Semi-Supervised Multimodal Representation Learning Through a Global Workspace

Benjamin Devillers, Léopold Maytié, and Rufin VanRullen

***Abstract*— Recent deep learning models can efficiently combine inputs from different modalities (e.g., images and text) and learn to align their latent representations or to translate signals from one domain to another (as in image captioning or text-to-image generation). However, current approaches mainly rely on brute-force supervised training over large multimodal datasets. In contrast, humans (and other animals) can learn useful multimodal representations from only sparse experience with matched cross-modal data. Here, we evaluate the capabilities of a neural network architecture inspired by the cognitive notion of a "global workspace" (GW): a shared representation for two (or more) input modalities. Each modality is processed by a specialized system (pretrained on unimodal data and subsequently frozen). The corresponding latent representations are then encoded to and decoded from a single shared workspace. Importantly, this architecture is amenable to self-supervised training via cycle-consistency: encoding–decoding sequences should approximate the identity function. For various pairings of vision-language modalities and across two datasets of varying complexity, we show that such an architecture can be trained to align and translate between two modalities with very little need for matched data (from four to seven times less than a fully supervised approach). The GW representation can be used advantageously for downstream classification and cross-modal retrieval tasks and for robust transfer learning. Ablation studies reveal that both the shared workspace and the self-supervised cycle-consistency training are critical to the system's performance.**

***Index Terms*— Cycle-consistency, global workspace (GW) theory, multimodal learning, semi-supervised learning.**

I. Introduction

HUMANS learn about the world from various sources: images when looking around, language describing objects and their properties, sounds from the environment or from conversations, and so on. These diverse inputs come together, sometimes asynchronously, to build a joint representation of the external world. Thanks to this multimodal convergence, human language is *grounded* in the sensory environment, and conversely, sensory perception is semantically *grounded* by its relation to language [1].

Recent works have shown the importance of training deep learning models using several modalities such as vision and language. Paired inputs across modalities can be leveraged as natural and readily available annotations [2]; they can be used as semantic constraints to train zero-shot learning models [3], [4] and more capable and general systems [5] or to infuse each modality with additional semantic knowledge (i.e., multimodal grounding) [6], [7], [8]. The performance of these models still heavily depends on the availability of large-scale paired multimodal datasets (400 million image–caption pairs used in contrastive language-image pretraining (CLIP) [2], more than 4 billion pairs for contrastive captioners (CoCa) [9] or DALL·E 2 [10]). This trend of brute-force supervised training on ever larger datasets has led to an impressive boost in the models' performance and to the emergence of new abilities in recent large AI models [11]. However, it departs from the more frugal learning strategies adopted by the human brain. In addition, recent studies have shown that current multimodal networks sometimes fail to improve upon unimodal networks, i.e., they do not always achieve proper multimodal grounding [12], [13].

Here, we explore the capabilities of a multimodal system taking inspiration from the cognitive science theory of the global workspace (GW) [14], [15]. GW theory explains how different modalities in the human brain are integrated into a common shared representation, subsequently redistributed or *broadcast* among the specialized unimodal modules (see Section II-B). We define essential properties that multimodal networks should verify (see Section III). In particular, translating signals between modalities and aligning representations across modalities are important and complementary abilities that should be jointly optimized. Yet, most recent multimodal systems used either contrastive learning (e.g., CLIP [2] and temporal shift module (TSM) [16]), which forces alignment of the modalities without preserving modality-specific information or translation objectives (e.g., visual representations from textual annotations (VirTex) [17] and image-conditioned masked language modeling (ICMLM) [18]), which do not provide a joint multimodal space for downstream tasks. Recently, the CoCa model [9] confirmed that using both translation and contrastive objectives for supervision could lead to significantly better performance. We show that our proposed GW-inspired architecture combines these two desired properties. Furthermore, to reduce annotations and encourage more

Manuscript received 19 February 2024; accepted 10 June 2024. This work was supported by the High Performance Computing (HPC) Resources from CALMIP under Grant 2020-p20032. The work of Léopold Maytié was supported in part by the Agence Nationale de la Recherche (ANR) grant Natural Language Programming for Conversational Cobots (COCOBOT) under Grant ANR-21-FAI2-0005, in part by the Region Occitanie grant Cobots Conversationnels pour Processus Industriels et Logistiques (COCOPIL), and in part by the "Défi-clé Robotique centrée sur l'humain" Ph.D.-thesis grant. The work of Rufin VanRullen was supported in part by the Artificial and Natural Intelligence Toulouse Institute (ANITI) Chair (ANR grant) under Grant ANR-19-PI3A-004 and in part by the European Research Council (ERC) Advanced Grant Global Latent Workspace (GLoW) under Grant 101096017. *(Corresponding author: Rufin VanRullen.)*

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

The authors are with the CerCo, CNRS UMR 5549, Université de Toulouse, 31013 Toulouse, France, and Artificial and Natural Intelligence Toulouse Institute (ANITI), 31400 Toulouse, France (e-mail: rufin.vanrullen@cnrs.fr).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNNLS.2024.3416701, provided by the authors.

Digital Object Identifier 10.1109/TNNLS.2024.3416701

frugal (and, thus, more human-like) learning, we also advocate for a semi-supervised learning setting by adding unsupervised cycle-consistency objectives to the model. In short, we propose and evaluate a neural network architecture combining a GW with semi-supervision, in a bimodal setting where only few paired examples are available.

II. Related Work

As explained above, multimodal representation learning for neural networks is a vast and fast-growing research area, whose exhaustive coverage would require an extensive review well beyond the scope of this article. However, two specific features of our proposed model deserve a more in-depth treatment: unsupervised training via cycle-consistency and the GW theory.

A. Cycle-Consistency

The idea of using back-translations to synchronize two latent spaces has been introduced previously. Kalal et al. [19] presented a forward–backward error to solve a visual point-tracking task. It consisted of predicting a forward trajectory of the tracked point in an image sequence and then predicting a reverse trajectory, considering the reversed image sequence. The two trajectories were then compared together.

More recently, a similar cycle-consistency principle was applied in natural language processing (NLP) for unsupervised neural translation: language alignment is successful when the successive translation from language A to language B, and then, back-translation from B to A returns the original sentence [20], [21], [22], [23]. For instance, Lample et al. [23] used back-translations to optimize a sequence-to-sequence model with attention [24] so that it could translate between two languages without ever having access to aligned multilingual corpora during training. Their training objective combined cycle-consistency with an adversarial loss to force the generation in each domain to match the actual language distributions.

Based on this logic, cycle-consistency was also used to synchronize multiple visual domains, i.e., unsupervised image-to-image translation [25], [26], [27]. For instance, in CycleGAN [25], Zhu et al. trained two generative adversarial networks (GANs) to generate images in the style of one specific domain and then used a cycle-consistency loss to synchronize the latent spaces of the GANs so that an image from one domain (e.g., a horse) could be translated into the equivalent image in the other domain (e.g., a zebra).

From this point, it was not long until the technique was applied to multimodal use cases, such as text-to-image [28], [29], [30] or touch-to-image translation [31]. For instance, Pham et al. [8] applied cycle-consistency training to a multimodal (image, text, and sound) sentiment analysis task. They showed that back-translations produced robust representations, and the model could deal with missing modalities during inference. They used a hierarchical architecture, where two modalities are first aligned, and then, the third modality is aligned with the common latent space of the first two. This architecture is asymmetric and, thus, requires favoring some modalities over others. Upon trying different combinations, their best performance was obtained when first learning to translate between vision and text and then including audio.

Overall, it is clear that unsupervised learning via cycle-consistency is a powerful method to train multimodal systems. However, the technique is typically applied to multimodal translation tasks *or* alignment tasks, but rarely to both; and it has never been combined with the GW architecture.

B. Global Workspace

How the brain combines information from multiple modalities into a unified representation that can be flexibly reused for a wide array of tasks is still the subject of active research. One prominent conjecture, however, is Baar's GW theory [14], [32], extended into a neuronal framework by Dehaene et al. [15]. The theory comprises several components: a number of "specialist" modules, each independently processing one modality (visual stream, auditory stream, memory, motor, and so on); an attention mechanism that determines the relevant specialist modules at each moment in time, based on both exogenous (saliency) and endogenous aspects (task, prior state); and a shared space with fixed capacity, the "GW" itself. Because of its fixed capacity, all modules cannot simultaneously access the workspace; this is why they must compete against each other through the attention mechanism. The winning modules transmit their information to the GW. Finally, the workspace representation is automatically broadcast to all modules. According to the theory, it is this broadcast of information that represents our inner experience, enabling multimodal grounding and flexible use for downstream tasks (including decision and action planning). To illustrate his theory, Baars makes an analogy with a theater, where specialist modules are simultaneously the actors on a stage and the audience. While they are "on stage" (i.e., mobilized in the shared workspace), they can broadcast information to all other modules.

A recent opinion paper [33] proposed that current working AI principles could already be used to implement this theory and provided a step-by-step roadmap for this implementation. Moreover, other implementations have recently been put forward. For instance, Juliani et al. [34] highlighted that the Perceiver architecture recently proposed by Jaegle et al. [35] could be used as a GW. Goyal et al. [36] introduced an architecture for sharing information between modules (e.g., transformer layers) that they explicitly labeled as a GW implementation.

The present work is not intended to compete with these prior systems, nor does it contend to offer a full implementation of Baar's theory. Instead, we make use of some prominent features of the GW theory (a unique, limited-capacity multimodal representation that can be broadcast and reused for other tasks) while leaving others aside for future work. In particular, as we chose to work here with only two modalities, there is no need for us to consider attentional competition between modules (as only one domain can occupy the GW at each moment) though this aspect will be important to examine in future studies.

C. Cross-Modal Retrieval

Multimodal representations can be used for various applications. One of them is cross-modal retrieval, where the goal is

to retrieve samples in one domain using a related query from another domain. The prevalent form is image–text retrieval, which consists of either retrieving the caption of an image or the image that matches a specific description.

Hu et al. [37] use a multimodal space that can be linearly projected (with a fixed matrix P) into the vectors of the classes pictured in the image. They use a combination of cycle-consistency and multiclass supervision. Alignment of modalities is achieved via multiclass supervision. This method is, thus, particularly well suited for a cross-modal model that ranks examples solely based on common classes. Tian et al. [38] use a combination of coupled discrete variational autoencoders (DVAEs) and a fusion-exchange variational autoencoder (VAE). They leverage disentanglement learning to extract modality-specific and modality-invariant information. They also introduce a fusion-exchange VAE to improve the alignment of modality-invariant features. Finally, they introduce the counter-intuitive cross-reconstruction strategy (CICR) where they learn to reconstruct the information of one modality with the decoder of the other modality. The model achieved state of the art (SOTA) results on image–text retrieval benchmarks.

We explore the performance of a GW-like architecture in a cross-modal retrieval task in Section VII. As opposed to the aforementioned methods, we focus on a setting with only a limited amount of paired image/text pairs.

