Influence beyond similarity:
A Contrastive Learning approach to Object Influence Retrieval

Teresa Liberatore[1], Paul Groth[1], Monika Kackovic[1], Nachoem Wijnberg[1 2]

University of Amsterdam, Amsterdam, Netherlands
University of Johannesburg, Johannesburg, South Africa
{t.liberatore,p.t.groth,m.kackovic,n.m.wijnberg}@uva.nl

Abstract. Innovative art or fashion trends do not spring out of nowhere: they are products of societal contexts, movements and economic turning points. To understand the dynamics of innovation, it is necessary to understand influence relations between agents (e.g. artists, designers, creatives) and between the objects (e.g. clothes, paintings) that these agents produce. However, acquiring knowledge about these connections is challenging given that they are frequently undocumented. Recent literature has focused on discovering influence relations between agents, utilizing either object similarity or social network information. However, these methods often overlook the importance of direct relations between objects or oversimplify the complex nature of influence by approximating it with similarity.
To overcome this gap, we introduce Object Influence Retrieval (OIR), a task aimed at retrieving objects that potentially influenced a given object. To measure task performance, we describe two datasets for OIR: WikiartINFL (paintings) and iDesignerINFL (fashion items), both enriched with agent influence information. Additionally, we present CLOIR, a Contrastive Learning approach leveraging transfer learning from a pretrained model to represent objects, incorporating agent influence information through contrastive learning. CLOIR shows up to a 30% improvement in Precision@k and Mean Reciprocal Rank in the OIR task compared to a baseline based on similarity between objects.

Keywords: Creative Influence · Computational Creativity · Knowledge Discovery · Content Based Image Retrieval · Contrastive Learning

1 Introduction

Determining the technology upon which an innovation is built or identifying the inspirations behind a painting can help uncover patterns of influence [1,2]. The web of influence relations that shape innovations across various domains has long fascinated academics and practitioners alike. Being able to acquire knowledge about the underlying influence relations between objects and agents is crucial to understanding the complexity of the creative process [3] and multifaceted nature of innovation [4].

However, acquiring such knowledge is challenging because it is often undocumented. For example, while academics explicitly reference their peers' papers, this is not true in other domains. A case in point is that painters often find inspiration in their peers' content, style, or approach; nevertheless, they do not declare it explicitly. Similarly, there is no established convention for explicitly acknowledging sources of inspiration in other creative industries, like fashion, design, architecture, and literature.

To acquire such knowledge at scale, systems are needed to help discover such influence information. Prior work has focused on relations between agents (i.e. individual creators) [5,6]. This is likely due to the fact that ground truth data for developing models is available in specific domains such as fine arts and music, which have been extensively studied by domain experts [5]. On the other hand, the task of retrieving influence relations between objects has been largely overlooked.

Hence, in this study, we introduce a new task - Object Influence Retrieval (OIR) - aimed at retrieving objects that potentially influenced another object. Along with the task, we present two datasets to develop and evaluate approaches to perform OIR: WikiartINFL, a collection of paintings with metadata enriched by artist influence information, and iDesignerINFL, which includes images of fashion items created by renowned designers also with corresponding influence relations. We also introduce CLOIR, a Contrastive Learning approach to perform OIR on the presented datasets. In CLOIR, a contrastive learning model is trained to represent objects in an embedding space that accounts for both (i) the similarity between objects and (ii) the influence relation between creators.

CLOIR outperforms baselines where similarity between objects serves as a proxy for influence, suggesting that CLOIR is better suited for finding potential influence between objects compared to similarity alone.

Summarizing, the main contributions of this paper are as follows:

1. Object Influence Retrieval (OIR): a new task with the goal of, given an object, retrieving the objects that potentially influenced it;
2. Contrastive Learning Object Influence Retrieval (CLOIR): a Contrastive Learning approach to solve OIR;
3. WikiartINFL and iDesignerINFL: two datasets augmented with agent influence information for evaluating approaches for solving OIR.

2 Related Work

To the best of our knowledge, this is the first study focused on retrieving object influences. Previous research on influence detection has primarily addressed the reverse problem: identifying influences between agents using either object similarity or social network information. Content-based image retrieval (CBIR) has explored finding similar images given a query image, but the problem of influence retrieval has not been explored yet.

Object similarity to determine agent influence: The literature [7,8,6,9] adopting this approach focuses on the fine art domain, where agents are artists and objects are artworks. A key characteristic of artworks is that, like pictures, they can be fully represented through their visual depiction. Additionally, in the fine art domain, art experts have extensively documented the influence of artists, providing reliable ground truth. Although the methodologies vary, these studies share a common framework:

1. Artworks are represented as feature vectors.
2. A similarity score is computed between these feature vectors.
3. The similarity between artworks is used to infer similarity between artists and suggest influence among the artists.
4. The discovered influences are then evaluated against the established artistic influence ground truth.

