Par Mohammed Bouayoun, Janvier 2025 (bouyao@gmail.com)

Chapitre 8 : DSL Multimodal : Fusion de la Vision, du Langage et des Sons

8.1. Introduction

L'un des **intérêts majeurs** du Deep Synergy Learning (DSL) réside dans sa capacité à traiter **simultanément** plusieurs types de données (visuelles, textuelles, sonores) et à faire émerger des **synergies** entre ces différentes modalités. Dans les chapitres antérieurs :

- Le **Chapitre 3** a posé les bases de la **représentation** d'entités (vecteurs, symboles, hybrides).
- Le **Chapitre 4** a explicité la **dynamique** auto-organisée (mise à jour ω), tandis que le **Chapitre 5** a présenté l'**architecture** SCN et ses modules.
- Le **Chapitre 6** s'est concentré sur l'**apprentissage multi-échelle** et la fractalité, et le **Chapitre 7** a introduit diverses **méthodes d'optimisation** (recuit, inhibition avancée, etc.).

À présent, nous allons **montrer** comment le DSL peut intégrer et **fusionner** plusieurs canaux de données (images, textes, sons), permettant de dégager une structure **multimodale** riche et auto-organisée.

8.1.1. Contexte

8.1.1.1. Rappel

Dans les chapitres précédents, plusieurs **fondements** ont été présentés afin d'établir la base du **Deep Synergy Learning** (**DSL**) et de son architecture en **Synergistic Connection Network** (**SCN**). Le **Chapitre 3** a notamment abordé la **représentation des entités**, décrivant chaque entité $\mathcal{E}_i$ à l'aide d'un **vecteur** (comme un *embedding* neuronal), d'un **formalisme symbolique** (comme un ensemble de règles logiques), ou d'une **structure hybride** combinant aspects continus et sémantiques symboliques.

Un **DSL multimodal** mobilise ainsi différents **espaces**. L'**espace visuel** repose sur des représentations extraites de réseaux de neurones tels que **ResNet** ou **VGG**, ou encore des descripteurs comme **SIFT**. L'**espace textuel** utilise des embeddings tels que **Word2Vec**, **BERT** ou **GPT**, tandis que l'**espace audio** s'appuie sur des représentations comme les **spectrogrammes**, les **MFCC**, ou des modèles d'apprentissage profond spécialisés.

Dans le **Chapitre 4**, la **dynamique auto-organisée** a été analysée. Les pondérations $\omega_{i,j}$ reliant deux entités $\mathcal{E}_i$ et $\mathcal{E}_j$ évoluent selon la règle

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) \; + \; \eta\big[S(i,j) \; - \; \tau\,\omega_{i,j}(t)\big],$$

où $\eta > 0$ est le **taux d'apprentissage** et $\tau > 0$ est un **facteur de décroissance**. Cette **mise à jour** exploite la **fonction de synergie** $S(i,j)$. Dans un **contexte multimodal**, la **similarité** n'est plus exclusivement un score intra-canal (image–image, texte–texte), mais peut devenir un **score cross-modal** (image–texte, audio–image, etc.), ce qui entraîne une **auto-organisation** plus riche.

Le **Chapitre 5** a présenté l'**architecture SCN**, dont le cœur repose sur la gestion des **pondérations** $\omega_{i,j}$ et l'interaction avec différents **modules** dédiés à l'ajustement des connexions. Parmi eux, on trouve des composants comme le **Module Synergie** et le **Module Inhibition**, qui interviennent pour moduler les interactions en fonction du contexte.

En **multimodal**, ces modules se spécialisent selon les types de données traitées. Un **Module Synergie** dédié à la **vision-langage** évalue $S(\text{image,texte})$, tandis qu'un autre prend en charge $S(\text{audio,image})$. Chaque module applique ainsi une **mesure de similarité** adaptée à son propre **canal**.

En **Chapitre 6**, le concept d'**apprentissage multi-échelle** a été exploré. Les **entités** peuvent se **regrouper** en **clusters** locaux, puis se rassembler en **macro-clusters** représentant des thèmes plus vastes. Dans un **DSL multimodal**, cette **hiérarchie** sert à structurer des **images**, des **textes** et des **signaux audio** autour de **concepts** ou d'**événements** partagés.

Enfin, le **Chapitre 7** a présenté différentes **méthodes d'optimisation** visant à éviter les **minima locaux**, à gérer l'**inhibition** ou la **saturation** et à permettre un **apprentissage continu**. Une fois le caractère **multimodal** intégré, la **dimension** du réseau s'accroît ; il devient plus fréquent que le SCN contienne de multiples **entités** réparties dans plusieurs **canaux** (visions, textes, sons), ce qui requiert d'autant plus de **régulation** et de **compétition** entre liaisons afin de maintenir un **niveau** cohérent de **pondérations** $\omega_{i,j}$.

8.1.1.2. Ici, on se focalise sur la fusion multimodale : comment le DSL peut agréger des données provenant de la vision, du langage et de l'audio pour en extraire des synergies plus riches.

Le **Deep Synergy Learning (DSL)** ne se limite pas à un seul type de données, mais s'adapte naturellement à un traitement **multimodal**, où plusieurs canaux, tels que la **vision**, l'**audio** et le **texte**, sont intégrés dans un **Synergistic Connection Network (SCN)** unique. Cette approche est particulièrement pertinente pour des applications nécessitant la corrélation d'informations **hétérogènes**.

Cela inclut l'**analyse audio-visuelle** d'une scène, où les signaux image et son doivent être reliés, la recherche de correspondances entre une **description textuelle** et des **images** dans les systèmes de vision-langage, ou encore les **assistants conversationnels**, qui exploitent simultanément des flux **vocaux**, **visuels** et **textuels**. Dans ces contextes, le **DSL** se distingue par sa capacité à **fusionner** plusieurs **modalités** de manière naturelle, grâce à sa **dynamique auto-organisée** et à sa **fonction de synergie**.

Dans cette perspective, un **SCN** peut intégrer un **ensemble** d'entités $\{\mathcal{E}_i\}_{i=1}^{n}$ provenant de canaux distincts. Chaque entité $\mathcal{E}_i$ peut correspondre à un **extrait visuel**, un **descripteur audio** ou une **unité textuelle** (mot, token, phrase).

Pour gérer cette **hétérogénéité**, des représentations adaptées sont utilisées. Un objet visuel $\mathcal{E}_i^{(\text{visuel})}$ est encodé sous forme d'un vecteur $\mathbf{x}_i^{(\text{visuel})} \in \mathbb{R}^{d_v}$, issu par exemple d'un **embedding** extrait d'un réseau convolutionnel tel que **ResNet** ou **VGG**. Un objet audio $\mathcal{E}_j^{(\text{audio})}$ est représenté par un vecteur $\mathbf{x}_j^{(\text{audio})} \in \mathbb{R}^{d_a}$, comme des **MFCC** ou un **embedding profond**. Un segment textuel $\mathcal{E}_k^{(\text{texte})}$ est associé à un **embedding** $\mathbf{x}_k^{(\text{texte})} \in \mathbb{R}^{d_t}$, utilisant des modèles comme **Word2Vec**, **GloVe** ou **BERT**.

Bien que ces espaces aient des **dimensions différentes**, le **DSL** permet leur intégration dans un **SCN unique** grâce aux mécanismes d'**extension** et de **construction** de la fonction de synergie (voir **Chapitres 2 et 3**).