Zhen et al. [39] also trained their model with limited pairs of image/text. They tackle the issue that available paired samples could have different labels than additional unpaired samples. To address this, they use a discrimination loss with real labels whenever labels are available, and they use pseudolabels (iteratively refined during training) as targets for unpaired data. They also use Kullback–Leibler (KL) divergence to align feature representations of the image and text encoders, as done in other works [40], [41]. Our model differs by the use of cycle-consistency and contrastive loss (similar to CLIP) rather than a symmetric KL-divergence. Another crucial difference is that we never use class-label information. Our models are trained only to learn a valid multimodal representation, i.e., one that supports vision-language alignment, translation, and cycle-consistency objectives.

III. Problem Statement

In this study, we will focus on a bimodal network receiving inputs from two modalities, such as vision (images) and text (captions). Fig. 1(a) shows a diagram of a generic bimodal network, where we define visual and text encoders (e_v and e_t) and their respective decoders (d_v and d_t).

$\{x_i\}$ represents a set of images, and $\{y_i\}$ represents their matched captions for $i \in \mathcal{S}$ a supervised set of matched examples. In the most general setting, our datasets could also include unmatched data samples, so we additionally define the supersets $\mathcal{U}_v$ and $\mathcal{U}_t$ for images and captions, respectively, such that $\mathcal{S} = \mathcal{U}_v \cap \mathcal{U}_t$. Images in $\mathcal{U}_v \backslash \mathcal{S}$ do not have a matched caption in the dataset and similarly for captions in $\mathcal{U}_t \backslash \mathcal{S}$.

Multimodal networks can express different desirable properties. We define here two primary and two secondary properties that we believe represent fundamental behaviors that multimodal networks should possess. The first primary property is

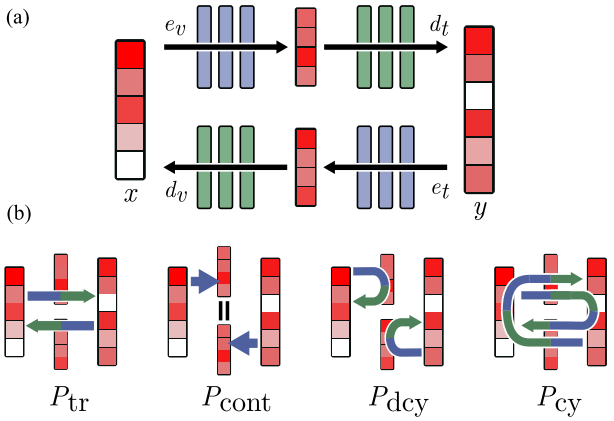

Fig. 1. (a) Generic bimodal network. Inputs can be from two modalities x and y (for instance, visual images, and text captions). e_v and e_t are feed-forward neural networks that project each modality into a latent space; d_v and d_t are decoders that decode the latent space into the respective modality (note that this generic model aims at clarifying our definitions; it does not yet correspond to our GW architecture). (b) Illustration of the primary and secondary desirable properties for multimodal systems. Each arrow shows a learned path to convert one latent vector into another. For instance, in P_{dcy}, we can convert from one domain to itself via the central representation. Note that the four properties are not independent but can be causally related, as we describe in relations R_1–R_4.

translation, whereby it should be possible to translate each modality into the other one

$$\begin{cases} d_t(e_v(x_i)) = y_i \\ d_v(e_t(y_i)) = x_i \end{cases}, \quad \forall i \in \mathcal{S}. \tag{P_{tr}}$$

The second primary property is the contrastive encoding property, which forces the alignment of latent spaces for matched data

$$\begin{cases} e_v(x_i) = e_t(y_j), & \forall i = j \in \mathcal{S} \\ e_v(x_i) \neq e_t(y_j), & i \neq j. \end{cases} \tag{P_{cont}}$$

As emphasized in the Introduction, these two properties correspond to two prominent objectives for multimodal learning in the literature; for example, text-to-image generation systems (such as DALL·E 2 [10] or Stable Diffusion [42]) and image captioning models (such as CoCa [9]) are based on translation objectives, while CLIP [2] relies on contrastive learning for vision-language representation alignment.

Although less often encountered in the literature, we believe that a bimodal network could also benefit from two additional "secondary" properties. We define the demi-cycle-consistency, where $\forall (i, j) \in \mathcal{U}_v \times \mathcal{U}_t$

$$\begin{cases} d_v(e_v(x_i)) = x_i \\ d_t(e_t(y_j)) = y_j \end{cases} \tag{P_{dcy}}$$

and the (full) cycle-consistency property, where $\forall (i, j) \in \mathcal{U}_v \times \mathcal{U}_t$

$$\begin{cases} d_v(e_t(d_t(e_v(x_i)))) = x_i \\ d_t(e_v(d_v(e_t(y_j)))) = y_j \end{cases}. \tag{P_{cy}}$$

These two properties can be seen as unsupervised versions of the primary properties, inspired by the unsupervised language translation literature [23], [43]. Importantly, they are defined independently of the set $\mathcal{S}$ of paired samples and

remain valid even if $\mathcal{S}$ is an empty set (i.e., unsupervised learning). Fig. 1(b) provides an illustration of these four properties.

Note that our primary and secondary properties are not independent, and we can, in fact, highlight four relations between them that should be true for any matched samples in $\mathcal{S}$.

R_1: $P_{\text{tr}} \Rightarrow P_{\text{cy}}$.
R_2: P_{tr} & $P_{\text{cont}} \Rightarrow P_{\text{dcy}}$.
R_3: If d_v or d_t injective, P_{tr} & $P_{\text{dcy}} \Rightarrow P_{\text{cont}}$.[1]
R_4: P_{cont} & $P_{\text{dcy}} \Rightarrow P_{\text{tr}}$.

The first two relations show that if we restrict ourselves to the set of matched samples, the secondary properties are automatically obtained if the primary properties are verified. Furthermore, the last two relations show that each primary property can follow from the other primary property combined with a secondary one (with some additional constraint on injectivity for R_3). Thus, we hypothesize that optimizing *all four* properties in a single network could prove advantageous since they will tend to reinforce each other as per relations R_1–R_4. Importantly, this means that the secondary properties could be used in an *unsupervised* way to take advantage of unpaired data and still enhance the network's primary properties.

How do the four properties relate to the GW architecture defined previously? Within the scope of our study (i.e., for bimodal systems where attentional competition between modules is not required), a GW architecture must verify two criteria. First, it must have a common shared latent space across modalities, where a given input produces the same representation regardless of its modality of presentation; this corresponds exactly to property P_{cont}. Furthermore, the *broadcast* aspect of GW theory implies that this shared space should be able to inform other modalities; thus, P_{cont} is necessary but not sufficient, as it only constrains the encoders e_v and e_t but not the corresponding decoders. Put another way, a model trained only for representation alignment, such as CLIP [2], cannot be considered to implement a GW. To train the decoders and permit broadcast, at least one of the other properties must also be optimized. In summary, a bimodal network can be said to include a GW if it verifies P_{cont} and at least one additional property in $\{P_{\text{tr}}, P_{\text{dcy}}, P_{\text{cy}}\}$.

Given these considerations and our initial goal to study both the usefulness of a GW and the need for supervision in bimodal representation learning, we chose to focus our comparisons on four main models, as shown in the first section of Table I. The models differ by the properties that they are designed to optimize such that two of them rely only on supervised training with paired bimodal samples, while the other two can also take advantage of additional unpaired data (semi-supervised training). Furthermore, two of them do not satisfy the criteria for a GW, while the other two do. Indeed, a model designed solely to optimize translation (P_{tr} and possibly its cycle-consistent version P_{cy}) can be thought of as operating with two entirely independent latent spaces, as illustrated in Fig. 1(a) (middle). It is only when imposing the representation alignment property P_{cont} (either directly or indirectly via relation R_3) that the two latent spaces can be considered to work jointly as a unique shared space—the GW.

[1] $d_t(e_v(x_i)) \overset{P_{\text{tr}}}{=} y_i \overset{P_{\text{dcy}}}{=} d_t(e_t(y_i))$. Then, using the injectivity of d_t, $e_v(x_i) = e_t(y_i)$. Similarly, if d_v is injective, start with $d_v(e_t(y_i)) = x_i = d_v(e_v(x_i))$. Then using the injectivity of d_v, $e_t(y_i) = e_v(x_i)$.

TABLE I
COMPARED MODELS AND THEIR PROPERTIES. ALL MODELS SHARE THE SAME ARCHITECTURE AND DIFFER ONLY IN TERMS OF THE PRIMARY AND/OR SECONDARY PROPERTIES THAT THEY ARE DESIGNED TO OPTIMIZE. THE SECOND COLUMN INDICATES WHETHER EACH MODEL RELIES ON SEMI-SUPERVISED TRAINING; THE THIRD COLUMN INDICATES WHETHER THE PROPERTIES ENFORCE THE EMERGENCE OF A GW, I.E., A COMBINATION OF AN ALIGNED MULTIMODAL REPRESENTATION SPACE AND THE ABILITY TO BROADCAST THE MULTIMODAL REPRESENTATION BACK TO EACH MODALITY. WE MAINLY FOCUS ON THE FIRST FOUR MODELS IN OUR EXPERIMENTS. IN ADDITION, P_{cont} OFFERS ALIGNMENT BUT NOT BROADCAST. P_{tr} & P_{dcy} RELIES PARTLY ON SEMI-SUPERVISED LEARNING AND COULD MEET SOME REQUIREMENTS FOR A GW (ACCORDING TO RELATION R_3). P_{tr} & P_{cy} TAKES BETTER ADVANTAGE OF SEMI-SUPERVISION, AS THE FULL CYCLE LOSS OPTIMIZES MORE PARAMETERS THAN DEMI-CYCLES. BY COMBINING ALL THE LOSSES, P_{tr} & P_{cont} & P_{dcy} & P_{cy} CAN SIMULTANEOUSLY ENFORCE ALL OF THE REQUIRED PROPERTIES AND IS, THUS, OUR PROPOSED "TARGET" MODEL

Model Properties	Semi-supervision	"Global Workspace" (Alignment + Broadcast)
P_{tr}	–	–
P_{tr} & P_{cont}	–	++
P_{tr} & P_{cy}	++	–
P_{tr} & P_{cont} & P_{dcy} & P_{cy}	+++	+++
P_{cont}	–	± (no broadcast)
P_{tr} & P_{dcy}	+	+

With this model selection (first section of Table I), we can systematically investigate our two factors of interest: GW and semi-supervision. To train the models, we use the corresponding properties listed in Table I as our optimization targets. For evaluation, we measure the two primary desired properties (translation loss and contrastive loss) on a separate test set. For completeness, we also evaluate the secondary properties of each model after training. Finally, we check how the models perform on some downstream tasks.