Social network information and agent influence: Most works that use social interactions to find influence between agents focus on the music domain [5,10]. In the music domain, there is abundant knowledge about interactions between artists, and the influence between objects is often made explicit through samples or covers of existing songs. In particular, one paper focuses on modeling the interactions between agents within the music domain using Knowledge Graph and Semantic Web technologies [10], whilst other works analyze interactions in the music industry explicitly linked to influence, such as sampling and covering [11,12]. Another work uses graph theory on artists' social networks to predict the corresponding influences [5].

In contrast to prior work on influence detection, our paper focuses on the connections between objects, introducing the OIR task and the CLOIR approach that incorporates object similarity with information about agent influences.

Content Based Image Retrieval Image retrieval is a well-studied problem where given query images, similar images are retrieved from a database [13]. Among works in CBIR, studies that focus on unsupervised or pseudo-supervised CBIR are of particular interest for the problem at hand given that there is a lack of ground truth object influence information. In particular, [14,15] overcome the lack of supervision using a triplet network, where pseudo-labels for positive and negative examples are based on image similarity.

Drawing from these works, in CLOIR, we use a triplet network to shape the embedding space for retrieval, but differently from them, we aim at influence rather than similarity-based retrieval. Thus, the sampling of positive and negative examples is based on agent influences rather than similarity. We hypothesize that this dual-faceted representation will enhance the object representation space for OIR, compared to similarity alone. Our work thus introduces an operationalization of influence between objects that goes beyond object similarity.

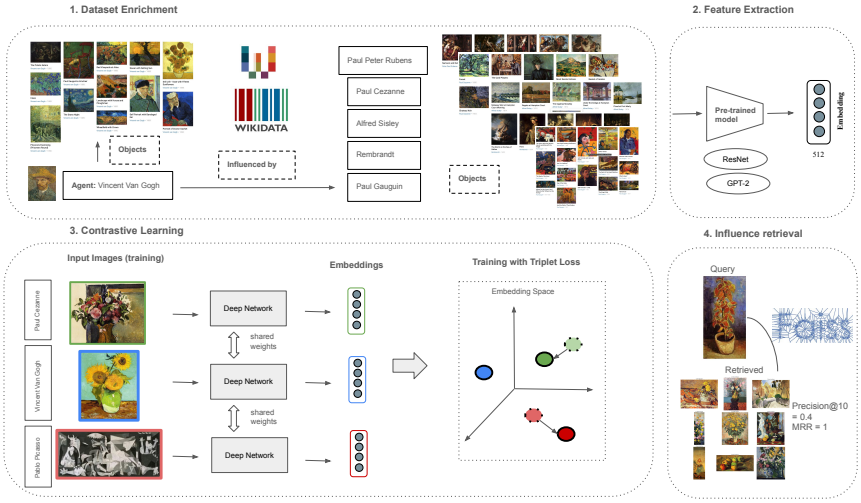

Fig. 1. Overview of CLOIR: Contrastive Learning approach to Object Influence Retrieval.

3 Contrastive Learning Object Influence Retrieval

An overview of the approach is illustrated in Figure 1 where the aim is, given an object, retrieve which objects potentially influenced its creation. We focus on objects whose main characteristics can be depicted through visual features, such as artworks and fashion items. However, our approach can be extended to domains where objects are represented in other modalities, such as text or audio, provided that (i) the modality, or a combination of modalities, can represent the fundamental characteristics of the objects, and (ii) pre-trained models can be used to extract features from the modalities of interest.

Specifically, with CLOIR, we aim to structure the object embedding space so that objects with visual similarities and which are produced by agents that are connected via their influence are positioned closely together.

Source code available at https://github.com/traopia/CLOIR .

3.1 Dataset Enrichment

The initial step involves sourcing information about influence between agents which is then mapped to the objects created by the agents.

Information about agent influence can be sourced in different ways. For WikiartINFl, for instance, it was sourced by querying Wikidata and Wikiart, as depicted in Figure 1.

As a result, the enriched dataset for each observation should include (i) the object, (ii) the agent who produced it, (iii) the known influencers of the agent, and (iv) any additional metadata, if available.

3.2 Feature Extraction

The objects are represented with vector embeddings, leveraging vision pre-trained models, following prior works on fine-art object representations [16,17,18]. If metadata is available, text features are extracted from a language pre-trained model and concatenated to the visual embeddings.

We experiment with two different setups of pre-trained models to extract visual and textual features from data: a combination of ResNet34 and GPT2-small, and CLIP.

ResNet34 and GPT2 In the first setup we extract visual features with ResNet34 [19][1], a widely-used convolutional neural network architecture pre-trained on ImageNet; and text features with GPT2-small [20][2], a language model known for its proficiency in natural language processing tasks. Our choice of ResNet-34 and GPT-2 small, over other larger pre-trained models such as ResNet-50 or GoogleNet for vision, and BERT or GPT-4 for language, is motivated by the balance between model complexity and computational efficiency.

CLIP In the second setup, we extract both visual and text features, using CLIP [21], a neural network trained to learn visual concepts from natural language. The multi-modal nature of the model allows us to extract both visual and text features from the same model, allowing for a smoother pipeline of feature extraction.

