある. 多くの家紋で, 輪や角などの外殻の中には 1 つ以上の他のモチーフが含まれている. 図 2 は色々な例と「家紋用語」の説明を示している. モチーフ自体も変更される可能性がある. 例えば, (c) の「鬼」という変更では植物のモチーフを「鋭く尖らせて描くものである」[8] という意味がある. もう一つは「豆」という寸法を縮める変更である (図 4b). 家紋は数百のモチーフがあるが, 組み合わせや変更の可能性は限定されている. そのため, 家紋と家紋分析は高度に制約された問題である.

2　データセット

データセットは 3 つの情報源からの家紋画像を含んでいる. 一つ目は人文学オープンデータ共同利用センター[2)] の江戸時代の安政武鑑 (812 個) である.2 つ目は Wikimedia からのオープンソース家紋画像で

1) https://github.com/SakanaAI/Kamon, https://huggingface.co/datasets/SakanaAI/Kamon.

2) https://codh.rois.ac.jp/

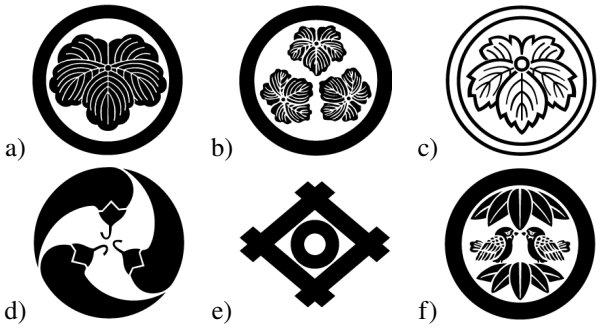

図 2　色々な家紋デザイン:a) 丸に蔦, b) 丸に尻合わせ三つ蔦, c) 総陰丸に鬼蔦, d) 唐辛子巴, e) 井桁に蛇の目, f) 丸に二弾五枚笹に対い雀

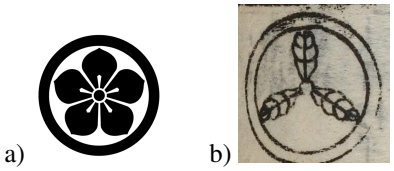

図 3　データベースから 2 つの例

ある (190 個). 3 つめは「Rebolforces」[3]からの画像である (6,406 個). 合計 7,408 個の紋がある. すべての画像は 224x224 に正規化されている. 江戸時代の例以外, 各画像は白黒である. 図 3 は (a)Wikimedia と (b) 江戸時代の例を示す. なお, 江戸時代のデータは他のデータとは非常に異なるため, 以下に報告する実験では使用されていない.

Wikimedia の家紋には家紋用語の説明がすでにあるが, 他の紋の説明は手作業で行った. LLM (Claude 3.5 Sonnet) を使って, 各説明の依存関係解析と英語翻訳が実行された.

インターネットには, たくさんの家紋専門サイト[4]があるが, そういうサイトのデータは全て独自のものである. 著者の知る限りでは, このようなオープンソースデータに関しては存在するのは我々のデータセットのみである.

3　合成データ

データ拡張のために, 文法に基づいた合成システムが開発された. モチーフは「外殻」と「他の」に区別した. 簡単な画像操作を使って「他の」のモチーフは「外殻」に配置される. モチーフも色々な組み合わせを使って, より複雑なモチーフを作れる. 図 4 は色々な操作を示す. 操作は再帰的であるため, 複数レベルのネストが可能である.

3)　https://github.com/Rebolforces/kamondataset
4)　例えば https://kamondb.com, https://irohakamon.com