La **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ revêt alors plusieurs formes selon que $\mathcal{E}_i$ et $\mathcal{E}_j$ appartiennent ou non au même **canal**. On distingue les **similarités intra-modales**, comme la distance euclidienne inversée ou la similarité cosinus entre deux entités visuelles (image–image) ou deux entités audio (spectrogramme–spectrogramme). On distingue aussi les **corrélations inter-modales** (image–texte, audio–image, texte–audio), qui nécessitent un mécanisme ou un module spécifique pour calculer un score de compatibilité. Sur le plan mathématique, on peut définir une fonction globale

$$S(i,j) = \begin{cases} S_{\text{intra}}\left(\mathbf{x}_i^{(\text{visuel})}, \mathbf{x}_j^{(\text{visuel})}\right), & \text{si } i,j \text{ sont tous deux du canal visuel,} \\ S_{\text{cross}}\left(\mathbf{x}_i^{(\text{visuel})}, \mathbf{x}_j^{(\text{audio})}\right), & \text{si } i,j \text{ appartiennent à des canaux différents,} \\ \ldots & \text{(autres cas : texte–image, texte–audio, etc.).} \end{cases}$$

Chacun de ces termes peut être implémenté par une **distance** ou une **similarité** adaptée (ex. distance cosinus, corrélation basée sur un espace latent partagé, kernel RBF, etc.), d'où l'on dérive un score entre $[0,1]$ ou $\mathbb{R}^+$.

Dans un **SCN** multimodal, la règle de mise à jour,

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) \; + \; \eta\left[S(i,j)\tau\,\omega_{i,j}(t)\right],$$

reste inchangée. Toutefois, la présence de synergies cross-modales multiplie la variété des liens susceptibles d'émerger ou de se renforcer. Si une image $\mathcal{E}_i^{(\text{visuel})}$ montre une scène de conversation et qu'un segment $\mathcal{E}_k^{(\text{audio})}$ capture des voix humaines, on peut observer un **renforcement** progressif du poids $\omega_{i,k}$ dès lors que la corrélation audio–visuelle est élevée (temporellement et sémantiquement). De même, si un **mot** $\mathcal{E}_\ell^{(\text{texte})}$ décrit exactement l'objet visible dans $\mathcal{E}_i^{(\text{visuel})}$, un score de similarité "texte–image" élevé conduira à un $\omega_{i,\ell}$ plus fort. C'est la co-occurrence ou la **cohérence** entre modalités qui guide la consolidation ou l'extinction des liaisons.

Le principal **bénéfice** de cette fusion multimodale consiste à **capturer** des correspondances plus richement sémantiques. Un seul canal, par exemple le canal visuel, ne peut pas nécessairement lever toutes les ambiguïtés (un même type d'objet peut apparaître dans des contextes différents). Mais si, en parallèle, la piste audio ou la piste textuelle donne des indices compatibles (bruit caractéristique de l'objet, ou champ lexical qui décrit la scène), la **dynamique** du DSL va "coller" ces entités ensemble et former des **clusters** multimodaux qui représentent des objets ou des événements plus finement définis. Au niveau mathématique, ce principe correspond à la superposition de scores de synergie provenant de différents canaux, laquelle amplifie les liens réellement fiables et écarte les couplages purement fortuits dans un canal donné.

Un autre aspect fondamental concerne la **constitution** de **macro-clusters** hiérarchiques dans un **DSL** à plusieurs échelles, comme décrit dans le **Chapitre 6**. À un niveau fin, des associations locales émergent entre des éléments spécifiques, qu'il s'agisse d'une **région d'image**, d'un **motif audio** ou d'un **mot** isolé. Progressivement, l'**auto-organisation** regroupe ces sous-ensembles en **sous-réseaux**, encapsulant des concepts ou des événements multimodaux.

Un **macro-cluster** tel que *"conversation téléphonique dans un véhicule"* peut ainsi intégrer un **extrait audio** de voix, le **bruit du moteur**, des **images de passagers** et des **mots-clés** pertinents. La **formation** de ces macro-clusters repose sur l'**héritage** des liens accumulés dans chaque modalité, ce qui permet l'émergence d'une **sémantique globale** à partir des interactions locales.

Sur le plan **algorithmique**, la fusion multimodale dans le **DSL** ne nécessite pas de modification conceptuelle majeure. La règle locale de mise à jour de $\omega_{i,j}$ reste inchangée, avec l'ajout d'une capacité à **calculer** $S(i,j)$ pour des paires (i,j) appartenant à des canaux différents. La structure du réseau évolue ainsi de manière naturelle, faisant **émerger** des liaisons **cross-modales**, parfois renforcées ou filtrées par une règle de **parsimonie** ou d'**inhibition** (voir **Chapitre 7**) afin d'écarter les connexions trop faibles.

L'adaptation repose essentiellement sur le paramétrage des **modules de similarité** ou de **corrélation**, définis pour chaque combinaison de modalités. Cela inclut les relations entre **image et image**, **audio et audio**, **texte et texte**, ainsi que les interactions **cross-modales** telles que **image–audio**, **image–texte** ou **audio–texte**. L'influence relative de chaque modalité peut être ajustée à l'aide de coefficients de pondération spécifiques, comme α_{image}, α_{audio} et α_{texte}, selon l'importance accordée à chaque canal dans l'apprentissage et l'organisation du **SCN**.

8.1.2. Objectifs et Intérêt

Dans le contexte du **DSL** (Deep Synergy Learning) appliqué aux données multimodales (chap. 8), l'objectif est de **fusionner** plusieurs sources d'information — images, textes, sons, capteurs divers — dans un **même** SCN (Synergistic Connection Network). On souhaite non seulement traiter ces modalités séparément, mais surtout **exploiter** la dynamique du DSL pour créer des **liaisons synergiques** entre elles, permettant l'émergence de clusters ou de structures qui reflètent les **corrélations** (ou complémentarités) entre objets visuels, mots-clés, signaux audio, etc. Les **sections** suivantes (8.1.2.2, 8.1.2.3) détailleront comment cette approche peut déboucher sur des **applications** concrètes (annotation automatique, recherche audio-visuelle, etc.).

8.1.2.1. Gérer simultanément plusieurs modalités (image, texte, sons) dans un même SCN

L'un des enjeux majeurs dans les approches multimédias actuelles est de parvenir à **combiner** et à **corréler** plusieurs types de données (images, textes, sons, etc.) au sein d'un même cadre d'apprentissage. Le **Deep Synergy Learning (DSL)**, via sa conception de **Synergistic Connection Network (SCN)**, fournit une structure adaptée à cette fusion, en permettant d'intégrer dans un seul et même réseau des entités issues de canaux hétérogènes. Les pondérations $\omega_{i,j}$ au sein d'un SCN peuvent en effet relier des entités visuelles, des segments audios ou des tokens textuels, s'il s'avère que les **synergies** calculées entre elles sont élevées.

A. Notion de Multi-Modalité et Enjeux

Dans de nombreux scénarios concrets, comme la **classification d'images avec annotation textuelle**, l'**analyse audio-visuelle** ou les **systèmes de recommandation contextuelle**, les entités $\mathcal{E}_i$ proviennent de **canaux variés**. Un ensemble $\{\mathcal{E}_i^{(\text{image})}\}$ représente les caractéristiques visuelles d'un objet ou d'une scène, un ensemble $\{\mathcal{E}_i^{(\text{texte})}\}$ encode des segments linguistiques

ou des mots-clés, et un ensemble $\{\mathcal{E}_i^{(\text{audio})}\}$ décrit des descripteurs sonores sous forme de **spectrogrammes**, de **MFCC** ou d'**embeddings profonds**.