3.3 Contrastive Learning

The core of CLOIR revolves around fine-tuning the object representations obtained from pre-trained models according to the information about the influence between agents. In particular, we aim to shape the embedding space such that objects are proximal if (i) they are similar and (ii) the agents that produced them are linked through an influence relation.

We achieve this goal by combining approaches proposed in Multiple Instance Learning (MIL), fine-grained feature representation, and Content Based Image Retrieval (CBIR). MIL is a form of weakly supervised learning where training instances are arranged in sets, called bags, and a label is provided only for the entire bag [22]. In our case, the training instances are the objects arranged by agents, and the labels represent the influence relations. Specifically, from the MIL literature[23], we adopt the idea of mapping the class-level label (influence between agents) to all the objects and using an objective function based on instance-level similarity, which also respects the group-level label constraints. An objective function with these characteristics is the triplet constraint, which is also used in fine-grained feature representation tasks [24], where the goal is

[1] https://huggingface.co/microsoft/resnet-34
[2] https://huggingface.co/openai-community/gpt2

to distinguish subordinate classes by identifying instances with the same attributes. The triplet loss, learning from positive and negative examples for each learning anchor, encourages the model to identify objects sharing the same attributes, while preserving intra-class variation within the sub-classes. Triplet loss is widely used in CBIR too, because it directly optimizes the embedding space for similarity-based retrieval, ensuring that similar images are close together and dissimilar ones are far apart, thus improving retrieval performance [13].

To suggest influence between objects, we thus train a pseudo-supervised contrastive model with triplet loss, with pseudo-supervision coming from labels about influence between agents, as we don't have labels about object influences.

Triplet Construction For an anchor object, we consider as positive examples the objects made by agents influential for the agent who produced the anchor object. On the other hand, negative examples are objects made by agents not considered influential. Both the pools of potential positive and negative examples are extensive and exhibit a high degree of internal variability. Therefore, samples of positives and negatives are considered for each anchor object, and in CLOIR we experiment with different sample sizes and sample strategies for triplet construction.

Triplet Network The triplet network aims to maximize the discrimination of image representations, and it consists of three same networks that share weights. The triplet loss minimizes the distance between an anchor and positive examples and maximizes the distance between the anchor and negative examples.

Loss: In particular the triplet loss is defined as:

$$L(a, p, n) = \max\{d(a_i, p_i) - d(a_i, n_i) + \text{margin}, 0\}$$

where a_i represents the embedding of the anchor sample, p_i of the positive sample, n_i of the negative sample, d is the function measuring the distance between the samples, and the margin is a hyperparameter defining the minimum margin between positive and negative distances.

Model: The contrastive model trained with triplet loss is designed as a feed-forward neural network composed of three linear layers with ReLU activation functions and dropout layers in between. The model reduces the dimensionality of the input features coming from the pre-trained models, to learn the patterns within the feature vectors. Training batches of anchor, positive, and negative examples are thus passed through the same network, whose weights are updated according to the triplet loss.

Implementation details The models are trained for 30 epochs with early stopping with patience set to 10 epochs, and batches of size 32. The optimization of the loss function is done with Adam optimizer with learning rate set

to 0.0005, as suggested in [25]. To allow reproducibility and consistency among experiments, thus removing the randomness involved, a random seed is set to 42. All experiments have been performed on a GPU partition on an NVIDIA A100 GPU node.

3.4 Object Influence Retrieval

The final stage of our approach involves retrieving influential objects for a given query object via a vector search within the fine-tuned embedding space. To evaluate this, objects in the test set — those not previously encountered by the model as anchors, positive, or negative examples during training — are used as query objects. For each query object, we retrieve k-nearest objects in the trained embedding space. The FAISS library [26], which specializes in embedding similarity search tasks, is employed for this purpose, using Euclidean distance as the metric for vector closeness.

Evaluation As is standard in information retrieval systems, performance is assessed using metrics such as Precision at K and Mean Reciprocal Rank (MRR). Given that there is no existing ground truth for this task, we make use of agent influence to compare performance. Specifically, retrieval is deemed correct if the retrieved object was created by an agent recognized as influential to the agent who created the query object. Additionally, to account for chains of agent influences, we extend these metrics to a second degree, where retrieval is also considered correct if the retrieved object was created by a direct influencer or an influencer of an influencer of the query object's agent.

4 Experiments

To evaluate the performance of our proposed approach, we conduct a series of experiments comparing our results against a baseline model. The baseline model retrieves potential influence objects, based solely on object similarity. We investigate various configurations within our approach, focusing on different sampling strategies, for example, selection in the contrastive model, and varying training/test splits.

Sampling strategies The core of our method is a contrastive model that requires both positive and negative examples for each query. We explore the impact of different sample sizes and sampling strategies on model performance. Specifically, we use sample sizes of 10 and 100, adhering to the minimum object requirement per artist. For positive examples, we compare random sampling against similarity-based sampling. We hypothesize that similarity-based sampling will improve performance by selecting examples that are more likely to be semantically related to the anchor objects, thereby reducing variability within the samples.