IV. DATASETS

A. Simple Shapes

To start with, we designed a multimodal dataset called "Simple Shapes." The *Simple Shapes* dataset is reminiscent of the 2-D shapes dataset of [44] but is extended with more varying attributes. This dataset fits several objectives: first, we want an automated generation procedure to obtain as many samples as needed. It also allows us to control the number of annotations (i.e., matched samples) in the dataset. Second, we want the modalities to overlap by representing the same content so that we can train translation and alignment models between the modalities. Third, we want the models' architecture (and correspondingly, the data distribution) to be

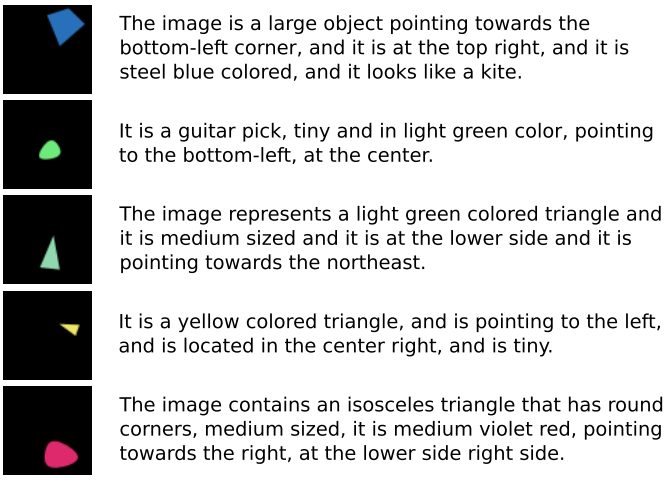

Fig. 2. Examples from the *Simple Shapes* dataset. Each image contains a unique object of differing shape, color, rotation, size, and position. The image is paired with a natural language sentence describing the attributes.

Fig. 3. Examples from the *Factory* dataset. Each image is taken from a fixed point of view; a table is randomly positioned in the environment, while the other objects (robots, cones, crates, conveyer belts, and so on) remain in a fixed position. Each image can be associated with a "proto-language" description of the table's attributes (position, orientation, and color) or with a natural language English description (as shown on the right).

relatively simple, so as to iterate quickly over several model training regimes for our analysis.

We consider two modalities: vision and language.

1) Visual Modality: For the visual domain, we create small images of 32×32 pixels with a black background and a unique object fully visible in the frame. Fig. 2 shows examples of the visual domain. The object can be of three categories: an egg-like shape, an isosceles triangle, and a diamond. These categories were chosen so that the shape's orientation can be defined unambiguously. Each object has several attributes that are sampled uniformly: a size $s \in [s_{\min}, s_{\max}]$, a location $(x, y) \in [(s_{\max}/2), 32 - (s_{\max}/2)[^2$ (we add a margin of size $s_{\max}/2$ so that images of all sizes are completely in frame), a rotation $r \in [0, 2\pi[$, and a hue, saturation, lightness (HSL) color $(c_h, c_s, c_l) \in [0, 1]^2 \times [l_{\min}, 1]$ and then translated into RGB (this ensures that images can always be seen on a black background by setting a minimum lightness value).

2) Language Modality:

a) Proto-language: First, we use a form of "proto-language," defined by the attributes and categories that were used to produce the images. It contains a 3-D one-hot annotation for the class, two numerical values for the position (x, y), one for the size, and three for the colors (in RGB), and we transform the angle value of the rotation into two values for its sine and cosine. We found that describing the angle in the cos/sin space yields better results, as it avoids the wrap-around discontinuity around 0 or 2π, which can lead to a significant error signal when using the mean square error (mse) loss. Besides, all the attributes are normalized to have a value between -1 and 1. This *proto-language* modality is useful because it guarantees an exact and unique match between the descriptions of any data sample in the two modalities.

b) Natural language: In addition, we implement a natural language modality to describe the visual aspects of the image in plain English. The text is automatically generated from the semantic (proto-language) vectors using a heuristic method described in Appendix C (see the Supplementary Material). Training with natural language adds complexity, as the attributes are quantized into words, meaning that we go from a continuous distribution to a categorical one. Moreover, natural language contains uncertainty since a single word can be used to capture a distribution instead of a specific range of attribute values: for instance, "small" and "medium" may be used to refer to the same object size depending on the person or context. Using natural language also affords more liberty in the description of the shapes, the structure of the sentence, or even the vocabulary. This makes multimodal translation and alignment inherently more difficult because different sentences can be used to describe the same object, and slightly different objects could be described by the exact same sentence. In other words, when using natural language, multimodal alignment is typically not bijective.

B. Factory

We create a second synthetic dataset called *Factory*, composed of $K = 200\,000$ image–text pairs. This dataset remains very close to the *Simple Shapes* dataset in principle but uses more realistic 128×128 pixel images from a simulated robotic environment (defined using the Webots simulator [45]). The scene viewpoint, the overall layout, and the position of most objects (robot, barrels, crates, cones, conveyer belt, and so on) are fixed across images; only a table varies, with randomly chosen attributes: position (x, y), orientation, and color hue. Fig. 3 (left) shows examples of images from the *Factory* dataset.

As in Simple Shapes, we can describe images using a "proto-language" (attribute vectors) or using natural language (English). The attributes describing the image in the proto-language contain two values (x, y) for the position of the table (normalized between -1 and 1). For the table orientation, the angle θ around the z-axis is transformed as $[\cos(2\theta), \sin(2\theta)]$ (angle multiplied by 2 because of the table's symmetry modulo π). The table's color only varies in the (circular) hue domain, so it is transformed as an angle: $[\cos(2\pi H), \sin(2\pi H)]$. As for the *Simple Shapes* dataset, we also generated natural language sentences describing each image, using a heuristic method based on the attributes (see the Appendix in the Supplementary Material). Examples of generated sentences are shown in Fig. 3 (right).

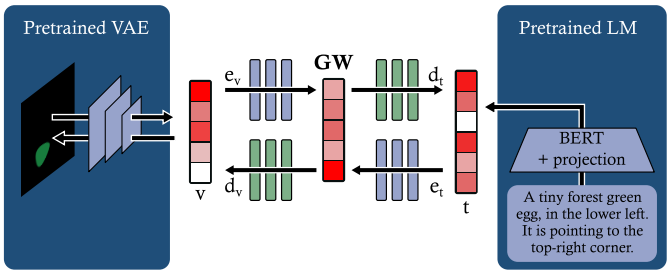

Fig. 4. Diagram of our GW architecture. Specialist modules for vision and language have a blue background. We use each modality's encoder (e_v or e_t) to project data samples into a common latent space (GW) and the corresponding decoders (d_t or d_v) to translate GW activations into the input domains. In this figure, we make the assumption that the model verifies property P_{cont}, thus having the GW representation shared across modalities.

C. COCO Captions

Finally, to replicate our findings on a dataset with naturalistic images and human-written captions, we will evaluate COCO Captions [46]. The dataset contains 82 783 training, 10 000 validation, and 30 504 test visual scenes with text descriptions. For each image, there are five captions describing the visual scene in natural language. We use the popular Karpathy split [47] with 5000 images for both validation and testing and use the remaining 30 504 images for additional training examples ("restval").

V. Model

All of our models are connected to two modalities: vision and language (proto-language or natural language). The corresponding specialist modules are pretrained independently on their modality and subsequently frozen when training the multimodal networks.

A. Specialist Modules

1) Visual Domain: For the visual specialist module, we use a β-VAE. The details of the architecture are provided in Appendix B (see the Supplementary Material). We chose a VAE [48] instead of a regular autoencoder to take advantage of the $\mathcal{N}(0, 1)$ normally distributed latent space, which may be helpful for multimodal alignment and translation. Moreover, we use a β-VAE [44] as opposed to a regular VAE to obtain a more regularized latent space. Indeed, the latent space of a β-VAE has more disentangled dimensions, which again could assist the translation and alignment between the modalities. We train the VAE on the visual domain data only, using a 12-D latent space, and use $\beta = 0.1$, a value chosen to optimize the reconstruction quality while keeping a normal latent vector to allow sampling.

For the Factory dataset, given that most of the background is identical across images, and only the table varies, a standard VAE architecture (with a pixel-based reconstruction loss) was found to be inappropriate, as it tended to favor the background at the expense of the table details. To encourage our vision module to properly encode the table, we used a learnable tensor image X_0 that was subtracted from the image before entering the VAE and added back to the VAE reconstruction. This way, the learnable tensor would capture fixed elements of the scene, while the VAE could focus on changing elements. Details of this architecture and examples of encoded and reconstructed images are given in Appendix B (see the Supplementary Material).

2) Proto-Language Domain: For the proto-language, we directly use a vector containing the concatenated attributes, normalized between -1 and 1.

3) Language Domain: For the text domain, we use a pretrained bidirectional encoder representations from transformers (BERTs) model to encode the natural language sentences into a latent vector. This high-dimensional vector (768-D) encodes extra information (e.g., syntactic or grammatical) that cannot be aligned with or translated into the visual domain. Thus, to simplify the task and maintain reasonably compact latent representations, we use another VAE (see architecture details in Appendix B in the Supplementary Material) to project the BERT vector into a smaller (12-D) latent representation; in addition to the standard VAE training objectives (reconstruction of the initial BERT vector, KL loss), we add an attribute- and grammar-prediction head (see architecture details in Appendix B in the Supplementary Material) and use its prediction loss in the optimization.

For Factory, the latent space dimensionality was increased from 12 to 20, and we added layers for the regression of attributes (see Appendix B in the Supplementary Material).

B. Objectives of the Multimodal System

We train a multimodal system structured around a GW, i.e., an intermediary space that allows unimodal latent spaces to communicate (see Fig. 4). To connect the modalities to the GW, we use one encoder e_m for each connected modality m. Each encoder projects the unimodal latent space into the workspace. Moreover, a decoder d_m translates the GW representations back to the domain's unimodal latent space. In our experiments, e_m and d_m are four-layer feedforward models with a 12-D input, 256-D hidden layers, and a 12-D output.

In Factory, we used four hidden layers of size 512 each for the encoders and decoders. The GW latent space itself had ten dimensions (see Appendix B in the Supplementary Material).

As explained in Section III and Table I, our full GW system is intended to both align visual and language inputs into a common latent space and to translate inputs from one domain into the other. These different objectives correspond to our "primary properties" P_{cont} and P_{tr} and are enforced in our system via distinct training losses. In addition, our training setting is semi-supervised, i.e., we employ both supervised and unsupervised losses. The unsupervised objectives (corresponding to our "secondary properties" P_{dcy} and P_{cy}) are based on the cycle-consistency principle and can be thought of as regularization terms that can accelerate the optimization process and improve the model's generalization. We now describe the different loss components in detail.