Sur le plan mathématique, chaque modalité est associée à un **espace vectoriel propre**. La composante visuelle est décrite par $\mathbf{x}_i^{(\text{image})} \in \mathbb{R}^{d_{\text{img}}}$, la composante linguistique par $\mathbf{x}_i^{(\text{texte})} \in \mathbb{R}^{d_{\text{txt}}}$ et la partie sonore par $\mathbf{x}_i^{(\text{audio})} \in \mathbb{R}^{d_{\text{aud}}}$.

Le défi pour le **SCN** est d'**intégrer** ces espaces au sein d'un **graphe unique** $\{\omega_{i,j}\}$, où chaque nœud $\mathcal{E}_i$ peut entretenir des **liaisons intra-modales** avec des entités du même type, ou des **liaisons cross-modales** connectant différentes modalités.

Cette fusion multimodale se justifie par le fait que de nombreux **concepts** ou **événements** ont une existence multi-canal (par exemple, une vidéo avec son commentaire textuel et sa piste audio). En traitant l'image, le texte et l'audio dans des systèmes totalement séparés, on risquerait de rater les corrélations qui apparaissent pourtant de manière évidente (le son du piano et l'image d'un clavier, le mot "chien" et la photo d'un chien, etc.). Le **DSL** permet, au contraire, d'**auto-organiser** un réseau unique où chaque lien $\omega_{i,j}$ représente la pertinence d'associer deux entités, même si elles proviennent de modalités différentes.

B. Construction de la Synergie Inter-Modal

Le point délicat réside dans la **définition** de la fonction de synergie $S(\mathcal{E}_i, \mathcal{E}_j)$ lorsqu'on compare des entités issues d'espaces différents (image vs. texte, audio vs. image, texte vs. audio, etc.). Au sein du Chapitre 2 (sections 2.2.1.2 et 2.2.1.3), on a déjà vu comment "coller" des mesures de similarité ou d'information mutuelle sur des ensembles hétérogènes. Concrètement, on introduit souvent un **espace latent partagé** où l'on projette chacune des entités. On définit alors, par exemple,

$$S\left(\mathcal{E}_i^{(\text{image})}, \mathcal{E}_j^{(\text{texte})}\right) \;=\; \text{similarité}\left(\phi_{\text{img}}\left(\mathbf{x}_i^{(\text{image})}\right), \phi_{\text{txt}}\left(\mathbf{x}_j^{(\text{texte})}\right)\right),$$

où ϕ_{img} et ϕ_{txt} sont des fonctions (le plus souvent neuronales) qui permettent de projeter respectivement l'image et le texte dans un espace vectoriel commun, comme $\mathbb{R}^d$. Une fois ce **score** défini, la synergie $S(i,j)$ alimente la mise à jour des pondérations $\omega_{i,j}$. Chaque couple d'entités $\left(\mathcal{E}_i^{(\text{image})}, \mathcal{E}_j^{(\text{texte})}\right)$ voit ainsi sa liaison renforcée si les descripteurs latents sont jugés suffisamment proches ou corrélés.

De la même façon, pour l'audio et l'image, on peut concevoir un score de correspondance temporelle ou sémantique, en appuyant la similarité sur des méthodes de co-occurrence (scènes vidéo–audio) ou d'embeddings partagés. Le **SCN** évolue alors simultanément selon la règle :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) \;-\; \tau\,\omega_{i,j}(t)\big],$$

ce qui autorise la naissance de liens forts entre canaux différents lorsque la corrélation est considérée comme robuste.

C. Intégration dans un SCN Unique et Émergence de Clusters Multimodaux

Une fois la **synergie inter-modale et intra-modale** définie, l'ensemble des entités visuelles, textuelles et audio est **intégré** dans un **même** graphe $\{\omega_{i,j}\}$. Les nœuds $\mathcal{E}_i$ ne sont plus limités

à un seul type de feature, ce qui permet d'inclure simultanément des entités comme $\mathcal{E}_1^{(\text{img})}$, $\mathcal{E}_2^{(\text{txt})}$, $\mathcal{E}_3^{(\text{audio})}$ dans une seule matrice de pondérations $\{\omega_{i,j}\}$.

La dynamique du **DSL** ajuste progressivement les liaisons : les pondérations $\omega_{i,j}$ se renforcent lorsqu'il existe une **grande similarité intra-modale** ou une **forte compatibilité inter-modale**. Ce processus conduit à la **formation de clusters**, où se regroupent des entités de différentes natures partageant un même concept ou renvoyant à un même événement. Ainsi, un **thème multimodal** peut émerger, associant une **image de chien**, un **segment audio d'aboiement** et les mots **"chien"** ou **"dog"**.

Cette **émergence** de **clusters multimodaux** présente des avantages significatifs pour diverses applications. Elle constitue d'abord un levier puissant pour la **recherche d'information**, en permettant à un **mot-clé textuel** de se retrouver dans le même groupe qu'un **extrait sonore** et une **image** thématiquement liée. Elle facilite également la **classification** et l'**annotation** automatiques, en identifiant rapidement qu'un ensemble combinant **image, texte et audio** correspond à un même concept.

Au-delà de ces aspects, cette structuration ouvre la voie à des **scénarios cognitifs avancés** (voir **Chapitre 9**), où un **agent conversationnel** peut exploiter simultanément des **signaux visuels** et des **indices textuels** pour améliorer ses réponses. Le **SCN** joue alors un rôle central en assurant la **correspondance** et la **consolidation** des liens entre ces flux d'information hétérogènes.

D. Problèmes de Complexité et Normalisation

Naturellement, l'intégration de plusieurs modalités fait croître la **taille** du réseau (nombre de nœuds) et, potentiellement, la complexité $O(n^2)$ des connexions $\omega_{i,j}$. Sur le plan algorithmique, il peut donc être nécessaire de **filtrer** en amont les couples d'entités trop éloignés, ou de restreindre la mise à jour aux plus proches voisins, pour éviter la saturation en ressources (voir Chapitre 7, section sur l'équilibre entre densité et parsimonie). De plus, il convient de **normaliser** la fonction de synergie S afin qu'aucune modalité ne domine trop les autres. Dans certains cas, on introduit des coefficients $\alpha_{\text{img}}, \alpha_{\text{txt}}, \alpha_{\text{aud}}$ pour pondérer l'influence de chaque canal ; on peut également calibrer les échelles de similarité (entraînement d'un réseau contrastif, usage de kernel RBF, etc.) pour harmoniser le traitement de chaque couple modal.

8.1.2.2. Utiliser la dynamique DSL pour faire émerger des clusters ou liens synergiques entre modalités (objets visuels, mots, sons)

Lorsque l'on traite simultanément des données **visuelles**, **textuelles** et **sonores**, l'enjeu consiste à **faire ressortir** les correspondances et les regroupements les plus pertinents à travers ces trois canaux. Le **Deep Synergy Learning (DSL)**, grâce à sa mécanique d'**auto-organisation** et à sa **fonction de synergie** appliquée à un **Synergistic Connection Network (SCN)** unique, offre une **méthode** systématique pour encourager la **création** de liens cohérents entre des éléments multimodaux et, de ce fait, **faire émerger** des **clusters** sémantiquement riches (ex. relier l'image d'un chat, le mot "chat" et un son de miaulement).