Training/Test split Additionally, we experiment with different training/test splits of the dataset. The first split is stratified, with the training set containing 70% of the objects from all agents and the remaining 30% in the test set. The second split, the Leave-out Agents split, is designed to evaluate the model's ability to retrieve object influences for agents not seen during training. In this configuration, the training set includes objects from 70% of the agents, while the remaining 30% of agents are excluded from the training set and reserved for the test set.

4.1 Data

We introduce two distinct datasets to evaluate our approach: WikiartINFL and iDesignerINFL. Both datasets contain objects, whose visual representations capture their main characteristics, along with the names of the agents who created them. In particular, WikiartINFL includes images of artworks made by artists, and iDesignerINFL images of fashion items made by fashion designers.

WikiartINFL Dataset The Wikiart dataset[3] is a comprehensive collection of paintings and their associated metadata. It is one of the largest online repositories of digitized paintings and is frequently utilized to develop computational approaches to study fine arts. We use previously curated Wikiart data presented in prior studies [27]. This original dataset includes 75,921 artworks encompassing paintings, drawings, and illustrations.

We extended the Wikiart dataset with artist influence relations to create WikiartINFL. This information was gathered by querying Wikidata and Wikiart, specifically utilizing the "influenced by" property (P737)[4] to capture influence connections between artists within the dataset. We considered only artists with over 100 artworks in the painting collection and retained only those artists whose influencers were also present in the dataset. This selection ensures access to the artworks created by influential artists, which is essential for gathering positive examples to train a contrastive model.

iDesignerINFL Dataset The iDesigner dataset[5] contains images of fashion items captured during runway shows of various designers. This dataset, introduced on Kaggle by Hearst magazine, was used in a challenge to predict which fashion designer created each item. The dataset includes multiple images of the same items taken from different angles during runway shows. To prevent data leakage, we ensured that images of the same item were assigned to the same split during the training and test phases. We considered images to refer to the same item if they exceeded a 95% similarity threshold and were made by the same designer.

[3] https://www.wikiart.org
[4] https://www.wikidata.org/wiki/Property:P737
[5] https://paperswithcode.com/dataset/idesigner

To source information about the influences between fashion designers for the iDesigner dataset, we utilized a Large Language Model (LLM). This decision was driven by the scarcity of accessible information on designer influences and the potential of LLMs to provide labels when they are otherwise unavailable. Demonstrating that meaningful results can be achieved using influence data sourced via a LLM suggests that this approach could be applied in other domains with similarly scarce information. In particular, we prompted GPT-3 with the following query to gather information on designer influences: "Can you help me find the fashion designers that influenced the designers in this list? Specifically, can you create a dictionary where the keys are designers from the list and the values are their influencers, chosen from the same list of designers?".

	WikiartINFL	iDesignerINFL
number of objects	39815	44204
number of agents	154	49
mean objects per agent	258	902
mean influencers	2.8	1.5
min,max influencers	1, 10	1, 3

Table 1. Descriptive statistics for WikiartINFL and iDesigner.

Table 1 presents statistical summaries of the final versions of the WikiartINFL and iDesignerINFL datasets, both enriched with influence information. These statistics offer insights into the characteristics of the datasets after preprocessing and agent influences incorporation. Notably, while the two datasets are similar in size, WikiartINFL includes roughly three times the number of agents compared to iDesignerINFL. This implies that, on average, more objects are available per agent in WikiartINFL. However, it is important to note that for iDesignerINFL, these numbers may not reflect distinct objects, as multiple images can depict the same runway fashion item. Conversely, WikiartINFL reports nearly double the number of influencer agents per agent on average, resulting in greater variability across positive examples.

5 Results

5.1 Stratified Training/Test Split

Here we report the results from experiments conducted using a stratified training/test split, where: 70% of the objects for each agent are included in the training set, while the remaining 30% of objects for each agent are used for the test set.

In Table 2, we present the results for the WikiartINFL dataset. CLOIR outperforms the baselines across all experiments. Both when considering visual features only, or in combination with text features, the most significant improvement over the baseline is observed in the model where positive examples are sampled based on similarity with a sample size of 100. For visual features only, the

	Sampling	Size	Feature	Model	P@10	P@10(2)	MRR	MRR(2)
Baseline	-	-	Image	ResNet	0.108	0.162	0.188	0.283
				CLIP	0.128	0.172	0.199	0.275
			Image-Text	ResNet+GPT2	0.104	0.148	0.181	0.27
				CLIP	0.132	0.148	0.199	0.299
CLOIR	Random	10	Image	ResNet	0.17	0.217	0.238	0.339
				CLIP	0.169	0.203	0.215	0.198
			Image-Text	ResNet+GPT2	0.306	0.323	0.302	0.388
				CLIP	0.266	0.28	0.291	0.333
CLOIR	Similarity	10	Image	ResNet	0.125	0.181	0.217	0.315
				CLIP	0.133	0.175	0.215	0.288
			Image-Text	ResNet+GPT2	0.125	0.165	0.201	0.284
				CLIP	0.299	0.254	0.282	0.352
CLOIR	Random	100	Image	ResNet	0.193	0.238	0.25	0.351
				CLIP	0.18	0.212	0.236	0.31
			Image-Text	ResNet+GPT2	0.391	0.406	0.38	0.434
				CLIP	0.308	0.318	0.321	0.353
CLOIR	Similarity	100	Image	ResNet	0.201	0.255	0.276	0.371
				CLIP	0.165	0.199	0.24	0.303
			Image-Text	ResNet+GPT2	0.417	0.442	0.419	0.513
				CLIP	0.385	0.402	0.408	0.447