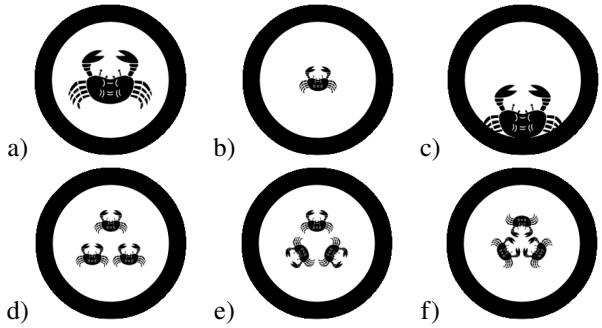

図 4　色々な合成された家紋: a) 丸に蟹, b) 丸に豆蟹, c) 丸に覗き蟹, d) 丸に三つ盛り蟹, e) 丸に尻合せ三つ蟹, f) 丸に頭合せ三つ蟹

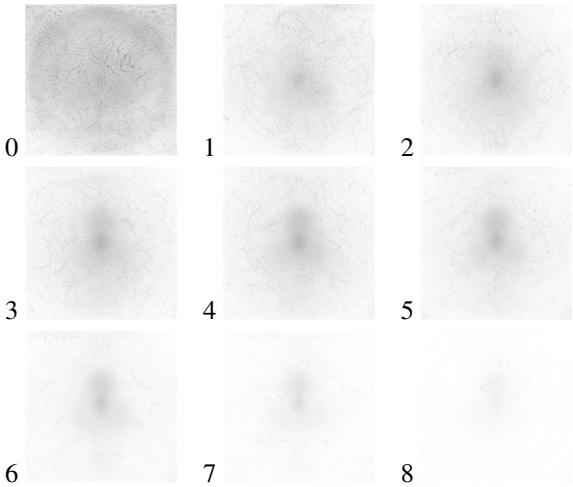

図 5　学んだ位置依存のマスクの形

4　ベースラインモデル

ベースラインとして, 簡単な Vision-Language Model(VLM) を開発した. モデルの主要部分は VGG(ディープ畳み込みニューラルネットワーク) である [11]. HuggingFace[5] の実装を使う. VGG は学習可能な特徴抽出器として使って, 最後の層 (dense layer) を取り除く (例えば [12]). 事前学習済みのウェイトには `IMAGENET1K-V1` を使う. 最後の層を取り除いた VGG は今後「VGG′」と呼ぶ. トレーニング中, VGG′ のウェイトもファインチューニングされる. VGG のに加えて, `openai/clip-vit-base-patch32` を使う CLIP モデル [13] の encoder のバージョンも提供する.[6] VGG′ または CLIP は今後「FE」(feature encoder) と呼ぶ。

VLM は出力記述内のトークンにそれぞれ対応す

5)　https://huggingface.co/learn/computer-vision-course/en/unit2/cnns/vgg
6)　学習可能特徴抽出器として, β-VAE (β-Variational Autoencoder[14]) を調査したが, 良い結果を得られなかった. そのため, ここで報告しない.