A. Principes Généraux : Dynamique DSL en Contexte Multimodal

Dans un environnement **multimodal**, on définit un ensemble $\{\mathcal{E}_i\}_{i=1}^{n}$ rassemblant toutes les entités possibles :

- Des **objets visuels** (ou régions d'images) annotés ou détectés,
- Des **tokens** ou **segments textuels** (mots, chunks de phrases, etc.),
- Des **caractéristiques** ou **extraits** sonores (spectrogrammes, MFCC, événements acoustiques, etc.).

Ces entités cohabitent dans un **même** SCN, où chaque nœud $\mathcal{E}_i$ possède une pondération $\omega_{i,j}$ avec chaque autre nœud $\mathcal{E}_j$. Le nombre total d'entités n peut être élevé, rendant critique la question de la **complexité** (voir Chapitre 7 pour les techniques de parsimonie et de filtrage).

Pour calculer la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$, on prend en compte la **nature** de $\mathcal{E}_i$ et $\mathcal{E}_j$. Quand on compare deux entités **visuelles**, on utilise une similarité intra-canal (distance cosinus, distance euclidienne inversée sur les embeddings visuels, etc.). Quand on compare un objet visuel à un **mot**, ou un segment audio à un **mot**, on recourt à des méthodes "cross-modales" (ex. projection dans un espace latent partagé, ou évaluation de co-occurrences temporelles).

$$S(\mathcal{E}_i, \mathcal{E}_j) = \begin{cases} \text{similarité\_visuelle}\left(\mathbf{x}_i^{(\text{visuel})}, \mathbf{x}_j^{(\text{visuel})}\right), & \text{(intra-image)} \\ \text{similarité\_cross}\left(\mathbf{x}_i^{(\text{visuel})}, \mathbf{x}_j^{(\text{texte})}\right), & \text{(image–texte)} \\ \text{co-occurrence\_audio}\left(\mathbf{x}_i^{(\text{son})}, \mathbf{x}_j^{(\text{texte})}\right), & \text{(audio–texte)} \\ \ldots \end{cases}$$

Une fois $S(i,j)$ défini, on l'emploie pour la mise à jour de $\omega_{i,j}$.

La **dynamique** auto-organisée du DSL s'exprime par la règle habituelle :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) \; + \; \eta\big[S(i,j) \; - \; \tau\, \omega_{i,j}(t)\big].$$

Ainsi, si la **cohérence** (ou co-occurrence) entre $\mathcal{E}_i$ et $\mathcal{E}_j$ est élevée (i.e. $S(i,j)$ grand), leur liaison $\omega_{i,j}$ aura tendance à **se renforcer** et, au fil des itérations, à devenir un lien stable du réseau. À l'inverse, si la corrélation inter- ou intra-modale reste faible, la liaison $\omega_{i,j}$ **décroît** progressivement.

B. Formalisation Mathématique Simplifiée

Cette section explore une extension du **Deep Synergy Learning (DSL)** aux données **multimodales**, en intégrant trois catégories d'entités : **visuelles**, **textuelles** et **sonores**. Cette approche s'appuie sur les principes établis en **section 2.2.1**, qui définit les entités et la fonction de synergie, ainsi qu'en **section 2.2.2**, qui décrit la mise à jour des pondérations ω. Elle précise comment un **SCN** (Synergistic Connection Network) peut gérer ces différentes modalités au sein d'un **même espace d'apprentissage auto-organisé**, favorisant ainsi une intégration cohérente des informations issues de plusieurs sources.

Pour introduire la pluralité des modalités, on se donne trois ensembles :

$$\mathcal{V} = \{\mathcal{V}_1, \ldots, \mathcal{V}_p\} \quad \text{(entités visuelles)}, \qquad \mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_q\} \quad \text{(entités textuelles)}, \qquad \mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_r\} \quad \text{(entités sonores)}.$$

Le **réseau** complet de n entités, noté $\mathcal{E} = \mathcal{V} \cup \mathcal{T} \cup \mathcal{S}$, satisfait ainsi $n = p + q + r$. Chaque entité $\mathcal{E}_i$ appartient à l'une de ces trois modalités. On représente ensuite les **pondérations** par

une matrice ω de taille $n \times n$, dont chaque terme $\omega_{i,j}$ indique la **force** de la liaison entre $\mathcal{E}_i$ et $\mathcal{E}_j$. La mise à jour de $\omega_{i,j}$ suit la **règle locale** :

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\big[S\big(\mathcal{E}_i, \mathcal{E}_j\big) - \tau\, \omega_{i,j}(t)\big],$$

conformément aux principes du **DSL** décrits en **section 2.2.2**. La **fonction de synergie** $S\big(\mathcal{E}_i, \mathcal{E}_j\big)$ doit alors être définie pour chaque couple (i, j) selon la nature (modalité) des entités.

Lorsque $\mathcal{E}_i = \mathcal{V}_a$ est une entité visuelle et $\mathcal{E}_j = \mathcal{T}_b$ est une entité textuelle, on peut s'appuyer sur un **embedding** commun (par exemple ϕ_{img} pour les images et ϕ_{txt} pour les textes) afin de mesurer la similarité :

$$S(\mathcal{V}_a, \mathcal{T}_b) \;=\; \cos\Big(\phi_{\text{img}}(\mathbf{v}_a),\, \phi_{\text{txt}}(\mathbf{t}_b)\Big).$$

Cette formule évalue le **cosinus** de l'angle entre deux **vecteurs latents**, mesurant ainsi la correspondance sémantique entre l'image $\mathbf{v}_a$ et le texte $\mathbf{t}_b$. Une alternative consiste à utiliser une **fonction de coïncidence**. Lorsqu'un large corpus fournit des **fréquences conjointes d'apparition** entre un objet visuel $\mathcal{V}_a$ et un mot $\mathcal{T}_b$, la synergie peut être définie comme

$$S(\mathcal{V}_a, \mathcal{T}_b) = \text{freq\_cooc}(\mathcal{V}_a, \mathcal{T}_b),$$

ou à partir d'un score dérivé de cette **cooccurrence**, comme une **probabilité normalisée**, permettant d'affiner l'estimation du lien entre les deux modalités.

Dans le cas d'une **entité sonore** $\mathcal{S}_c$, l'évaluation de $S(\mathcal{S}_c, \mathcal{T}_b)$ ou $S(\mathcal{S}_c, \mathcal{V}_a)$ peut faire appel à des **corrélations temporelles** ou à des modèles d'alignement entre audio et texte ou audio et image. Par exemple, si un mot « cat » est identifié à l'instant t dans un flux textuel associé à la vidéo ou à la scène, l'extrait audio $\mathcal{S}_c$ couvrant l'intervalle $[\; t - \delta,\; t + \delta \;]$ est davantage susceptible de présenter une synergie élevée avec ce mot, ce qui se reflète dans $\omega_{i,j}$ si $\mathcal{S}_c$ et $\mathcal{T}_b$ coïncident fréquemment sur ce type de segments.

À mesure que l'on incrémente t, la pondération $\omega_{i,j}(t)$ évolue jusqu'à se rapprocher d'un état stable $\omega_{i,j}^*$. Les entités $\big(\mathcal{E}_i, \mathcal{E}_j\big)$ qui présentent une corrélation sémantique (ou statistique) élevée et répétée voient leurs liens se renforcer ; les paires moins cohérentes ou rarement associées aboutissent à des pondérations proches de zéro, sous l'effet de la **décroissance** $\tau\, \omega_{i,j}(t)$ et de la **parsimonie** (voir **section 2.2.3**).