Table 2. Results for the WikiartINFL dataset, stratified split. In green the highest values for the metrics. P@10 refers to Precision at 10 and MRR to Mean Reciprocal Rank, and their version with (2) represent the metric considering the second degree of chain of influence.

	Sampling	Size	Model	P@10	P@10(2)	MRR	MRR(2)
Baseline	-	-	ResNet	0.083	0.093	0.218	0.24
			CLIP	0.068	0.077	0.224	0.247
CLOIR	Random	10	ResNet	0.094	0.108	0.258	0.285
			CLIP	0.072	0.984	0.25	0.273
CLOIR	Similarity	10	ResNet	0.072	0.082	0.232	0.255
			CLIP	0.051	0.061	0.226	0.249
CLOIR	Random	100	ResNet	0.126	0.144	0.273	0.304
			CLIP	0.086	0.1	0.251	0.278
CLOIR	Similarity	100	ResNet	0.129	0.146	0.294	0.324
			CLIP	0.085	0.099	0.261	0.288

Table 3. Results for the iDesignerINFL dataset, stratified split. In green the highest values for the metrics.

improvement to the baseline reaches 10% across all metrics, which reaches up to 30% when considering text features too. Across metrics, the pre-trained model setup that leads to better performance is the combination of ResNet+GPT2. Furthermore, it is interesting to observe that sampling based on similarity leads to better performance when 100 examples per anchor are used, this trend does not hold when only 10 examples are considered. This result indicates that with fewer examples, greater variability ensures better generalization of the model. Conversely, when more examples are available, the increased sample size introduces variability, and a similarity-based sampling constraint helps the model better generalize the concept of influence.

A similar pattern can be observed in the experiments performed on the iDesignerINFL dataset, whose results are reported in Table 3. Namely, with

a sample size of 10, random sampling performs better than similarity sampling - which in this case performs even worse than the baseline - but this trend reverses when 100 examples are considered, and the improvement is between 5% and 10% across metrics. The main conclusion from these experiments is that with a smaller sample size, introducing more variability through random sampling leads to better performance in OIR. Conversely, with a larger sample size, similarity-based sampling yields better results.

5.2 Leave-out-agents Training/Test Split

	Sampling	Size	Feature	Model	P@10	P@10(2)	MRR	MRR(2)
Baseline	-	-	Image	ResNet	0.116	0.149	0.19	0.272
				CLIP	0.142	0.173	0.203	0.289
			Image-Text	ResNet+GPT2	0.114	0.143	0.188	0.27
				CLIP	0.163	0.187	0.213	0.294
CLOIR	Random	10	Image	ResNet	0.125	0.156	0.189	0.283
				CLIP	0.123	0.152	0.184	0.277
			Image-Text	ResNet+GPT2	0.201	0.226	0.197	0.314
				CLIP	0.163	0.182	0.192	0.283
CLOIR	Similarity	10	Image	ResNet	0.12	0.155	0.19	0.287
				CLIP	0.117	0.146	0.188	0.269
			Image-Text	ResNet+GPT2	0.15	0.183	0.218	0.318
				CLIP	0.153	0.18	0.211	0.289
CLOIR	Random	100	Image	ResNet	0.125	0.157	0.183	0.275
				CLIP	0.13	0.159	0.178	0.275
			Image-Text	ResNet+GPT2	0.178	0.2	0.197	0.301
				CLIP	0.165	0.187	0.185	0.281
CLOIR	Similarity	100	Image	ResNet	0.132	0.166	0.202	0.289
				CLIP	0.111	0.139	0.171	0.253
			Image-Text	ResNet+GPT2	0.191	0.217	0.21	0.326
				CLIP	0.15	0.169	0.187	0.252

Table 4. Results for the WikiartINFL dataset, leave-out-agents split. In green the highest values for the metrics.

Here we report the results of experiments conducted using the leave-out-agents training/test split. In this setup, objects from 70% of the agents are included in the training set, while the remaining 30% of agents are left out for testing. This experimental design allows us to further study the generalization abilities of our approach, as it must suggest influences for objects of agents not seen during training.

In Table 4, the results for the WikiartINFL dataset are reported. We can observe that with this data split, CLIP leads to a decrease in performance with respect to the baseline. In general, the same pattern observed in the stratified data split can be observed: when only 10 examples per anchor are considered, random sampling leads to better performance, whilst similarity-based sampling is better when 100 examples are considered. In particular, the best-performing setup, leading to an improvement of up to 8% with respect to the baseline, can be observed with random sampling and sample size equal to 10.