1) Translation Loss: The first primary property P_{tr} is handled by the translation loss, where we predict one domain from the other (and vice versa), using pairs of matching examples (x_i, y_i) with $i \in \mathcal{S}$. First, let us define the translation function from a domain to another as

$$\tau_{v \to t}(x_i) = d_t(e_v(x_i)) \quad \forall i \in \mathcal{S}. \tag{1}$$

Now, we can express the translation objective

$$\mathcal{L}_{\text{tr}}^{v\to t} = \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \ell_{\text{mse}}(\tau_{v\to t}(x_i), y_i) \tag{2}$$

where ℓ_{mse} measures the Euclidian distance in the target space.[2] The full translation loss is the average of the losses from both "directions"

$$\mathcal{L}_{\text{tr}} = 0.5\left(\mathcal{L}_{\text{tr}}^{v\to t} + \mathcal{L}_{\text{tr}}^{t\to v}\right). \tag{3}$$

2) Contrastive Loss: P_{cont} is optimized using the contrastive loss

$$\mathcal{L}_{\text{cont}} = -\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}} \ell_{\text{CE}}\left(\frac{e_v(x_i)\cdot e_t(y_j)}{\|e_v(x_i)\|\|e_t(y_j)\|}, \mathbb{1}_{i=j}\right) \tag{4}$$

where $\mathbb{1}_{i=j} = 1$ when $i = j$ and 0 otherwise, and

$$\ell_{\text{CE}}(p, q) = q\log(p) + (1-q)\log(1-p).$$

In a nutshell, the contrastive loss ensures that the normalized dot product between matching exemplars across modalities is close to 1 (i.e., that the latent vectors are aligned) but close to 0 for nonmatching exemplars (i.e., the latent vectors are orthogonal). The translation objective [see (3)] optimizes both the encoders and decoders, which allows us to use it as a standalone objective in some of our ablated models. On the other hand, contrastive learning [see (4)] only optimizes the encoders. When learning with the contrastive loss, we, thus, require at least one additional objective [see Table I (right)].

3) Cycle-Consistency: Unsupervised objectives (corresponding to the secondary properties of Section III) are split into (full) cycle P_{cy} and demi-cycle P_{dcy} consistency losses. They are defined over the entire training set ($\mathcal{U}_v$ or $\mathcal{U}_t$) rather than only the paired data $\mathcal{S}$.

To introduce the full cycle-consistency loss (P_{cy}), we first define the cycle function by combining two translations

$$c_v(x_i) = \tau_{t\to v}(\tau_{v\to t}(x_i)) \quad \forall i \in \mathcal{U}_v. \tag{5}$$

Now, let us define the loss over (for instance) the visual modality as

$$\mathcal{L}_{\text{cy}}^{v} = \frac{1}{|\mathcal{U}_v|}\sum_{i\in\mathcal{U}_v} \ell_{\text{mse}}(c_v(x_i), x_i) \tag{6}$$

so that the full objective when training with two modalities is

$$\mathcal{L}_{\text{cy}} = 0.5\left(\mathcal{L}_{\text{cy}}^{v} + \mathcal{L}_{\text{cy}}^{t}\right). \tag{7}$$

4) Demi-Cycle-Consistency: Finally, the demi-cycle-consistency loss over (for instance) the visual domain is defined by

$$\mathcal{L}_{\text{dcy}}^{v} = \frac{1}{|\mathcal{U}_v|}\sum_{i\in\mathcal{U}_v} \ell_{\text{mse}}(\tau_{v\to v}(x_i), x_i). \tag{8}$$

Note that evaluating $\tau_{v\to v}$ defined in (1) with the same source and target domains amounts to performing a demi-cycle $d_v(e_v(x))$. Then, we obtain the full loss P_{dcy} by averaging the two possible demi-cycles

$$\mathcal{L}_{\text{dcy}} = 0.5\left(\mathcal{L}_{\text{dcy}}^{v} + \mathcal{L}_{\text{dcy}}^{t}\right). \tag{9}$$

Both unsupervised objectives serve a different purpose: the cycle-consistency loss ensures that the two domains are synchronized by translation (i.e., $\tau_{t\to v}$ and $\tau_{v\to t}$ are mutually inverse functions); the demi-cycle-consistency ensures that d_v (respectively, d_t) and e_v (respectively, e_t) are inverse functions of one another and, thus, forces the GW to coordinate the representations of the domains.

C. Is Supervision Necessary?

Cycle-consistency and demi-cycle-consistency losses are intended to align the encoders and decoders (and the resulting translations) across the two modalities so that each representation can be consistently inverted. However, aligning two domains without any additional constraint is an intrinsically ambiguous problem. Indeed, if there exists at least one bijection from one domain to the other, then, by randomly permuting the samples, we can produce many other equally valid bijections. For example, imagine mapping a letter domain $\{a, b, c\}$ onto a number domain $\{1, 2, 3\}$. Any one-to-one mapping is technically correct, but we might want to enforce that $a \leftrightarrow 1$, $b \leftrightarrow 2$, and $c \leftrightarrow 3$. Unsupervised learning techniques cannot directly enforce this constraint, but a small amount of supervision (i.e., labeled examples) might be sufficient. From the example above, if we additionally provide that a maps to 1 and c maps to 3, then the ambiguity is completely removed. This explains why supervision can be important for our multimodal learning problem, i.e., why translation and/or contrastive losses are needed. However, just like in our $\{a, b, c\}$ example, the number of labeled samples that are needed may be relatively small. The need for supervision will be explicitly quantified in our experiments.

D. Combining the Objectives

The final loss function is a combination of the four different objectives with different weights

$$\mathcal{L} = \alpha_{\text{tr}}\mathcal{L}_{\text{tr}} + \alpha_{\text{cont}}\mathcal{L}_{\text{cont}} + \alpha_{\text{cy}}\mathcal{L}_{\text{cy}} + \alpha_{\text{dcy}}\mathcal{L}_{\text{dcy}}. \tag{10}$$

In our implementation, the shared workspace is implicit, i.e., it emerges from the chosen training objectives. For example, training a model with only the translation loss or the cycle loss does not truly produce a GW. Indeed, they do not force the output of the encoders to project into a similar space, effectively resulting in a situation akin to the illustration in Fig. 1. On the contrary, the contrastive loss (and, to some extent, the demi-cycle loss, as per relation R_3) explicitly forces the encoders' output to be aligned across modalities, effectively resulting in the situation illustrated in Fig. 4. Thus, by setting the weight of some of the loss terms to zero in (10), we can easily modulate the effective architecture of the model and probe the functional relevance of our two factors of interest, GW and semi-supervision, as proposed in Table I.

[2] In the case of the proto-language, we combine an mse loss for the size, rotation, location, and color, with a cross-entropy loss for the prediction of the object category.

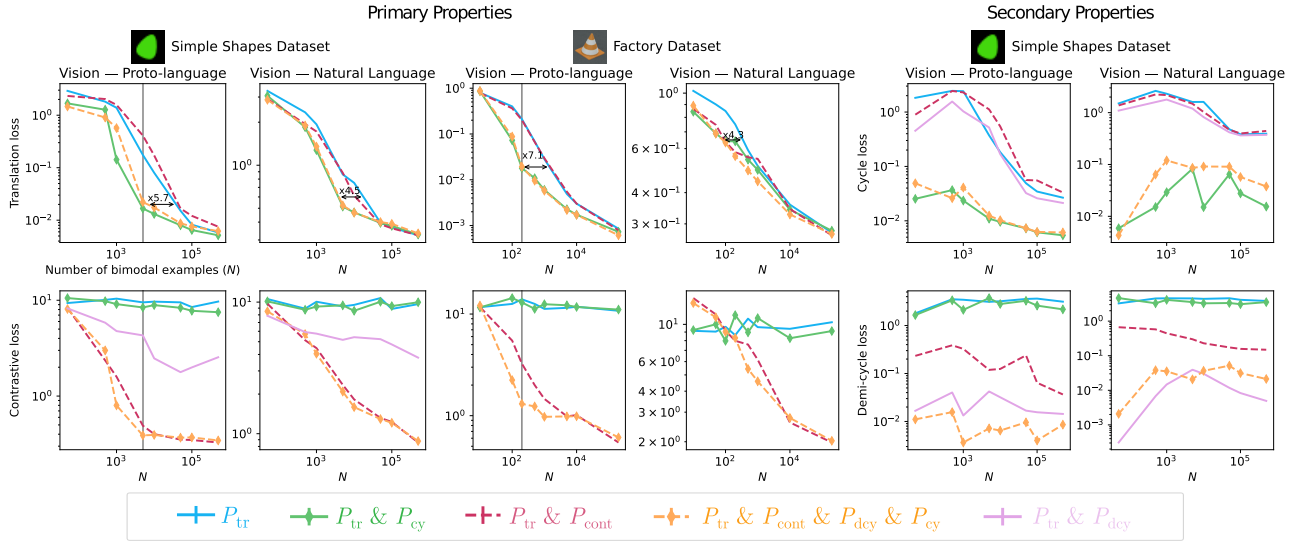

Fig. 5. Left: evaluates the primary properties (translation and contrastive alignment). Right: assesses the secondary properties (cycle and demi-cycle-consistency). Each point in each graph is a different model trained until convergence, using a particular number of matched bimodal examples N (x-axis). Dashed lines correspond to GW models, and curves with markers are semi-supervised models. P_{tr} & P_{dcy} was included as a way to assess relation R_3. The first and second rows on the left display the test translation and contrastive losses of the selected models, respectively. The first and second rows on the right show the test cycle and demi-cycle losses, respectively. Columns refer to different language modalities (proto-language or natural language) (the vertical gray line in the leftmost column marks the chosen value of N that will be used later to assess the influence of the total number of unsupervised data samples).

VI. Experiments

Our main experiments involve the *Simple Shapes* dataset, composed of $K = 500\,000$ image–text pairs (in Section VII, we will also validate our findings on another dataset). We report all translation/alignment experiments with both language types, i.e., vision $\leftrightarrow$ proto-language training and vision $\leftrightarrow$ natural language training. To characterize the need for labeled data across the two domains, we artificially split our training dataset into aligned data $\mathcal{S}$ (both domains are available) and unaligned data $\mathcal{U}_v$ and $\mathcal{U}_t$ (only one of the two domains is available), by randomly selecting $N \leq K$ matched examples from our original set. During training, we use the pool of N matched samples and another of $2K$ unmatched samples (for each modality, $K - N$ that are truly unmatched, plus N of $\mathcal{S}$ that have been artificially decoupled). We create batches by drawing samples from these two pools with equal probability, thus having equal numbers of paired and unpaired data at every learning step, regardless of the value of N. (However, of course, for small N, the same paired data will reoccur more often during training, while the training diversity will increase with increasing N).

We optimize the model with different values for $N \in \{50, 100, 500, 1000, 5000, 10\,000, 50\,000, 100\,000, 500\,000\}$ using the loss in (10). For the loss coefficients, we fix $\alpha_{\text{tr}} = 1$. For α_{cont}, α_{cy}, and α_{dcy}, we either do not use the loss (coefficient set to 0, reflecting the model choice in Table I) or choose, among three possible values $\{0.1, 1, 10\}$, the one yielding the best result for both translation and contrastive objectives.[3]

Retrospectively, we found that keeping a coefficient of 1 for translation, cycle and demi-cycle, and a low coefficient for the contrastive loss works in most of our experiments. Indeed, we used these hyperparameters for the COCO experiments instead of the grid search strategy (using a contrastive coefficient of 0.05). Regardless of the training regime (i.e., the value of N), we evaluate all of our models on the same independent test set of 1000 images and matching descriptions.