Dans ce **SCN** multimodal, on observe in fine l'émergence de **clusters** où se regroupent plusieurs **objets visuels** $\mathcal{V}_a$, **concepts textuels** $\mathcal{T}_b$ et éventuellement **segments audio** $\mathcal{S}_c$, tous reliés par des liens $\omega_{i,j}$ élevés. Ces **clusters** peuvent être interprétés comme des "thèmes" ou "concepts" reliant plusieurs modalités de données, formant ainsi des sous-ensembles d'entités fortement synergiques, analogues à ceux décrits en **section 2.2.5** pour le cas purement unimodal.

C. Avantages : Émergence de Clusters Inter-Modalités

L'un des points forts de cette approche réside dans son **auto-organisation**. Contrairement à un modèle nécessitant des **labels externes** ou un alignement imposé de type "cette image illustre ce mot", le SCN multimodal laisse la dynamique DSL révéler les **associations** latentes à partir des co-occurrences ou des similarités vectorielles. Ainsi, si un objet visuel $\mathcal{V}_a$ apparaît systématiquement dans un corpus avec un mot $\mathcal{T}_b$, la règle de mise à jour fera grimper $\omega_{a,b}$. Dès lors que l'on ajoute la composante audio $\mathcal{S}_c$ et que celle-ci se révèle régulièrement liée à la

même scène ou au même mot, un “triplet” $(\mathcal{V}_a, \mathcal{T}_b, \mathcal{S}_c)$ peut se renforcer, signant la mise en place d’un **cluster** tri-modal.

En permettant à chaque entité $\mathcal{E}_i$ de se connecter librement à toutes les autres (qu’elles soient images, textes ou sons), le **SCN** favorise la formation de **micro-clusters** reliant quelques images spécifiques, un groupe de termes pertinents et certains signaux audio spécifiques. À plus grande échelle (voir **chapitre 6** si la référence renvoie à un traitement plus approfondi), on voit aussi des **macro-clusters** se constituer en agrégeant plusieurs de ces micro-clusters autour d’un thème élargi.

Les **conséquences pratiques** sont multiples. Une fois cette structure auto-organisée établie, on peut réaliser :

– de la **recherche** ou de la **recommandation** cross-modale, en identifiant les images “proches” d’un mot ou les mots les plus fortement reliés à un extrait audio ;

– de l’**annotation** automatique en sélectionnant les termes textuels qui apparaissent dans le cluster d’une image donnée ;

– un **système cognitif** multimodal pouvant rapidement basculer d’une modalité à l’autre (texte, image, audio) pour décrire, identifier ou comparer un contenu.

L’émergence de clusters enrichis sur le plan sémantique peut ainsi servir de base à des applications d’**indexation**, de **filtrage** ou de **fusion** de données, répondant au besoin de manipuler des informations hétérogènes sans imposer a priori un alignement rigide ou des labels manuels sur chaque entité. Cette adaptabilité vient directement de la **dynamique** du DSL, qui combine les notions de synergie, de compétition (inhibition) et de recuit potentiel (chap. 7.3), maintenant dans un même cadre la plasticité nécessaire à la mise en correspondance de modalités distinctes.

8.1.2.3. Montrer des applications (ex. annotation d’images, reconnaissance audio-visuelle, systèmes multimédias)

Une fois que l’on a posé les bases théoriques (sections 8.1.2.1 et 8.1.2.2) et qu’on comprend comment des entités de diverses modalités (visuel, audio, texte) peuvent interagir dans un **Synergistic Connection Network (SCN)** commun, il devient naturel d’explorer des **applications** mettant à profit cette **dynamique**. Le **Deep Synergy Learning (DSL)** se prête en effet à plusieurs cas d’usage, allant de la **classification** et **annotation** d’images jusqu’aux **systèmes multimédias** plus complexes, en passant par la **reconnaissance audio-visuelle**. Nous illustrerons ci-après trois domaines dans lesquels la synergie **inter-modale** et la **mise à jour** automatique des liaisons $\omega_{i,j}$ font la différence.

A. Annotation d’Images

L’annotation d’images vise à associer une **image** ou une **région d’image** à un ou plusieurs **mots-clés** décrivant son contenu. Contrairement aux approches classiques basées sur des classificateurs ou des modèles de détection d’objets spécifiques (reconnaissance de chat, voiture, bâtiment, etc.), la dynamique **DSL** permet une structuration plus **générale et auto-organisée** des relations entre les entités visuelles et textuelles.

Dans ce cadre, les **entités visuelles** $\{\mathcal{E}_i^{(\text{img})}\}$ correspondent aux images globales ou à des “patches”/“bounding boxes” spécifiques, tandis que les **entités textuelles** $\{\mathcal{E}_j^{(\text{txt})}\}$ représentent

des mots, phrases ou concepts associés (ex. “chat”, “ciel bleu”, “bâtiment”). L’ensemble des interactions est encapsulé dans un **SCN (Synergistic Connection Network)**, où la **synergie**

$$S\left(\mathcal{E}_i^{(\text{img})}, \mathcal{E}_j^{(\text{txt})}\right)$$

exprime la **compatibilité** entre une représentation visuelle (issue d’un CNN) et une représentation sémantique (obtenue par Word2Vec, GloVe, BERT, etc.). Une formulation typique est la similarité cosinus entre leurs **embeddings** respectifs :

$$S\left(\mathcal{E}_i^{(\text{img})}, \mathcal{E}_j^{(\text{txt})}\right) = \cos\left(\phi_{\text{vis}}(\mathbf{x}_i),\, \phi_{\text{txt}}(\mathbf{x}_j)\right),$$

où ϕ_{vis} et ϕ_{txt} sont des projections dans un espace latent partagé.

L’évolution des **liaisons** $\omega_{i,j}$ est gouvernée par la mise à jour **DSL** :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right].$$

Au fil des itérations, les **associations pertinentes** se renforcent, favorisant l’émergence de clusters reliant des images à des mots-clés. Une image contenant un chat développera progressivement une liaison forte avec le mot “chat”, tandis qu’une autre, représentant un chien, établira une connexion robuste avec “chien”.

Plutôt que de s’appuyer uniquement sur des annotations manuelles, le SCN **auto-organise** ces relations, permettant d’identifier des **groupements plus larges** :

- Une collection d’images de félins pourrait être associée aux mots “cat”, “lion”, “tiger”, créant un cluster cohérent.
- Un **macro-cluster** plus large (voir Chapitre 6) peut émerger, regroupant toutes les images liées à la faune et leurs descriptions textuelles associées.

Cette approche favorise une **adaptabilité** accrue, facilitant l’**insertion dynamique** de nouvelles images ou mots sans nécessiter d’apprentissage supervisé supplémentaire. De nouvelles entités s’intègrent naturellement, ajustant leurs connexions ω avec les structures existantes, en cohérence avec la logique d’**apprentissage continu** (voir Chapitre 7.6).

L’annotation d’images basée sur le **SCN** présente plusieurs **bénéfices** majeurs :

- **Robustesse et Correction d’Erreurs** : L’**auto-organisation** du réseau compense les imperfections d’un modèle de reconnaissance. Si un mot-clé est **pertinent** pour un sous-ensemble d’images, cette influence structurelle **attire** progressivement les connexions ω, même si l’algorithme de détection initial a omis certaines instances.
- **Recherche Cross-Modale** : L’interrogation du **SCN** devient simple. Il suffit de demander *“Quelles images sont les plus liées au mot ‘chat’ ?”* en utilisant directement les pondérations $\omega_{i,j}$, sans entraîner de modèle dédié pour chaque tâche.
- **Scalabilité et Adaptabilité** : Le réseau croît de manière **incrémentale** au fur et à mesure que de nouvelles entités sont ajoutées. Grâce aux mécanismes de **parsimonie** (filtrage des liens faibles, régularisation par seuil $\omega_{\min}$), la taille du SCN reste maîtrisée, garantissant une **structure efficace** et optimisée pour la recherche et l’interprétation des relations image-texte.