	Sampling	Size	Model	P@10	P@10(2)	MRR	MRR(2)
Baseline	-	-	ResNet	0.083	0.086	0.253	0.245
			CLIP	0.073	0.075	0.258	0.249
CLOIR	Random	10	ResNet	0.049	0.05	0.254	0.246
			CLIP	0.045	0.047	0.255	0.246
CLOIR	Similarity	10	ResNet	0.069	0.071	0.257	0.249
			CLIP	0.052	0.054	0.257	0.249
CLOIR	Random	100	ResNet	0.048	0.05	0.25	0.243
			CLIP	0.046	0.048	0.252	0.244
CLOIR	Similarity	100	ResNet	0.066	0.068	0.254	0.245
			CLIP	0.046	0.048	0.254	0.246

Table 5. Results for the iDesignerINFL dataset, leave-out-agents split. In green the highest values for the metrics.

For what concerns the iDesignerINFL dataset, the baseline performs similarly to CLOIR across experiments, as it can be observed in Table 5. This is a symptom that for this dataset the influence signal is not clear enough for CLOIR to capture and generalize over unseen agents.

5.3 Embedding space

The retrieval results show that our approach generally captures influence relations more effectively than the baseline. To further evaluate this, we compare the CLOIR embedding space with the baseline, quantifying their differences to demonstrate that our approach better supports object influence retrieval based on proximity. Additionally, we perform a qualitative analysis by visualizing the embedding space, focusing on a specific agent to illustrate how our approach captures influence through spatial proximity.

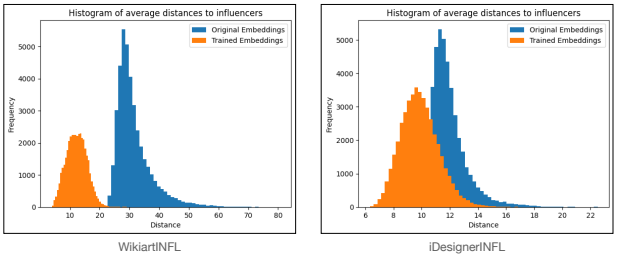

Fig. 2. Histogram of average distances between objects and their influencers. Comparison between baseline and CLOIR embedding space.

Quantitative embedding space evaluation To quantitatively assess the difference between baseline and CLOIR object representations in the embedding space, we calculate the average Euclidean distance between each agent's objects

and those of their respective influencers. This metric evaluates the effectiveness of our approach in capturing influence relations.

Figure 2 shows histograms of these average distances, indicating that CLOIR leads to a more compact embedding space where objects are closer to their respective influencers in both datasets.

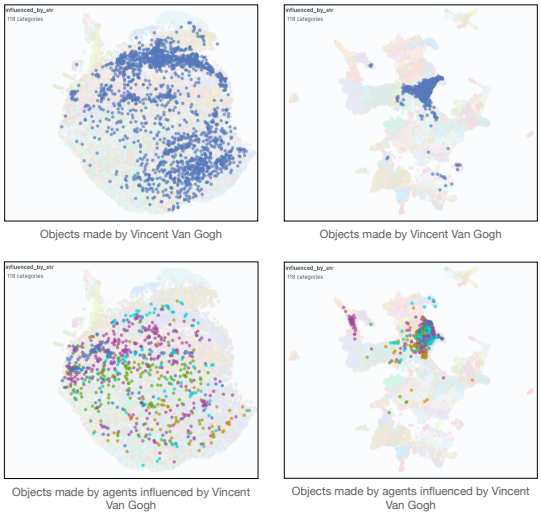

Fig. 3. WikiartINFL dataset embedding space: baseline on the left and CLOIR on the right. Top row: objects made by Vincent Van Gogh. Bottom row: objects made by agents influenced by Vincent Van Gogh.

Qualitative embedding space evaluation We qualitatively analyze the embedding space by visualizing it using the Uniform Manifold Approximation and Projection (UMAP) technique, which reduces dimensionality and computes Euclidean distances between objects. Our goal with CLOIR is to create an embedding space where object proximity indicates both similarity and influence.

To demonstrate this, we highlight objects from a specific agent and those influenced by them. In the WikiartINFL dataset, we use Vincent Van Gogh as an example (Figure 3), and in the iDesigner dataset, we use Alexander McQueen (Figure 4).

The visualizations show that in the CLOIR space, those two clusters overlap, whilst in the baseline embedding space, they do not.