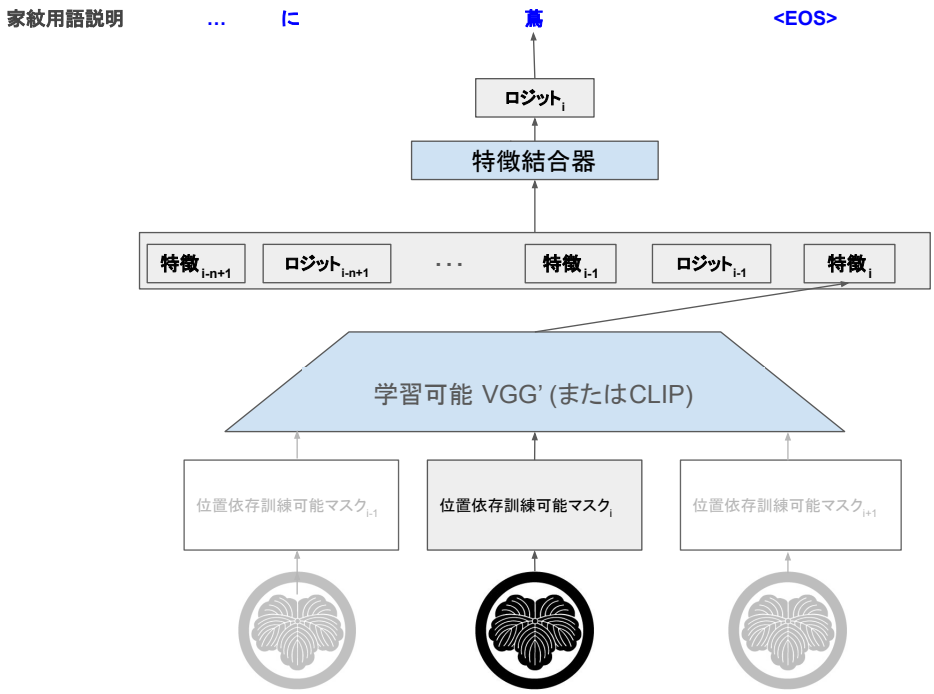

図 6　VGG′(または CLIP) モデルに基づいたベースライン VLM. 青い成分はポジション間で共有である

る複数の位置がある. 各位置では, インプット画像が学習できる位置依存のマスクに送られる. 家紋を「読む」時, 説明は外から内まで進む. マスクの背後にある考え方は, 単にモデルが見ているものを監視する手段として機能し, 注意機構よりも安価な方法である. 図 5 は合成データで学習されたモデル第 0–8 位置のマスクを示す. この図でマスクの画像は反転されて, より黒い部分はモデルが注目しているところを示す. 最初のマスクからわかるようにモデルははじめに画像全部を注目して, その後は内の部分を注目している.

マスクのアウトプットは学習できる FE に供給されて, FE の特徴は前の $n-1$ 個位置の特徴や前の $n-1$ 個位置の予測されたロジットと連結される. この組み合わせた特徴は特徴結合に供給される. n は ngram 長さである. 次の方程式には, I はインプット画像, M は位置依存のマスク, F_j は第 j の特徴, L_j は第 j のロジットで, FC は特徴結合器である:

$$F_j = \mathrm{FE}(I \cdot M_j)$$
$$L_i = \mathrm{FC}(\mathrm{Concat}[F_{i-n+1}, L_{i-n+1}, \ldots, F_{i-1}, L_{i-1}, F_i])$$

以下の実験では, $n = 3$. 特徴結合器は ReLU[15] を含む次の定義がある:

```
self.feature_combiner = nn.Sequential(
    nn.Linear(input_dim, hidden_dim),
    nn.ReLU(),
    nn.Dropout(0.1),
    nn.Linear(hidden_dim, hidden_dim),
    nn.ReLU(),
    nn.Dropout(0.1),
)
```

図 6 はベースラインモデルを示す.

5　ベースライン実験

5.1　定量分析

ベースラインモデルは江戸部分なし家紋データで評価した. 訓練部分は 5,276 個の紋を含んで, 検証やテスト部分には 660 個の紋がある. 訓練部分は標準的なノイズ (ガウシアン, ごま塩など [16]) を使って 9 倍拡張された. 検証で最小の文字エラー率 (character error rate—CER) まで学習された. テストの結果は表 1 に示す. 表には「Acc」(string accuracy) が文字列の精度を表し, 「$\mathrm{Acc}_{\mathrm{NIT}}$」(string accuracy not in training) がトレーニングデータに同じ説明のないことを表す. ご覧のように, 文字列精度は非常に低い. CLIP の検証損失, テスト CER や $\mathrm{Acc}_{\mathrm{NIT}}$ はよりいいが, 全体の Acc より低い. CLIP は明らかな勝利ではないので, 今後でより古いより簡単な VGG モデルの結果だ

表 1　ベースライン VGG と CLIP モデルのテスト結果. 損失=検証損失. テストデータ量=660

モデル	損失 ↓	CER↓	Acc↑	Acc_{NIT} ↑
VGG	3.26	0.389	**20.5%**	2.0%
CLIP	**2.72**	**0.369**	19.8%	**2.4%**

表 2　ベースライン VGG モデルのテスト結果: 合成データ. テストデータ量=1,000

CER	Acc	Acc_{NIT}
0.15	49.2%	45.0%

けを報告する.

合成データと同じ実験をした. この時, トレーニング部分は 8,000 個の紋を含み, 検証やテスト部分には 1,000 個の紋がある. テストの結果は表 2 に示す. 明らかに全体的な結果は悪く, このタスクでは改善の余地が多大にある. 合成データの結果は自然のデータの結果に比べて, 非常に良い. これは, 合成データのバリエーションがはるかに制限されているという事実を反映している.

5.2　定性分析

「鬼」(図 2c) のような細かい変換, モデルにとって難しい. 一方で, モデルは「ゲシュタルト」変換に得意である. 例えば, 一般的な変換は「桐」に関するものである. 普通の桐紋の例は図 7(a) に示す. 図 7(b) では鈴で作った桐形の「鈴桐」である. モデルの予測は「瓜桐」である.「瓜」は勿論正しくないが,「桐」の形のモデルが検出できた.