B. Reconnaissance Audio-Visuelle

La **reconnaissance** audio-visuelle constitue un cas particulier de **Deep Synergy Learning (DSL)** appliqué à des flux multimodaux où coexistent des données vidéo et des données audio. L'objectif, par exemple, est de détecter et de caractériser des **événements** dans une séquence vidéo, en faisant correspondre chaque **entité** visuelle (souvent un ensemble de frames ou de features extraites) à des **entités** audio (segments sonores ou embeddings).

Pour formaliser cette approche, on suppose l'existence d'un **SCN** comprenant deux ensembles d'entités : $\{\mathcal{E}_i^{(\text{vis})}\}$ pour la partie **visuelle** et $\{\mathcal{E}_j^{(\text{aud})}\}$ pour la partie **audio**. Chaque lien $\omega_{i,j}$ symbolise la force de la liaison entre $\mathcal{E}_i^{(\text{vis})}$ et $\mathcal{E}_j^{(\text{aud})}$. Conformément à la **section 2.2.2**, la mise à jour de $\omega_{i,j}$ se décrit par

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\left[S\left(\mathcal{E}_i^{(\text{vis})}, \mathcal{E}_j^{(\text{aud})}\right) \;-\; \tau\,\omega_{i,j}(t)\right],$$

où η est le taux d'apprentissage, τ un coefficient de décroissance, et $S(\cdot,\cdot)$ la **synergie** reliant l'entité visuelle $\mathcal{E}_i^{(\text{vis})}$ à l'entité audio $\mathcal{E}_j^{(\text{aud})}$.

La **fonction** S peut reposer sur une **coïncidence temporelle**, par exemple lorsqu'on souhaite aligner un segment audio $[\, t-\delta,\; t+\delta \,]$ à la frame vidéo correspondant à l'instant t. On peut également définir un **embedding** cross-modal, ϕ_{vis} et ϕ_{aud}, chacun projetant les données visuelles et sonores dans un espace latent partagé. Dans ce dernier cas, la synergie peut s'écrire sous forme de cosinus :

$$S\left(\mathcal{E}_i^{(\text{vis})}, \mathcal{E}_j^{(\text{aud})}\right) \;=\; \cos\left(\phi_{\text{vis}}(\mathbf{v}_i),\, \phi_{\text{aud}}(\mathbf{a}_j)\right).$$

Ces deux types de définitions (temporelle ou sémantique) ne sont d'ailleurs pas exclusifs et peuvent être combinés pour une robustesse accrue.

Au fil des itérations, la **descente** des pondérations $\omega_{i,j}$ tend à renforcer les liens qui révèlent une correspondance répétée ou stable, et à faire décroître ceux qui n'apportent pas de synergie suffisante. En s'appuyant sur la **parsimonie** décrite en **section 2.2.3**, on élimine les liaisons trop faibles qui ne contribuent guère à la structure globale, ce qui met en évidence des **clusters** audio-visuels d'autant plus nets.

Ces **clusters** représentent des événements ou des scènes dans la séquence vidéo, où un ensemble de frames $\mathcal{E}_i^{(\text{vis})}$ s'avère relié à un ensemble de segments audio $\mathcal{E}_j^{(\text{aud})}$. Par exemple, lors d'un match de football, l'instant du but associe une vue du stade et des joueurs célébrant (frames vidéo), ainsi qu'une montée de la clameur de la foule (segment audio) ou le commentaire spécifique du présentateur. De même, un "cluster concert" peut regrouper des images de musiciens sur scène et des extraits sonores contenant guitare ou batterie.

Les **applications** pratiques sont diverses. En **indexation** de vidéos, on cherche souvent à localiser automatiquement le début et la fin d'un événement particulier. Les liens $\omega_{i,j}$ élevés dans le **SCN** signalent la cooccurrence forte de frames et de segments audio, ce qui permet de pointer rapidement où se situe l'action. En **robotique**, un robot muni de capteurs auditifs et visuels peut, grâce à la dynamique DSL, associer un son spécifique à une scène visuelle, localisant par exemple un objet qui émet un signal sonore. Par ailleurs, sur le plan de l'**accessibilité**, détecter la synchronisation audio-visuelle (par exemple, la parole correspond à telle personne filmée) ouvre la porte à la **création** de légendes ou de sous-titres automatiques en temps réel.

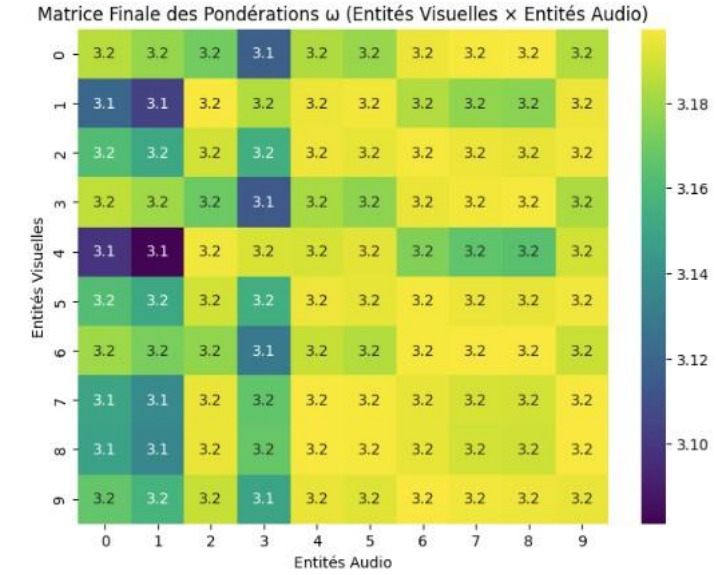
Pour éviter le risque de stagner dans des **minima locaux**, on peut employer un recuit simulé (voir chap. 7.3) qui injecte du bruit dans la mise à jour des pondérations, augmentant la probabilité de franchir des barrières d'énergie. L'**inhibition adaptative** (chap. 7.4) offre un mécanisme complémentaire en limitant la prolifération de liens au sein d'un nœud donné, ce qui favorise une représentation plus éparse et des clusters plus clairement séparés. L'aboutissement est une **organisation** audio-visuelle où chaque événement, scène ou action est identifié par un regroupement de frames et de segments sonores, rendant la reconnaissance des temps forts (but, concert, interview, etc.) plus aisée et plus explicite au sein du **réseau**.