A. Primary Properties

1) Simple Shapes: We start by evaluating the models on the two primary properties P_{tr} and P_{cont}. We report the value of the translation and contrastive test losses as a function of N in Fig. 5 (left). Each point in the graph is a model trained until convergence. To facilitate comparisons, the same matched examples are used for all models with the same value of N.

To relate the curves with the models listed in Table I, we use the following conventions: curves with solid lines (respectively, with dashed lines) correspond to models with the GW property (respectively, without the property), curves with no marker (respectively, with diamond markers) correspond to models without the semi-supervision property (respectively, with supervision), and the color of the curves also matches the text color in Table I.

Note that we consistently test all four baseline models for both translation and contrastive objectives even though some models are not explicitly trained to optimize them. This allows us to verify our four proposed relations R_1–R_4 (e.g., P_{tr} & P_{dcy} is not trained with a P_{cont} objective, but we can still observe alignment, in accordance with relation R_3).

As expected, we observe in Fig. 5 that increasing the number of aligned examples improves all of the models' translation performance. However, the different models improve at distinct rates. We observe that semi-supervised models (with diamond markers) require significantly fewer annotations to obtain the same results. For instance, a semi-supervised

[3]We choose the coefficient that minimizes a weighted average of the translation and contrastive losses. To select the weights of the weighted average, we train one model with only a translation loss and one with only a contrastive loss, and we select the weights so as to equalize the translation/contrastive losses at the end of training.

model trained with $N = 5000$ matching pairs {image, proto-language annotation} performs approximately, as well as a fully supervised model trained with $N = 30\,000$ matching pairs—an $\sim$ sixfold improvement. Similarly, a model trained to translate between images and natural language captions using semi-supervision requires 4.5 times fewer matching examples than the equivalent fully supervised model. In short, the advantage of semi-supervision is readily apparent for the translation property.

Looking at the contrastive property, we see that only the models equipped with a GW perform well (dashed lines). Indeed, without a GW, the encoded representations from each domain do not need to be aligned with each other (see illustration in Fig. 1). For example, in a model trained only for translation (blue curve), encoding/decoding sequences [such as $d_t(e_v(x))$ or $d_v(e_t(y))$] can be viewed as direct (merged) translation functions [that is, $\tau_{v\to t}(x)$ or $\tau_{t\to v}(y)$; see (1)], so there is no intermediate latent multimodal representation to speak of.

Measuring the contrastive loss also allows us to evaluate the validity of relation R_3: optimizing translation and demi-cycle losses (without explicitly optimizing the contrastive loss) should be sufficient to achieve property P_{cont}. Theoretically, this relation holds for samples in $\mathcal{S}$, given that the encoders are injective functions. To assess whether this relation can generalize to the test set, we trained additional models with only translation and demi-cycle losses and plotted the resulting curve (in pink) in Fig. 5. We see that combining translation and demi-cycle losses can help align the visual and text multimodal representations compared to a translation-only model. However, the resulting alignment is partial, i.e., weaker than when explicitly optimizing the contrastive loss. This likely indicates that the trained models cannot perfectly generalize to the test data and/or our encoders are not strictly injective functions.

In conclusion, semi-supervision was shown to be beneficial to learn the translation primary property P_{tr} by decreasing the need for annotated data. In addition, including a GW in the architecture permits multimodal alignment in the intermediate latent space, thereby satisfying property P_{cont}. Overall, the best model to jointly satisfy our two primary properties is the one combining a GW architecture with self-supervised training (P_{tr} & P_{cont} & P_{dcy} & P_{cy}).

2) Factory: As done with the *Simple Shapes* dataset, we used the loss in (10) to optimize the various models with different values for $N \in \{50, 100, 200, 300, 500, 5000, 10\,000, 200\,000\}$. We used a translation loss coefficient $\alpha_{\text{tr}} = 1$ for all experiments. The cycles, demi-cycles, and contrastive coefficients were chosen among $\alpha_{\text{cy}} \in \{1, 2, 4, 8, 10\}$, $\alpha_{\text{dcy}} \in \{1, 2, 4, 8, 10\}$, and $\alpha_{\text{cont}} \in \{0.0005, 0.005\}$, so as to jointly optimize the primary properties P_{tr} and P_{cont}.

For these validation experiments, we focused on the two primary properties P_{tr} and P_{cont}. Fig. 5 shows the test performance of each model trained with a certain amount of bimodal matched examples (N). Each point corresponds to one model trained until convergence; for a given value of N, all compared models used the same training data. Training batches were generated in the same way, as described in Section VI (but with $K = 200\,000$ instead of $K = 500\,000$).

Overall, the results are similar to the ones obtained with the *Simple Shapes* dataset. When the number of bimodal matched examples (N) increases, the translation loss of all models decreases, for both proto-language and natural language domains. There is also a performance gap between models trained only with supervision (blue and red curves) and semi-supervised models trained with unimodal (unmatched) data (orange and green curves). For the proto-language translation, the greatest effect of semi-supervision is visible at $N = 200$. At this point, the semi-supervised models have the same translation loss as the supervised models trained with $\sim$7 times more bimodal paired examples. A qualitatively similar but more modest effect of semi-supervision is also observed for the natural language translation (with, e.g., a gap of $\times 4.3$ between the model trained with translation loss only and the one trained with an additional cycle loss).

As previously, multimodal alignment [as measured by the contrastive loss in Fig. 5 (bottom)] only emerges for models including a GW (red and orange curves, P_{tr} & P_{cont} and P_{tr} & P_{cont} & P_{dcy} & P_{cy}). If no alignment constraint is applied on the encoders (with contrastive and/or demi-cycle losses), then the encoded visual and linguistic data are not aligned (blue and green curves, P_{tr} and P_{tr} & P_{cy}). The benefits of semi-supervision can also be observed here for the contrastive loss. In Fig. 5 (bottom), the semi-supervised GW model (in orange) outperforms the fully supervised GW model (red), i.e., it reaches a similar contrastive loss value with fewer matched examples. This advantage is visible for both vision–proto-language alignment and (albeit to a lower extent) vision–natural language alignment.

In summary, our main conclusion is also independently replicated on the *Factory* dataset. Semi-supervision primarily helps in optimizing the translation property with fewer annotated bimodal data, while the inclusion of a GW in the architecture is important to ensure the alignment of multimodal representations. Overall, the model that best satisfies our two primary properties P_{tr} and P_{cont} is the one combining a GW architecture with a semi-supervised training setting (P_{tr} & P_{cont} & P_{dcy} & P_{cy}).

B. Secondary Properties

In addition to the *primary* properties, we described in Section III two desirable *secondary* properties P_{cy} and P_{dcy} that multimodal networks could possess. Fig. 5 (right) shows the performance of the models on these secondary properties. Unsurprisingly, the models that explicitly optimize the corresponding losses reach the best performance: P_{tr} & P_{cy} and P_{tr} & P_{cont} & P_{dcy} & P_{cy} for the cycle loss and P_{tr} & P_{dcy} and P_{tr} & P_{cont} & P_{dcy} & P_{cy} for the demi-cycle loss. However, some of the other models also perform relatively well even though none of them were explicitly trained for the secondary properties. We surmise that this could be due to the relations R_1 and R_2 that exist between primary and secondary properties. In line with relation R_1, we see that all models, as they are trained with a translation loss, improve their cycle loss when the number of annotations N becomes sufficient. According to relation R_2, we see that the P_{tr} & P_{cont} curve has a much lower demi-cycle loss than P_{tr} (in blue) and P_{tr} & P_{cont} & P_{dcy} & P_{cy} (in green) curves.

Here again, we can conclude that the best model to jointly satisfy the two *secondary* properties is the one combining a GW architecture with self-supervised training (P_{tr} & P_{cont} & P_{dcy} & P_{cy}).

C. Downstream Tasks

We have seen that a GW can help a multimodal system learn useful representations for translation and alignment. Logically, these improved multimodal representations should also facilitate performance in downstream tasks, in particular, for multimodal transfer. We explicitly test this prediction by comparing the performance of our different trained models (with/without a GW) on two downstream tasks: the "odd-one-out" (OOO) and the shape classification tasks.

1) Odd-One-Out: For each trial of this task, three samples are given (at first, only in the visual domain). Two of them share at least one attribute (shape, size, color, position, or orientation), while the last image differs from the other two across all attributes and is, thus, considered the OOO. Fig. 6(a) shows some examples: in the first row, the first two images share the same shape and a similar orientation, and the OOO is the third image. Appendix D (see the Supplementary Material) details how we construct the dataset and how the triplets are selected.

To solve this task, we start by using the pretrained models from Section VI-A, with their encoders frozen. We concatenate three encoded latent vectors $[e_v(x_1), e_v(x_2), e_v(x_3)]$. We then classify the OOO with a two-layer classifier (from 3×12 to 16 hidden neurons and then 16 to 3 neurons) trained specifically on this task. We additionally evaluate three baseline models.

1) *task-optimized (TO) encoder + classifier* where we keep the pretrained unimodal visual VAE to encode the image and then jointly train the encoder e_v (from scratch) and classifier end-to-end for the downstream OOO task;
2) *no encoder + TO classifier* where we remove the encoder e_v and directly learn the OOO classifier from the unimodal pretrained model;
3) *random encoder + TO classifier* where we use a random and frozen encoder e_v, and only train the classifier.

Note that the first baseline has the same overall number of parameters as the evaluated models but more trainable parameters (in the encoder e_v), the second baseline has the same number of trainable parameters but fewer overall parameters, and the last baseline has the same number of both overall and trainable parameters.

The first column in Fig. 6(b) plots the OOO accuracy obtained by the models when tested in the same condition as during training (vvv, i.e., comparing three images). All of our pretrained models (trained using translation and/or alignment objectives) outperform the three baselines when given enough supervision ($N \gtrapprox 10\,000$). This is even true for the strongest baseline trained end-to-end ("TO encoder + classifier"), which has the same architecture but more trainable parameters allocated for the downstream task. This shows that pretraining multimodal representations can be helpful for downstream visual tasks; however, the various pretrained models remain qualitatively comparable in this setting.