5.3　合成データと実データ

もし合成データを追加したら, 実データのテスト結果を改善できるだろうか. これをテストするために, 実の訓練データに合成データを追加し増強した. 結果セットは 13,276 個例, これの中に 7,976 個の合成例, 検証セット 656 個の実の例, テストセット 640 個の実の例を含んでいる. 表 3 は結果を示す. 合成データを追加の場合には, 検証の損失が大幅に減少した. テストの文字エラー率と Acc_{NIT} はわずかな改善も見られて, Acc も少し改善した. これは合成データが役に立つことを示唆している.

6　VLLM (Very Large Language Model) の能力

VLLM はこの問題をすでに解決したのだろうか. この質問に答えるために, 合成家紋のテストセットから最初の 20 個の例を選択した. 2 つの

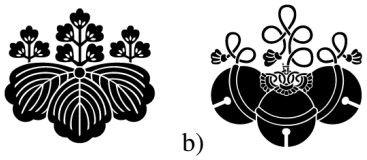

図 7　a) 五三桐, b) 鈴桐

表 3　実データ対実データ＋合成データ

データ	損失 $_{\text{val}}$ ↓	CER↓	Acc↑	Acc_{NIT} ↑
実	3.26	0.389	20.5%	2.0%
+ 合成	**1.05**	**0.376**	**21.7%**	**3.9%**

VLLM—GPT-5, Gemini-3 Pro—に 5-ショットプロンプト (紋と説明) を提供して, 20 個の合成例の説明を書くように指示した. トレーニング中に VLLM がどのようなデータにさらされたかわからないため, 実際の例を使用できないことに注意すべきである. 表 4 は結果を示す. 付録で, 表 5 は例の説明の参照転写や各モデルの予測を示す. 明らかに GPT や Gemini などの VLLM は家紋についてある程度の知識があるが, 十分とは言えない.

表 4　ベースラインと VLLM の文字エラーの比較. テストデータ量=20

モデル	CER↓	Acc↑	Acc_{NIT}↑
ベースライン VGG	**0.29**	**40%**	15%
GPT-5	0.78	0%	—
Gemini-3 Pro	0.82	0%	—

7　チャレンジのまとめ

本研究では, オープンソース家紋データセットに基づいた新たな画像・テキストタスクとベースラインを提供する. VGG(または CLIP) に基づいた VLM のベースラインは,「ゲシュタルト」の特性を捉えることができるが, 全体的なパフォーマンスが良くなく, タスクは改善の余地が多大にある. 一方, VLLM がこの問題を解決したと考える人もいるかもしれないが, 本研究はこれがそうではないと実証した. 家紋の構築は理論的に無制限であるが, 実際に大部分の紋は使うモチーフは多くない. 同時に, 家紋のデータ量, 特にオープンソースデータは, 限られている. したがって, このタスクはドメイン知識や制約を賢く使える技術を開発するような機会となる. そのような仕事への興味を奨励するため, このデータセットを研究コミュニティに提供している.

謝辞

人文学オープンデータ共同利用センターの北本朝展教授に安政武鑑のデータに感謝申し上げます.

本論文の校正にご協力いただいた林正頼氏や青木悦子氏にも心より感謝申し上げます.

参考文献

[1] Hugo Gerard Ströhl. **Japanisches Wappenbuch "Nihon Moncho"**. Verlag von Anton Schroll, Wien, 1906.

[2] John Dower. **The Elements of Japanese Design**. John Weatherhill, New York, 1971.

[3] 千鹿野茂. 日本家紋総鑑. 角川書店, 東京, 1993.

[4] 森本景一. 女紋. 染色補正森本, 京都, 2006.

[5] 高澤等. 家紋の辞典. 東京堂出版, 東京, 2011.

[6] 森本勇矢. 日本の家紋大辞典. 日本実業出版社, 東京, 2013.

[7] David Phillips. **Japanese Heraldry and Heraldic Flags**. Flag Heritage Foundation, Danvers, MA, 2018.

[8] 高澤等. 家紋大辞典. 東京堂出版, 東京, 2021.