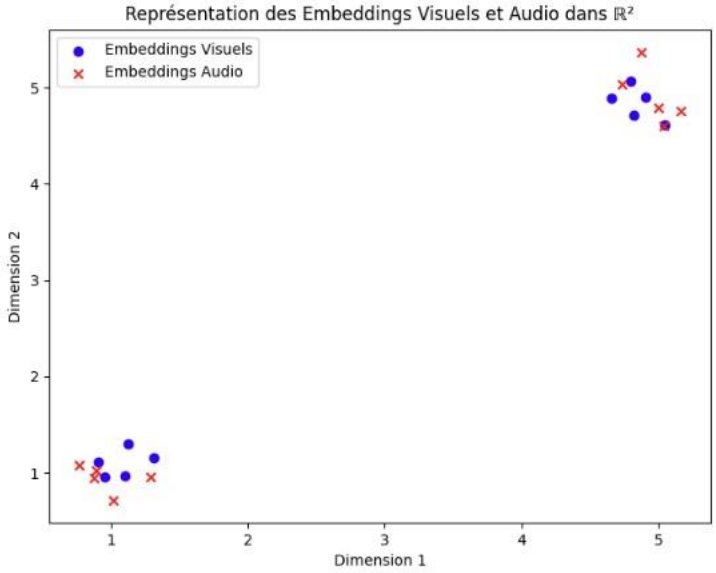
Programme Python 1

Voici ci-dessous un programme Python complet qui simule, de manière concrète, le concept de **reconnaissance audio-visuelle** dans un **Deep Synergy Learning (DSL)**. Le programme s'appuie sur une simulation de deux flux de données – un flux d'**entités visuelles** et un flux d'**entités audio** – et sur la mise à jour des pondérations dans un **Synergistic Connection Network (SCN)** à l'aide d'une fonction de **synergie** $S(i,j)$. Le programme est écrit dans un style académique et structuré, intégrant des formules mathématiques et des explications détaillées.

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Fixer une graine aléatoire pour assurer la reproductibilité
np.random.seed(42)

# Paramètres généraux
eta = 0.1       # Taux d'apprentissage local
tau = 0.2       # Coefficient de décroissance
num_iterations = 50  # Nombre d'itérations de la dynamique

# Simulation de deux groupes d'entités correspondant à deux événements distincts.
# Le groupe 1 (par exemple, un concert) aura des entités visuelles et audio centrées autour de (1, 1)
# Le groupe 2 (par exemple, un match de football) aura des entités centrées autour de (5, 5).

# Générer des embeddings visuels pour le groupe 1 et groupe 2
num_video_group1 = 5
num_video_group2 = 5
video_embeddings = np.vstack([
    np.random.randn(num_video_group1, 2) * 0.2 + np.array([1, 1]),
    np.random.randn(num_video_group2, 2) * 0.2 + np.array([5, 5])
])

# Générer des embeddings audio pour le groupe 1 et groupe 2
num_audio_group1 = 5
num_audio_group2 = 5
audio_embeddings = np.vstack([
    np.random.randn(num_audio_group1, 2) * 0.2 + np.array([1, 1]),
    np.random.randn(num_audio_group2, 2) * 0.2 + np.array([5, 5])
])
```

```
# Déterminer le nombre total d'entités pour chaque modalité
num_video = video_embeddings.shape[0]
num_audio = audio_embeddings.shape[0]

# Initialiser la matrice des pondérations, notée omega, de dimension (num_video, num_audio)
omega = np.full((num_video, num_audio), 0.05)

# Définir la fonction de synergie S(i, j)
# Ici, nous utilisons la similarité cosinus comme mesure de synergie.
def cosine_similarity(vec1, vec2):
    norm1 = np.linalg.norm(vec1)
    norm2 = np.linalg.norm(vec2)
    if norm1 == 0 or norm2 == 0:
        return 0.0
    return np.dot(vec1, vec2) / (norm1 * norm2)

def synergy(i, j):
    # La synergie entre l'entité visuelle i et l'entité audio j est évaluée via la similarité cosinus.
    # On s'assure de renvoyer une valeur positive (bornée dans [0,1]).
    sim = cosine_similarity(video_embeddings[i], audio_embeddings[j])
    return max(sim, 0)

# Simulation de la dynamique des pondérations
# La mise à jour suit la règle :
#   ω_(i,j)(t+1) = ω_(i,j)(t) + η [ S(i,j) - τ ω_(i,j)(t) ]
for t in range(num_iterations):
    for i in range(num_video):
        for j in range(num_audio):
            S_ij = synergy(i, j)
            omega[i, j] = omega[i, j] + eta * (S_ij - tau * omega[i, j])
    # Optionnel : affichage périodique de la matrice de pondérations
    if (t + 1) % 10 == 0:
        print(f"Iteration {t + 1} : omega =\n{np.round(omega, 3)}\n")

# Visualiser la matrice finale des pondérations sous forme de heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(omega, annot=True, cmap="viridis")
plt.title("Matrice Finale des Pondérations ω (Entités Visuelles × Entités Audio)")
plt.xlabel("Entités Audio")
plt.ylabel("Entités Visuelles")
plt.show()