Then, we investigate whether the models trained with images (vvv condition) can generalize across domains (i.e., transfer learning): what would happen if we test these models using representations coming from language instead $[e_t(y)]$? The second and third columns in Fig. 6(b) plot the models' transfer learning performance, respectively, in the ttt condition (new modality) and in a ttv condition (a truly cross-modal setting, where two of the representations come from language descriptions and one from a visual one). In both cases, we see that only the models that have a GW (P_{tr} & P_{cont} and P_{tr} & P_{cont} & P_{dcy} & P_{cy}) are able to properly generalize to the other domain; the models without a GW ("translation" and "trans. + full cycles"), on the other hand, suffer a considerable drop in performance when transferring across domains. Nonetheless, their performance remains above chance level (first/third), indicating that they can also transfer some (limited) knowledge across domains. Our hypothesis is that these models may have learned to solve the task by comparing the distances between encoded representations $[e_v(x)]$ and selecting the furthest one as the OOO. This strategy could generalize, to some extent, to latent vectors from a different encoder $[e_t(y)]$, even if the resulting representations are not actually aligned. In conclusion, we note again that models trained using a GW (and in particular, the one trained with semi-supervised learning, "all sup. + all cycles") outperform the other ones in a downstream task requiring domain transfer.

2) Shape Classification: As a second downstream task, we evaluate our models on a shape classification task on the Simple Shapes dataset (see Table II). We test two different settings: "linear probe" where we train a linear shape classifier from the multimodal (GW) representation and "zero-shot" where we use the alignment property of the model in the GW to classify images by matching them with captions *à la CLIP*.

More specifically, for the linear probe setting, we train a linear classifier on 500 000 pairs of (GW_v, s) or (GW_t, s) where GW_m is the GW representation of modality $m \in \{v, t\}$ and s is the shape category of the object (i.e., diamond, egg, and triangle); we measure performance on an independent test set of 1000 samples.

For the zero-shot setting, we generate 100 objects from each shape category with random attributes (rotation, size, position, and color), and we create the associated 100 sentences with our heuristic. We keep the same 100 sentences for each category and only change the shape information. For each shape, the sentences are encoded into the language encoder (BERT + projection). The class representatives (or "prototypes") are obtained either by averaging the 100 outputs of the language encoder and then encoding the average into the GW representation [column $\text{GW}_t(\overline{\text{BERT}})$] or by first encoding each sentence into the GW representation and then averaging [column $\overline{\text{GW}_t(\text{BERT})}$]. We then follow the "zero-shot classification" procedure of CLIP [2] to match each image input to the most similar prototype.

We see in the results presented in Table II that the worst model is P_{cont}, trained only for alignment with contrastive learning (as done in the CLIP study). This model aligns the representations but does not have a decoder, i.e., no broadcast (see Table I). P_{tr} & P_{cont} performs globally better than P_{cont}

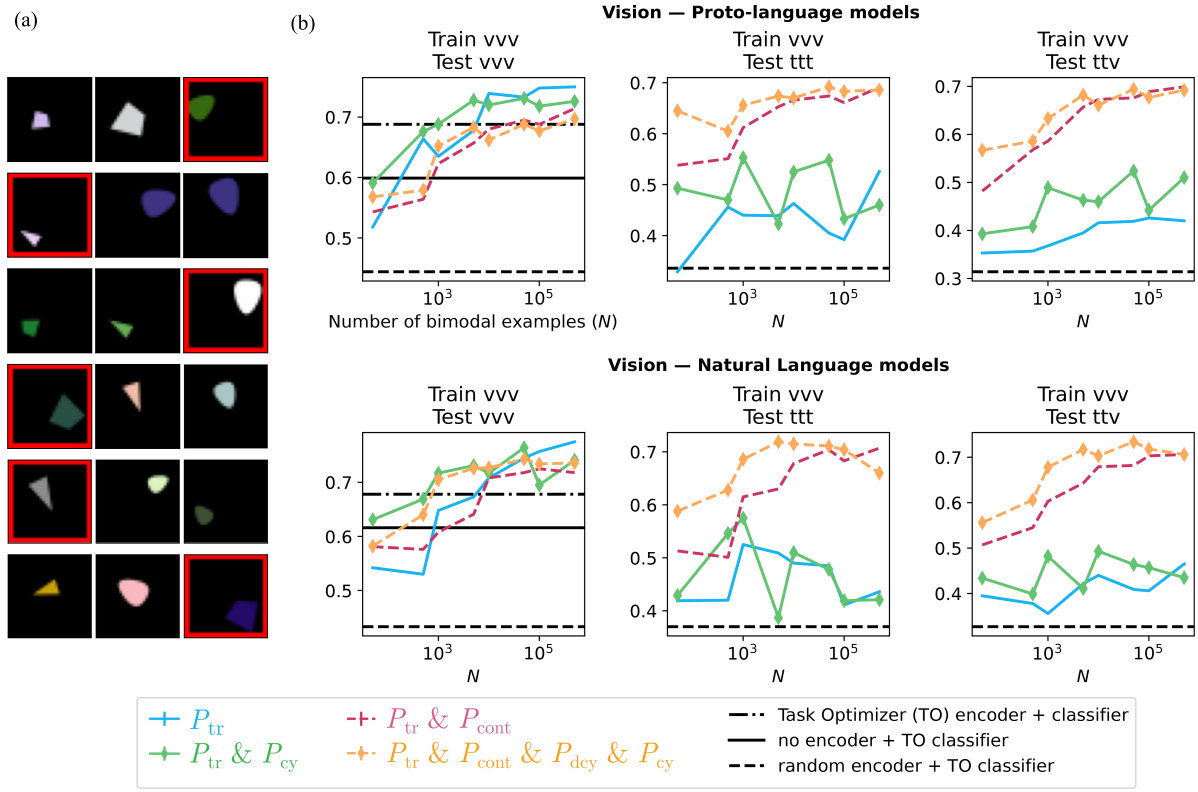

Fig. 6. Downstream task. (a) Examples of the OOO dataset. Each row corresponds to a triplet of images where two images have at least one common attribute, and the third one (with the red border) differs from the other two. (b) We plot the results obtained on this task. Each point in the colored curves represents a model pretrained on the domain translation and alignment task (using N annotated training pairs, x-axis), with the encoders subsequently frozen. For each model, we train a new classifier to predict the OOO from the concatenation of three encoded visual representations "vvv" $[e_v(x_i)]$. As baselines, we also train a new "TO encoder + classifier" end-to-end on the OOO task (dash-dotted line), we train a classifier directly on the visual latent representation instead of the encoded one (no encoder; solid black line), or we train a classifier based on an encoder with randomized weights (dashed line). In the first column, we show the results when testing the model on the trained domain "vvv." In the second column, we evaluate the "vvv"-trained classifier on three language (or proto-language) representations "ttt" $[e_t(y_i)]$. In the last column, we test it on a cross-modal version comprising two linguistic representations and one visual representation "ttv."

TABLE II
LINEAR PROBE AND ZERO-SHOT PERFORMANCE ON THREE-WAY SHAPE CLASSIFICATION ON THE SIMPLE SHAPES DATASET. ALL MODELS WERE TRAINED USING ALL AVAILABLE PAIRED DATA ($N = 500\,000$). FOR ZERO-SHOT CLASSIFICATION, IMAGE INPUTS WERE ENCODED INTO THE MULTIMODAL (GW) REPRESENTATION AND COMPARED WITH PROTOTYPE VECTORS, CALCULATED EITHER BY ENCODING THE AVERAGE BERT EMBEDDING OF 100 CLASS-REPRESENTATIVE TEXT CAPTIONS [$\mathrm{GW}_t(\overline{\mathrm{BERT}})$] OR BY AVERAGING THE ENCODED REPRESENTATIONS ACROSS CAPTIONS [$\overline{\mathrm{GW}_t(\mathrm{BERT})}$]

	Linear Probe		Zero-shot	
	GW_v	GW_t	$GW_t(\overline{\mathrm{BERT}})$	$\overline{GW_t(\mathrm{BERT})}$
P_{tr} & P_{cont} & P_{dcy} & P_{cy}	**0.9989**	**0.9997**	**0.8477**	**0.9863**
P_{tr} & P_{cont}	0.9901	0.9921	0.7402	0.9746
P_{cont}	0.8957	0.9176	0.5947	0.8721

due to its training with an additional translation loss that constrains the decoders and provides broadcast abilities. Finally, our "target" model P_{tr} & P_{cont} & P_{dcy} & P_{cy} performs best in all settings; this can be attributed to its improved GW, with the demi-cycle loss reinforcing alignment (Relation R_3) and the cycle loss improving the broadcast ability (as demonstrated already by improved translation in Fig. 5).

D. Effect of Unpaired Data

Up until now, we always used all available data as our unsupervised training sets $\mathcal{U}_v$ and $\mathcal{U}_t$ and only varied the number N of paired multimodal examples in the supervised training set $\mathcal{S}$. Here, we analyze how the performance of our semi-supervised models depends on the size of the unsupervised training sets. Let us define as M the number of strictly unpaired examples in the dataset such that $N + M$ is the total number of examples in the dataset. In our previous experiments, $M + N$ was always fixed at $M + N = 500\,000$, and N varied from 0 to 500 000.

Fig. 7(a) shows the performance of new models trained with a fixed number $N = 5000$ of paired samples and with M increasing from 0 to 495 000 (so $N + M$ varies between 5000 and 500 000). That is, the results plotted in Fig. 7(a) and those plotted in Fig. 5 (left column) can be envisioned as reflecting the same underlying "3-D surface" where losses are expressed as a function of N on one axis and as a function of $N+M$ on the other; the two figures depict cross sections of this same surface along orthogonal dimensions, intersecting on the gray vertical lines visible in each figure, i.e., $N = 5000$ and

TABLE III
IMAGE–TEXT RETRIEVAL PERFORMANCE FOR MODELS TRAINED WITH 20% PAIRED DATA FROM COCO TRAIN SET + RESTVAL OF THE KARPATHY [47] 1K SPLIT (AVERAGED OVER FIVE RUNS)

Model	Caption retrieval (i2t)				Image retrieval (t2i)			
	R@1	R@5	R@10	med rank	R@1	R@5	R@10	med rank
P_{tr} & P_{cont} & P_{dcy} & P_{cy}	34.4 ± 0.6	65.1 ± 0.3	79.0 ± 0.2	2.8 ± 0.2	30.3 ± 0.5	65.3 ± 0.1	79.0 ± 0.2	3.0 ± 0.0
P_{cont}	31.8 ± 0.6	62.5 ± 0.5	76.0 ± 0.4	3.0 ± 0.0	27.3 ± 0.2	62.0 ± 0.4	77.3 ± 0.3	3.6 ± 0.2

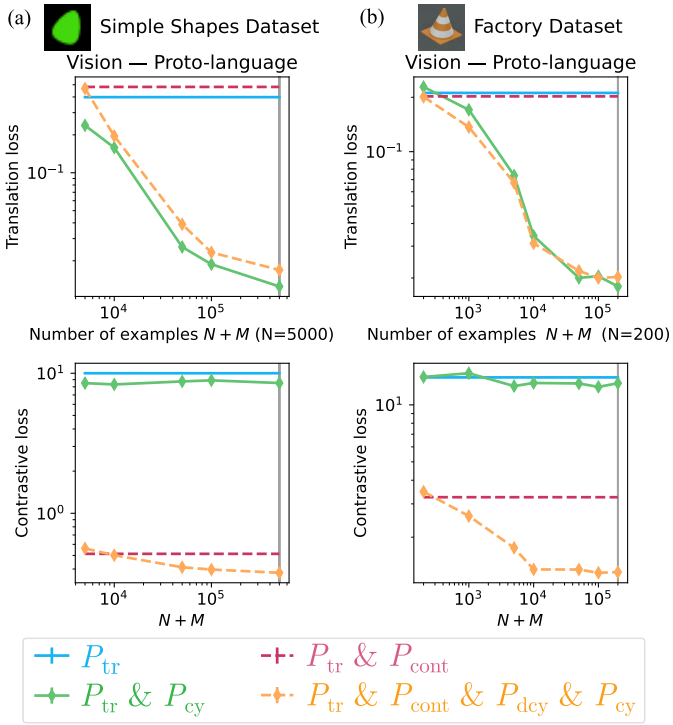

Fig. 7. Influence of the number of unpaired examples M on the models' performance. In order to facilitate comparisons, we increase M by adding unimodal data samples to the same set such that two models trained with $M1 < M2$ unpaired examples will share the same $M1$ training examples. (a) On the *Simple Shapes* dataset, we fix $N = 5000$ and $N + M$ varies from 5000 to 500 000. The results along the gray vertical line correspond to the results of Fig. 5, first column, on the same gray line. (b) Same measure is applied to the *Factory* dataset. Here, we fix $N = 200$ and $N + M$ varies from 200 to 200 000. The results along the gray vertical line correspond to the results of Fig. 5, first column, on the same gray line.