5.4 Example of influence-based retrieval

Lastly, we present example retrievals obtained using the baseline model and CLOIR. For each query object, we retrieve the 10 closest objects in both embed-

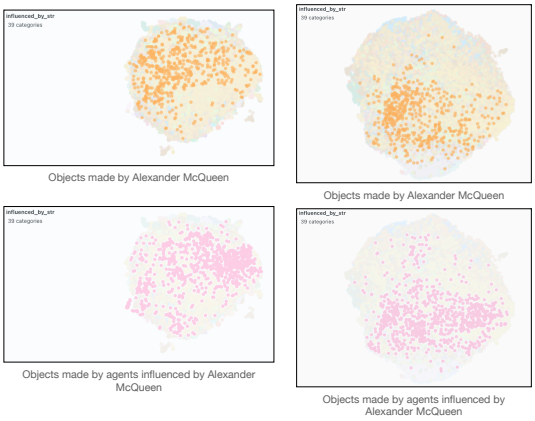

Fig. 4. iDesignerINFL dataset embedding space: baseline on the left and CLOIR on the right. Top row: objects made by Alexander McQueen. Bottom row: objects made by agents influenced by Alexander McQueen.

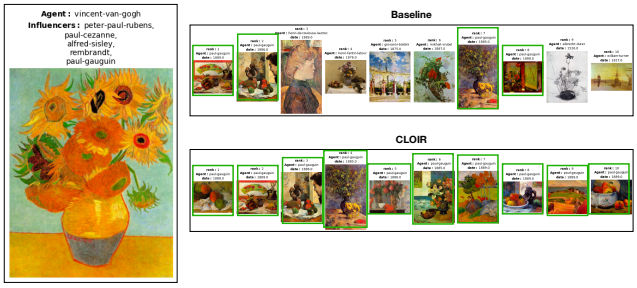

Fig. 5. WikiartINFL: Object Influence Retrieval comparing baseline with CLOIR retrieval.

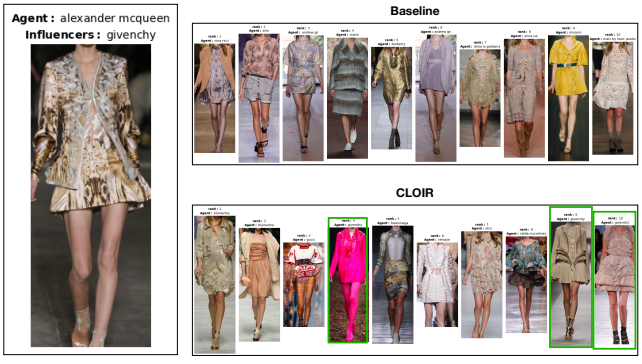

Fig. 6. iDesignerINFL: Object Influence Retrieval comparing baseline with CLOIR retrieval.

ding spaces. Figures 5 and 6 show retrievals for the WikiartINFL and iDesigner-INFL datasets, respectively. Objects created by influential agents are highlighted in green. We observe that CLOIR retrieves objects that not only share more characteristics with the query but also include a higher proportion of objects created by influential agents compared to the baseline.

6 Discussion

Our study demonstrates that CLOIR consistently outperforms traditional similarity-based methods in identifying potential influences between objects, establishing a foundation for performing Object Influence Retrieval (OIR).

While our results are promising, we acknowledge several limitations that also present opportunities for future research. A significant challenge is the lack of ground truth data on object influences to train and evaluate our approach robustly. To address this, we propose datasets incorporating agent influence as a proxy for object influence. However, this workaround, while innovative, would benefit from further ground truth data focused on object influence.

Data could be improved in other ways as well. Specifically, while the iDesigner dataset provides a collection of fashion item images and their creators, it lacks additional textual metadata and time information. Moreover, both datasets are unbalanced regarding the number of objects per agent. Future studies could thus focus on creating a new dataset that includes objects, corresponding agents, textual metadata (including time information), and an annotated test set with object influences.

Lastly, future research could explore other ways to integrate agent-influence information into object representation, leverage external domain knowledge to enhance representations and structure it in the form of knowledge graphs using neuro-symbolic approaches.

7 Conclusion

In this paper, we proposed a new task, Object Influence Retrieval (OIR), and CLOIR, an approach to solve it, by combining object similarity with influence knowledge between agents with a contrastive learning approach. We demonstrated the efficacy of CLOIR on two datasets, WikiartINFL and iDesignerINFL, achieving up to 30% improvement over a similarity-based baseline. These results suggest that combining similarity-based information with contextual influence knowledge can enhance the retrieval of objects that are influential for a query object. This work opens up new avenues for research in the automatic retrieval of influence, offering a framework that can be extended and refined in future studies. Research in this area has the potential to enable new applications in areas such as influence-based search engines, recommendation systems, historical media analysis, and the study of innovation dynamics. We hope that this work will pave the way for further studies in this direction, contributing to a deeper understanding of influence and innovation in various domains.