[9] Richard Sproat. **Symbols: An Evolutionary History from the Stone Age to the Future**. Springer Nature, Cham, Switzerland, 2023.

[10] Arthur Charles Fox-Davies. **A Complete Guide to Heraldry**. Dodge Publishing, New York, 1909.

[11] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. https://arxiv.org/abs/1409.1556.

[12] Mayank Mishra, Tanupriya Choudhury, and Tanmay Sarkar. CNN based efficient image classification system for smartphone device. **Electronic Letters on Computer Vision and Image Analysis**, pp. 1–7, 2021.

[13] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[14] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. In **International Conference on Learning Representations**, 2017. https://openreview.net/forum?id=Sy2fzU9gl.

[15] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In Geoffrey Gordon, David Dunson, and Miroslav Dudík, editors, **Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics**, Vol. 15 of **Proceedings of Machine Learning Research**, pp. 315–323, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. https://proceedings.mlr.press/v15/glorot11a.html.

[16] Robert Fisher, Simon Perkins, Ashley Walker, and Erik Wolfart. Image synthesis–noise generation. https://homepages.inf.ed.ac.uk/rbf/HIPR2/noise.htm.

A　VLLM とベースラインモデルの比較

表 5　合成テストデータの最初 20 個の例で VGG ベースラインモデルと VLLM の文字エラーの予測. 予測と参照テキストが同じ場合は予測のテキストを青色で示す.

家紋	説明の参照テキスト	VGG-ベースライン	GPT-5	Gemini-3 Pro
	一重亀甲に三つ盛り杏葉山吹	一重亀甲に三つ盛り真向き板屋貝	亀甲に三つ蔦	隅入角に三つ盛り亀甲花菱
	糸輪に三つ割り重ね打板	糸輪に三つ割り重ね打板	丸に唐草巴	丸に細輪に地紋散らしの剣片喰
	隅入り鉄砲角に太陰光琳蔦	隅入り鉄砲角に太陰光琳蔦	菱に梅鉢	隅切角に梅鉢
	細輪に架み鷹	細輪に架み鷹	丸に止まり鷹	丸に鷹の羽
	隅切り角に飛び三羽雀	隅切り角に飛び三羽雀	八角に向かい兎	隅切角に亀甲に鶴
	隅立て角に石持ち地抜き松皮菱	隅立て角に地抜胴角	菱に矢羽根	隅切角に菱
	抱き角に重ね糸巻板	抱き角に三つ鱗	鬼火焔	違い茗荷
	亀甲に葉付き三つ横見梅	亀甲に三つ割り葉付き崩し	亀甲に唐花	隅切角に亀甲に花菱
	反り亀甲に覗き立ち銀杏の丸	反り亀甲に覗き八重柏	剣片喰	隅切角に星
	隅切り角に三つ盛り鞠鋏み	隅切り角に三つ盛り鉄砲	八角に六つ星	隅切角に三つ盛亀甲花菱
	丸に尻合せ三つ蔓葵片喰	丸に尻合せ三つ八重向う	丸に三つ梅	丸に三つ盛り亀甲花菱
	亀甲に台洲浜	亀甲に台洲浜	亀甲に違い柏	隅切角に六角に花菱
	丸に井筒	丸に山山井筒	丸に井桁	丸に角立て四つ目
	外藤輪に豆西条三つ葵	外藤輪に豆丸梅鉢	輪違い柏に角梅	大岡越前に抱き茗荷
	毛輪に七本骨雁木扇	毛輪に七本骨雁木扇	丸に扇	丸に地抜き扇
	細隅入り角に三つ追い杜若	細隅入り角に三つ追い杜若	菱に龍巻	隅入角に地抜き三つ巴
	月輪に丸に左三つ巴	月輪に丸に左三つ巴	丸に三つ巴	丸に一つ巴
	源氏輪に尻合せ三つ子付き三つ巴	源氏輪に尻合せ三つ変り二つ巴	総陰丸に割三つ巴	地抜き丸に三つ巴
	陰隅切り角に豆生花桔梗	陰隅切り角に豆変り竜胆鎧蝶	八角に小菊	八角に松
	菊輪に乱れ梶の葉	菊輪に葉折れ梶の葉	菊輪に蔦	団扇に蔦