$N + M = 500\,000$. Indeed, the results along the gray lines are identical in both plots.

The results reveal how the semi-supervised models make use of additional unpaired data. The purely supervised models ("translation" and "trans. + cont.") only rely on the number of paired examples N [resulting in horizontal lines in Fig. 7(a)]. However, the semi-supervised models improve with additional unpaired samples on the translation and contrastive objectives. Fig. 7(a) also highlights that these improvements eventually saturate with increasing M. In other words, a dataset of roughly 50 000 unpaired samples could have been sufficient to observe qualitatively similar behavior in our semi-supervised models as the full *Simple Shapes* dataset with 500 000 samples.

Fig. 7(b) depicts very similar results for the Factory dataset as for the *Simple Shapes* dataset. We fixed the number of bimodal matched training samples to $N = 200$, as it was where the effect of semi-supervision was highest (gray line in Fig. 5). When enough additional unimodal unmatched data are available (roughly between $M + N = 1000$ and $M + N = 10\,000$), the performance of semi-supervised models quickly surpasses the supervised ones. This implies that a dataset of $\sim$10 000 unpaired samples could have been sufficient to observe qualitatively similar behavior in our semi-supervised models as the full *Factory* dataset with 200 000 samples.

VII. RETRIEVAL ON COCO CAPTIONS

Up until now, we have exclusively tested our models on synthetic data. It has allowed us to systematically test the effect of each loss on a controlled dataset (Simple Shapes and Factory). To verify the effectiveness of our method on a more natural dataset, we now train GW on the COCO Captions dataset.

In the Simple Shapes and Factory experiments, all our unimodal modules were (variational) autoencoders. This choice was made to allow us to visualize (as an image) the outcome of translations, cycles, and demi-cycles and help us evaluate the relevance of each model. This is unfortunately not as easily achievable with more natural datasets (autoencoding models exist for natural images but are either too inaccurate or too computationally intensive for our purposes). We, thus, decided to remove the auto-encoder constraint for this experiment and used another proxy (than image reconstruction) to evaluate performance. We selected the COCO dataset [49] for the experiments and image/caption retrieval as the proxy task.

To emphasize the importance of semi-supervised training, we only keep 20% of the train + restval split as paired examples (P_{tr} and P_{cont}) and use the full training set for the unsupervised losses (P_{cy} and P_{dcy}). We use a ResNet50 [50] pretrained on ImageNet [51] for our visual domain encoder and Beijing Academy of Artificial Intelligence general embeddings (BGEs) [52] for the text encoder. The full GW model (P_{tr} & P_{cont} & P_{dcy} & P_{cy}) is trained on the COCO dataset with the translation, demi-cycle, cycle, and contrastive objectives and compared against P_{cont} (a baseline trained only with a contrastive loss as in the CLIP study [2]). We evaluate their retrieval performance as a downstream task without additional training.

Table III reports the average recall of the top K samples ($R@K$) on a fivefold 1K random split of Karpathy's test split [47]. We also provide the median rank of the correct sample among the 1000 alternatives. Note that for caption retrieval, there are five correct captions for each image, and we keep the minimum rank of the five captions. We see that P_{tr} & P_{cont} & P_{dcy} & P_{cy} performs better than P_{cont}. These results

go in the same direction as our previous results on the Simple Shapes and Factory datasets. They show that models trained with a semi-supervised GW have a better alignment and better generalize to downstream tasks (here, retrieval). Note again that our models were never trained on the retrieval task, and the performance is only a consequence of the translation, contrastive, and cycle-consistency training.

In addition to image–text retrieval, we can also evaluate the models trained on COCO for out-of-domain classification accuracy on ImageNet. This time, we use all training dataset examples for both supervised and semi-supervised objectives. To test the out-of-domain generalization, we learn a linear classifier from the visual GW representation to the 1000 classes of the ILSVRC 2012 ImageNet dataset. A model trained solely with the contrastive loss (P_{cont}) classifies with $50.3 \pm 0.2\%$ accuracy and lags behind the full GW model trained with all losses (P_{tr} & P_{cont} & P_{dcy} & P_{cy}) at $55.7 \pm 0.2\%$ accuracy (standard error of the mean across the 1000 classes of the accuracy difference between the two compared models). These results again show the advantage of our GW paradigm over a pure contrastive loss as used, e.g., in the CLIP [2] study.

VIII. Conclusion

A. Summary

The GW theory offers an account of multimodal integration in the human brain [14], [15], [32]. Prior work [33] has provided theoretical insights on how to use a GW architecture to connect the latent spaces of pretrained deep neural networks for different modalities. In this work, we present an initial empirical validation of some of these ideas, by investigating the bimodal (vision-language) integration abilities that emerge with versus without a GW latent space. We further explore the possibility of improving the training via unsupervised objectives by varying the amount of matched (bimodal) data available for training the GW encoders and decoders.

Our results show that semi-supervision is particularly important to achieve efficient vision-language *translation*. Semi-supervised models—with unsupervised cycle-consistency losses—needed approximately four to seven times fewer annotated bimodal examples than their fully supervised counterparts to reach the same bimodal translation accuracy. The GW architecture, on the other hand, is critical for *(contrastive) alignment* between the vision and language representations. Overall, the semi-supervised GW model proved the best at jointly satisfying the two primary properties that we advocate for multimodal systems: *translation* and *contrastive alignment*. Furthermore, we showed that our semi-supervised GW pretraining method produced meaningful and better multimodal representations for downstream tasks than a dedicated model with the same number of weights. These aligned multimodal GW representations allowed the system to transfer knowledge from one modality to the other (i.e., bimodal and cross-modal domain transfer).

B. Limitations and Open Questions

Due to limited computational resources, all reported experiments were conducted with only one repetition (with an unoptimized seed set to 0 from the beginning). Thus, repeating all experiments with different random seeds for weight initialization and/or dataset splits (e.g., for the selected subset of N matched exemplars) could yield smoother performance curves and allow us to estimate statistical variability. However, we do not expect this to affect our general conclusions.

Similarly, we had to limit the present study to two handcrafted and relatively simple bimodal datasets (in addition to our tests in section VII using the COCO dataset in a restricted retrieval setting). Despite their simplicity, one advantage of these datasets was our ability to quickly and parametrically control image and text generation properties, which facilitated our understanding of the various model components. The fact that our main conclusions could be replicated across these two datasets already hints at their generality. However, an important next step would be to extend the study to more realistic, large-scale bimodal datasets and benchmarks. Paradoxically, the inherent diversity and richness of real-world data, instead of impeding our system, could result in a more precise bimodal alignment from unimodal data and, thus, lead to better semi-supervised training and improved generalization—provided that sufficient computational resources would be available to train our models on such large-scale datasets.

A related question is whether our findings could generalize to bimodal translation and alignment problems when the two domains are not bijectively related. For instance, images and proto-language descriptions in our datasets were bijectively related, in the sense that a unique attribute vector could be inferred from each image and vice versa. Comparing this situation to the vision/natural language setting (where linguistic ambiguities, synonymy, categorical terms, and so on challenged the one-to-one mapping between domains), we already saw that our approach can still work without a perfect bijection. However, it worked less well than when using proto-language. Would it still work at all if the multimodal correspondence was only very loosely defined, e.g., matching impressionist paintings to Baudelaire poetry? If not, could the presence of additional modalities (e.g., a limbic system encoding emotions and an auditory system to process rhymes and prosody) help the model resolve ambiguities? These are exciting questions for follow-up studies.

In addition to the multimodal integration abilities that we already demonstrated here, additional properties could be expected from our GW model that could be tested in future studies. In particular, prior work [12] has shown that models trained only with a contrastive loss to align information across domains (such as CLIP [2]) tend to filter out domain-specific data. This side-effect hinders their unimodal generalization performance, e.g., when comparing against expert visual models [12]. Our setting combining translation, contrastive alignment, and semi-supervised cycle-consistency objectives could be a way around this problem. Specifically, the demi-cycle property should help align multimodal representations while, at the same time, forcing the encoder to retain domain-specific information. This should result in preserved unimodal generalization abilities compared to models trained only for contrastive alignment. In other words, we could expect a GW model to learn to represent the union of the two domains, rather than just their intersection.

C. Future Model Extensions

The proposed strategy proved successful for bimodal vision-language integration. However, much remains to be

incorporated into our model's architecture before this promising approach could be considered a full implementation of the GW theory, thereby possibly rivaling human-level capabilities.

A first extension would be to increase the number of domains and modalities integrated into the workspace. A simple way to achieve this within our current architecture could be to use n specialist modules instead of just 2, yet only one module would access the GW at any given time. Training such a system could still be performed via a combination of unsupervised cycle-consistency objectives on unimodal data and supervised training from pairwise matched data, as done in the present study. Such an extension would resemble the recently proposed extension of the CLIP vision-language alignment model [2] to the new "ImageBind" model, aligning several distinct modalities to a visual representation space [53]. Our version, however, would be centered around a GW latent space, with a symmetric architecture that does not favor vision compared to other modalities, and using the necessary encoders and decoders to satisfy both translation and cycle-consistency properties.

In a second step, given the availability of multiple modules and their encoders and decoders to/from the GW, it could become useful to allow two or more modules to simultaneously encode their representations in the GW. As in the original formulation of the theory by Baars [14], [32], this would require a dedicated attentional system to control access to the GW and some sort of attention-dependent fusion mechanism to combine information from these modalities into the GW latent space.