References

1. D. Park, J. Nam, and J. Park, "Novelty and influence of creative works, and quantifying patterns of advances based on probabilistic references networks," *EPJ Data Science*, vol. 9, 12 2020.
2. P. B. Paulus and M. Dzindolet, "Social influence, creativity and innovation," *Social Influence*, vol. 3, no. 4, pp. 228–247, 2008.
3. G. Hermeren, *Influence in Art and Literature*, ser. Princeton Legacy Library. Princeton University Press, 2015. [Online]. Available: https://books.google.nl/books?id=SXt9BgAAQBAJ
4. Y. Yoo, R. J. Boland Jr, K. Lyytinen, and A. Majchrzak, "Organizing for innovation in the digitized world," *Organization science*, vol. 23, no. 5, pp. 1398–1408, 2012.
5. F. Alfieri, L. Asprino, N. Lazzari, and V. Presutti, "Creative influence prediction using graph theory." in *CREAI@ AI* IA*, 2023, pp. 1–15.
6. B. Saleh, K. Abe, R. S. Arora, and A. Elgammal, "Toward automated discovery of artistic influence," *Multimedia Tools and Applications*, vol. 75, pp. 3565–3591, 4 2016. [Online]. Available: https://link.springer.com/article/10.1007/s11042-014-2193-x
7. L. Shamir and J. A. Tarakhovsky, "Computer analysis of art," *J. Comput. Cult. Herit.*, vol. 5, no. 2, aug 2012. [Online]. Available: https://doi.org/10.1145/2307723.2307726
8. B. Saleh, K. Abe, and A. M. Elgammal, "Knowledge discovery of artistic influences: A metric learning approach." in *ICCC*, 2014, pp. 163–172.
9. B. Dalmoro, C. Monteiro, and S. R. Musse, "Measuring the influence of painters through artwork facial features," in *2022 35th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, vol. 1, 2022, pp. 37–42.
10. A. Morales Tirado, J. Carvalho, M. Ratta, C. Uwasomba, P. Mulholland, H. Barlow, T. Herbert, and E. Daga, "Musical meetups knowledge graph (mmkg): a collection of evidence for historical social network analysis," in *European Semantic Web Conference*. Springer, 2024, pp. 110–127.
11. N. J. Bryan and G. Wang, "Musical influence network analysis and rank of sample-based music." in *ISMIR*, 2011, pp. 329–334.
12. M. Kopel, "Analyzing music metadata on artist influence," in *Intelligent Information and Database Systems: 7th Asian Conference, ACIIDS 2015, Bali, Indonesia, March 23-25, 2015, Proceedings, Part I 7*. Springer, 2015, pp. 56–65.
13. S. R. Dubey, "A decade survey of content based image retrieval using deep learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2687–2704, 2021.
14. Y. Gu, H. Zhang, Z. Zhang, and Q. Ye, "Unsupervised deep triplet hashing with pseudo triplets for scalable image retrieval," *Multimedia Tools and Applications*, vol. 79, no. 47, pp. 35 253–35 274, 2020.
15. S. Huang, Y. Xiong, Y. Zhang, and J. Wang, "Unsupervised triplet hashing for fast image retrieval," in *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*, 2017, pp. 84–92.
16. M. Banerjee, B. M. Cole, and P. Ingram, ""distinctive from what? and for whom?" deep learning-based product distinctiveness, social structure, and third-party certifications," *Academy of Management Journal*, vol. 66, pp. 1016–1041, 8 2023. [Online]. Available: https://journals.aom.org/doi/abs/10.5465/amj.2021.0175
17. E. Cetinic, T. Lipic, and S. Grgic, "Fine-tuning convolutional neural networks for fine art classification," *Expert Systems with Applications*, vol. 114, pp. 107–118, 12 2018.

18. A. Efthymiou, S. Rudinac, M. Kackovic, M. Worring, and N. Wijnberg, "Graph neural networks for knowledge enhanced visual representation of paintings," *MM 2021 - Proceedings of the 29th ACM International Conference on Multimedia*, pp. 3710–3719, 5 2021. [Online]. Available: https://arxiv.org/abs/2105.08190v1
19. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
20. A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
21. A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.
22. M. A. Carbonneau, V. Cheplygina, E. Granger, and G. Gagnon, "Multiple instance learning: A survey of problem characteristics and applications," *Pattern Recognition*, vol. 77, pp. 329–353, 5 2018.
23. D. Kotzias, M. Denil, N. D. Freitas, and P. Smyth, "From group to individual labels using deep features," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, vol. 2015-August, pp. 597–606, 8 2015. [Online]. Available: https://dl.acm.org/doi/10.1145/2783258.2783380
24. X. Zhang, F. Zhou, Y. Lin, and S. Zhang, "Embedding label structures for fine-grained feature representation," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, pp. 1114–1123, 12 2015. [Online]. Available: https://arxiv.org/abs/1512.02895v2
25. M. W. Gondal, S. Joshi, N. Rahaman, S. Bauer, M. Wuthrich, and B. Schölkopf, "Function contrastive learning of transferable meta-representations," in *International Conference on Machine Learning*. PMLR, 2021, pp. 3755–3765.
26. M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "The faiss library," 2024.
27. W. R. Tan, C. S. Chan, H. Aguirre, and K. Tanaka, "Improved artgan for conditional synthesis of natural image and artwork," *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 394–409, 2019. [Online]. Available: https://doi.org/10.1109/TIP.2018.2866698