Finally, another possible extension may be to include recurrent dynamics in the model. Having a model that can maintain its internal state over time but can also update it based on novel information from the internal or external environment is useful in many domains (planning, robotics, and so on). Moreover, more advanced modules could be envisioned in this dynamic context, such as a memory module [54], [55] or a world model [56].

D. Implications for Cognitive Science

The GW is a prominent theory of higher brain function. The present results already show that it is possible to start implementing this sort of cognitive strategy in multimodal deep learning systems. In particular, the demonstrated feasibility of using semi-supervised learning techniques can go some way toward reconciling these models with human multimodal learning (which relies on much less explicit supervision than the standard deep learning approaches). Nonetheless, the present findings by themselves do not prove or disprove the GW theory, as they do not yet constitute a full implementation of the GW framework [33]. However, with the extensions proposed above (see Section VIII-C: additional modalities, attention control system, and recurrent implementation with temporally extended inputs and outputs), it might become possible, in the relatively short term, to use this sort of artificial system to draw conclusions of relevance to Cognitive Science.

ACKNOWLEDGMENT

The authors would like to thank Alexandre Arnold for creating and sharing the WeBots environment used for the Factory dataset.

REFERENCES

[1] S. Harnad, "The symbol grounding problem," *Phys. D, Nonlinear Phenomena*, vol. 42, nos. 1–3, pp. 335–346, Jun. 1990.

[2] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, vol. 139, 2021, pp. 8748–8763. [Online]. Available: https://proceedings.mlr.press/v139/radford21a.html

[3] A. Frome et al., "Devise: A deep visual-semantic embedding model," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 2121–2129.

[4] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning—A comprehensive evaluation of the good, the bad and the ugly," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2251–2265, Sep. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8413121/

[5] J. Lu, C. Clark, R. Zellers, R. Mottaghi, and A. Kembhavi, "UNIFIED-IO: A unified model for vision, language, and multi-modal tasks," in *Proc. 11th Int. Conf. Learn. Represent.*, 2023, pp. 1–34. [Online]. Available: https://openreview.net/forum?id=E01k9048soZ

[6] C. Silberer and M. Lapata, "Grounded models of semantic representation," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn.* Jeju Island, South Korea: Association for Computational Linguistics, Jul. 2012, pp. 1423–1433. [Online]. Available: https://aclanthology.org/D12-1130

[7] D. Kiela and S. Clark, "Multi- and cross-modal semantics beyond vision: Grounding in auditory perception," in *Proc. Conf. Empirical Methods Natural Lang. Process.* Lisbon, Portugal: Association for Computational Linguistics, 2015, pp. 2461–2470. [Online]. Available: http://aclweb.org/anthology/D15-1293

[8] P. Hai, P. P. Liang, T. Manzini, L. P. Morency, and B. Póczos, "Found in translation: Learning robust joint representations by cyclic translations between modalities," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 6892–6899. https://ojs.aaai.org/index.php/AAAI/article/view/4666

[9] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, "CoCa: Contrastive captioners are image-text foundation models," *Trans. Mach. Learn. Res.*, Jan. 2022. [Online]. Available: https://openreview.net/forum?id=Ee277P3AYC

[10] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with CLIP latents," 2022, *arXiv:2204.06125*.

[11] J. Wei et al., "Emergent abilities of large language models," 2022, *arXiv:2206.07682*.

[12] B. Devillers, B. Choksi, R. Bielawski, and R. VanRullen, "Does language help generalization in vision models?" in *Proc. 25th Conf. Comput. Natural Lang. Learn.*, 2021, pp. 171–182. [Online]. Available: https://aclanthology.org/2021.conll-1.13

[13] X. Zhai et al., "LiT: Zero-shot transfer with locked-image text tuning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 18102–18112. [Online]. Available: https://ieeexplore.ieee.org/document/9878889/

[14] B. J. Baars, *A Cognitive Theory of Consciousness*. Cambridge, U.K.: Cambridge Univ. Press, 1993.

[15] S. Dehaene, M. Kerszberg, and J.-P. Changeux, "A neuronal model of a global workspace in effortful cognitive tasks," *Proc. Nat. Acad. Sci. USA*, vol. 95, no. 24, pp. 14529–14534, 1998.

[16] J. Lin, C. Gan, and S. Han, "TSM: Temporal shift module for efficient video understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019.

[17] K. Desai and J. Johnson, "VirTex: Learning visual representations from textual annotations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11157–11168. [Online]. Available: https://ieeexplore.ieee.org/document/9577368/

[18] M. B. Sariyildiz, J. Perez, and D. Larlus, "Learning visual representations with caption annotations," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 153–170.

[19] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 2756–2759. [Online]. Available: http://ieeexplore.ieee.org/document/5596017/

[20] D. He et al., "Dual learning for machine translation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 820–828.

[21] M. Artetxe, G. Labaka, E. Agirre, and K. Cho, "Unsupervised neural machine translation," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–12.

[22] A. Conneau, G. Lample, M. Ranzato, L. Denoyer, and H. Jégou, "Word translation without parallel data," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–14.

[23] G. Lample, A. Conneau, L. Denoyer, and M. Ranzato, "Unsupervised machine translation using monolingual corpora only," 2017, *arXiv:1711.00043*.

[24] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. Learn. Represent.*, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA, May 2015, p. 15.

[25] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2242–2251.

[26] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 30, 2017, pp. 700–708.

[27] Z. Yi, H. Zhang, P. Tan, and M. Gong, "DualGAN: Unsupervised dual learning for image-to-image translation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2849–2857.

[28] S. Chaudhury, S. Dasgupta, A. Munawar, M. A. S. Khan, and R. Tachibana, "Text to image generative model using constrained embedding space mapping," in *Proc. IEEE 27th Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2017, pp. 1–6.

[29] T. Qiao, J. Zhang, D. Xu, and D. Tao, "MirrorGAN: Learning text-to-image generation by redescription," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1505–1514.

[30] K. J. Joseph, A. Pal, S. Rajanala, and V. N. Balasubramanian, "C4Synth: Cross-caption cycle-consistent text-to-image synthesis," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 358–366.

[31] Y. Li, J.-Y. Zhu, R. Tedrake, and A. Torralba, "Connecting touch and vision via cross-modal prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10601–10610.

[32] B. J. Baars, "Global workspace theory of consciousness: Toward a cognitive neuroscience of human experience," in *Progress in Brain Research*. Amsterdam, The Netherlands: Elsevier, 2005, vol. 150, pp. 45–53. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0079612305500049

[33] R. VanRullen and R. Kanai, "Deep learning and the global workspace theory," *Trends Neurosci.*, vol. 44, no. 9, pp. 692–704, 2021. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0166223621000771

[34] A. Juliani, R. Kanai, and S. S. Sasai, "The perceiver architecture is a functional global workspace," in *Proc. Annu. Meeting Cogn. Sci. Soc.*, vol. 44, no. 44, 2022, pp. 955–961.

[35] A. Jaegle, F. Gimeno, A. Brock, A. Zisserman, O. Vinyals, and J. Carreira, *Perceiver: General Perception With Iterative Attention*, document eprint: 2103.03206, 2021.

[36] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA, May 2015, pp. 1–23. [Online]. Available: http://arxiv.org/abs/1409.0473

[37] P. Hu, L. Zhen, D. Peng, and P. Liu, "Scalable deep multimodal learning for cross-modal retrieval," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2019, pp. 635–644, doi: 10.1145/3331184.3331213.

[38] J. Tian, K. Wang, X. Xu, Z. Cao, F. Shen, and H. T. Shen, "Multimodal disentanglement variational AutoEncoders for zero-shot cross-modal retrieval," in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2022, pp. 960–969, doi: 10.1145/3477495.3532028.

[39] L. Zhen, P. Hu, X. Peng, R. S. M. Goh, and J. T. Zhou, "Deep multimodal transfer learning for cross-modal retrieval," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 798–810, Feb. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9236655/

[40] H. Yin, F. Melo, A. Billard, and A. Paiva, "Associate latent encodings in learning from demonstrations," in *Proc. AAAI*, vol. 31, no. 1, Feb. 2017, pp. 3848–3854, doi: 10.1609/aaai.v31i1.11040. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/11040

[41] R. Silva, M. Vasco, F. S. Melo, A. Paiva, and M. Veloso, "Playing games in the dark: An approach for cross-modality transfer in reinforcement learning," in *Proc. 19th Int. Conf. Auton. Agents MultiAgent Syst.*, 2020, pp. 1260–1268.

[42] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 10684–10695.

[43] G. Lample, M. Ott, A. Conneau, L. Denoyer, and M. Ranzato, "Phrase-based & neural unsupervised machine translation," 2018, *arXiv:1804.07755*.

[44] I. Higgins et al., "beta-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–22. [Online]. Available: https://openreview.net/forum?id=Sy2fzU9gl

[45] Webots. *Open-Source Mobile Robot Simulation Software*. Accessed: Jan. 31, 2023. [Online]. Available: http://www.cyberbotics.com

[46] X. Chen et al., "Microsoft COCO captions: Data collection and evaluation server," in *Computer Vision—ECCV 2014*. Cham, Switzerland: Springer, 2014, pp. 740–755.

[47] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3128–3137.

[48] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[49] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. ECCV*, vol. 14, 2014, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.

[50] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Aug. 2009, pp. 248–255.

[52] S. Xiao, Z. Liu, P. Zhang, and N. Muennighoff, "C-Pack: Packaged Resources To Advance General Chinese Embedding," 2023, _eprint: 2309.07597.

[53] R. Girdhar et al., "ImageBind one embedding space to bind them all," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 15180–15190.

[54] A. Graves, G. Wayne, and I. Danihelka, "Neural Turing machines," 2014, *arXiv:1410.5401*.

[55] A. Graves et al., "Hybrid computing using a neural network with dynamic external memory," *Nature*, vol. 538, no. 7626, pp. 471–476, Oct. 2016. [Online]. Available: https://www.nature.com/articles/nature20101

[56] D. Hafner, J. Pasukonis, J. Ba, and T. Lillicrap, "Mastering diverse domains through world models," 2023, *arXiv:2301.04104*.

Benjamin Devillers received the Ph.D. degree in learning multimodal representations in neural networks from University Toulouse III–Paul Sabatier, Toulouse, France, in 2022.

He is currently a Post-Doctoral Student with the CerCo—CNRS Laboratory, Toulouse. His current research interests include modeling the cognitive theory of the global workspace to improve multimodal representations of AI models.

Léopold Maytié is currently pursuing the Ph.D. degree in multimodal representations for robotics use cases with University Toulouse III–Paul Sabatier, Toulouse, France.

His work focuses particularly on using the global workspace theory to enhance multimodal representation in robots.

Rufin VanRullen is currently a CNRS Research Director of the Brain and Cognition Research Center (CerCo), Toulouse, France. He also holds a Research Chair at the Artificial and Natural Intelligence Toulouse Institute (ANITI), Toulouse. His work explores the capabilities of advanced AI systems based on neurocognitive architectures.