# Visualiser les embeddings dans l'espace 2D pour illustrer les clusters
plt.figure(figsize=(8, 6))
plt.scatter(video_embeddings[:, 0], video_embeddings[:, 1], color="blue", label="Embeddings Visuels
")
plt.scatter(audio_embeddings[:, 0], audio_embeddings[:, 1], color="red", marker="x", label="Embeddi
ngs Audio")
plt.title("Représentation des Embeddings Visuels et Audio dans ℝ²")
plt.xlabel("Dimension 1")
plt.ylabel("Dimension 2")
plt.legend()
plt.show()
```





Dans ce **programme**, nous avons simulé deux groupes d'entités correspondant à deux types d'événements distincts, à l'instar d'un **cluster "concert"** et d'un **cluster "sport football"**. Les **embeddings visuels** (représentés par des vecteurs dans $\mathbb{R}^2$) et les **embeddings audio** sont générés à partir de distributions gaussiennes centrées respectivement autour de points caractéristiques (par exemple, $(1,1)$ pour le premier groupe et $(5,5)$ pour le second groupe).

La **fonction de synergie** $S(i,j)$ utilisée ici se fonde sur la **similarité cosinus**, ce qui permet de mesurer la proximité angulaire entre un vecteur visuel et un vecteur audio. Le principe mathématique est le suivant : pour deux vecteurs $\mathbf{x}_i$ et $\mathbf{x}_j$ dans $\mathbb{R}^d$, la similarité cosinus se définit par

$$\mathrm{sim}(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\| \mathbf{x}_i \| \| \mathbf{x}_j \|},$$

et est bornée entre -1 et 1. Dans notre simulation, nous nous assurons que $S(i,j) \geq 0$ en utilisant la fonction $\max\{0, \cdot\}$.

La **mise à jour** des pondérations $\omega_{i,j}$ est réalisée de manière itérative suivant la règle

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big],$$

où η et τ sont respectivement le **taux d'apprentissage** et le **coefficient de décroissance**. Cette formule combine deux aspects clés. Le **renforcement** de la connexion est assuré par un terme proportionnel à la synergie, tandis que la **décroissance** limite l'augmentation indéfinie des pondérations. En théorie, si la mise à jour était poursuivie à l'infini sans interruption, chaque $\omega_{i,j}$ convergerait vers un **équilibre** approximatif $\omega_{i,j}^{*} \approx S(i,j)/\tau$.

Les **simulations** réalisées sur un petit ensemble d'entités permettent d'observer que les pondérations se renforcent pour les paires dont les vecteurs sont proches (ou bien alignés) et restent faibles pour celles dont la synergie est faible. Ce phénomène conduit à l'**émergence** de **clusters** – par exemple, des entités visuelles et audio associées à un concert se regroupent et se distinguent d'un groupe lié à un match de football.

Enfin, nous avons inclus une visualisation de la **matrice** des pondérations ainsi que des **embeddings** dans l'espace $\mathbb{R}^2$. Ces graphiques aident à interpréter le processus de **clustering auto-organisé** et confirment l'efficacité de la **mise à jour locale** dans le cadre du DSL.

Ce programme constitue ainsi une **illustration** concrète du mécanisme de **Deep Synergy Learning** appliqué à un problème de **reconnaissance audio-visuelle**. Les applications potentielles, telles que l'indexation de vidéos, la robotique ou l'accessibilité, peuvent bénéficier de cette approche en permettant de détecter automatiquement des événements en regroupant les flux d'information visuels et audio au sein d'un même **SCN**.

Programme Python 2

Voici une version modifiée du programme qui résout le problème de dimensions incompatibles entre les embeddings visuels (de dimension 1280) et les embeddings audio (de dimension 13). Pour permettre le calcul de la **similarité cosinus** entre ces deux types d'entités, nous utilisons une projection linéaire aléatoire fixe qui mappe les features audio dans un espace de dimension 1280. Cette transformation assure que les deux vecteurs à comparer appartiennent au même espace vectoriel, et permet ainsi de calculer la similarité.

Le code ci-dessous est entièrement commenté et rédigé dans un style académique :

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import subprocess
import io
import soundfile as sf
import librosa
import librosa.display
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
```

```
from tensorflow.keras.preprocessing import image

# Pour assurer la reproductibilité
np.random.seed(42)

##########################################
# Section 1 : Extraction des Features Vidéo et Audio
##########################################

# 1.1. Extraction des frames vidéo à l'aide d'OpenCV
video_path = "video.mp4"
cap = cv2.VideoCapture(video_path)
if not cap.isOpened():
    raise IOError("Erreur lors de l'ouverture du fichier vidéo. Vérifiez que 'video.mp4' existe et est acce
ssible.")

total_frames = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
fps = cap.get(cv2.CAP_PROP_FPS)
duration_video = total_frames / fps

# On choisit d'extraire nb_frames frames de façon équidistante
nb_frames = 10
frame_indices = np.linspace(0, total_frames - 1, nb_frames, dtype=int)

video_frames = []
current_frame = 0
extracted = 0
while cap.isOpened() and extracted < nb_frames:
    ret, frame = cap.read()
    if not ret:
        break
    if current_frame in frame_indices:
        frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
        video_frames.append(frame_rgb)
        extracted += 1
    current_frame += 1
cap.release()

# 1.2. Extraction des embeddings visuels avec MobileNetV2
# Charger MobileNetV2 pré-entraîné (sans couche de classification et avec pooling global)
model = MobileNetV2(weights="imagenet", include_top=False, pooling='avg', input_shape=(224, 224
,3))
visual_features = []
for frame in video_frames:
    frame_resized = cv2.resize(frame, (224, 224))
    x = image.img_to_array(frame_resized)
    x = np.expand_dims(x, axis=0)
    x = preprocess_input(x)
    features = model.predict(x)
    visual_features.append(features.flatten())
visual_features = np.array(visual_features)  # Dimensions : (nb_frames, 1280)

# 1.3. Extraction de l'audio depuis video.mp4 via ffmpeg et subprocess
# La commande ffmpeg suivante extrait l'audio en format WAV (mono, 22050 Hz)
command = [
```

```
    'ffmpeg',
    '-i', video_path,
    '-f', 'wav',
    '-acodec', 'pcm_s16le',
    '-ac', '1',
    '-ar', '22050',
    'pipe:1'
]
try:
    result = subprocess.run(command, stdout=subprocess.PIPE, stderr=subprocess.PIPE, check=True)
except FileNotFoundError:
    raise FileNotFoundError("ffmpeg n'est pas installé ou n'est pas dans le PATH. Veuillez l'installer p
our exécuter ce programme.")
except subprocess.CalledProcessError as e:
    print("Erreur lors de l'extraction audio :", e.stderr.decode())
    raise e

audio_buffer = io.BytesIO(result.stdout)
audio_signal, sr = sf.read(audio_buffer)
if audio_signal.ndim > 1:
    audio_signal = np.mean(audio_signal, axis=1)
duration_audio = len(audio_signal) / sr

# Diviser l'audio en nb_frames segments (un segment par frame)
segment_duration = duration_audio / nb_frames
audio_features = []
for i in range(nb_frames):
    start_sample = int(i * segment_duration * sr)
    end_sample = int((i + 1) * segment_duration * sr)
    segment = audio_signal[start_sample:end_sample]
    mfcc = librosa.feature.mfcc(y=segment, sr=sr, n_mfcc=13)
    mfcc_mean = np.mean(mfcc, axis=1)
    audio_features.append(mfcc_mean)
audio_features = np.array(audio_features)  # Dimensions : (nb_frames, 13)

##########################################
# Section 2 : Projection des Features Audio
##########################################

# Pour calculer la similarité cosinus entre des vecteurs de dimensions différentes,
# nous projetons les features audio de dimension 13 dans un espace de dimension 1280.
# Nous utilisons une projection linéaire aléatoire fixe.
proj_matrix = np.random.randn(13, 1280)  # Matrice de projection de dimension (13, 1280)
audio_features_projected = np.dot(audio_features, proj_matrix)  # Dimensions : (nb_frames, 1280)

##########################################
# Section 3 : Définition de la Fonction de Synergie
##########################################

def cosine_similarity(vec1, vec2):
    norm1 = np.linalg.norm(vec1)
    norm2 = np.linalg.norm(vec2)
    if norm1 == 0 or norm2 == 0:
        return 0.0
    return np.dot(vec1, vec2) / (norm1 * norm2)
```

```
def synergy(i, j):
    # Calculer la similarité cosinus entre l'embedding visuel (dimension 1280) et
    # l'embedding audio projeté (dimension 1280) pour obtenir un score dans [0,1].
    sim = cosine_similarity(visual_features[i], audio_features_projected[j])
    return max(sim, 0)

# Construire la matrice de synergie S_matrix de dimension (nb_frames, nb_frames)
S_matrix = np.zeros((nb_frames, nb_frames))
for i in range(nb_frames):
    for j in range(nb_frames):
        S_matrix[i, j] = synergy(i, j)

##########################################
# Section 4 : Simulation de la Dynamique du SCN
##########################################

# Initialiser la matrice des pondérations ω avec une valeur initiale faible
omega = np.full((nb_frames, nb_frames), 0.05)

eta = 0.1   # Taux d'apprentissage
tau = 0.2   # Coefficient de décroissance
num_iterations = 50

# Appliquer la règle de mise à jour du DSL :
# ω(i,j)(t+1) = ω(i,j)(t) + η [ S(i,j) - τ ω(i,j)(t) ]
for t in range(num_iterations):
    for i in range(nb_frames):
        for j in range(nb_frames):
            omega[i, j] += eta * (S_matrix[i, j] - tau * omega[i, j])
    if (t + 1) % 10 == 0:
        print(f"Iteration {t+1}:\n{np.round(omega, 3)}\n")

##########################################
# Section 5 : Visualisation des Résultats
##########################################

# Visualisation de la matrice finale des pondérations sous forme de heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(omega, annot=True, cmap="viridis")
plt.title("Matrice Finale des Pondérations ω (Visuel × Audio)")
plt.xlabel("Indices Audio")
plt.ylabel("Indices Visuels")
plt.show()

# Projection 2D des embeddings pour une intuition sur leur distribution
plt.figure(figsize=(8, 6))
plt.scatter(visual_features[:, 0], visual_features[:, 1], color="blue", label="Features Visuelles")
plt.scatter(audio_features_projected[:, 0], audio_features_projected[:, 1], color="red", marker="x", lab
el="Features Audio (Proj.)")
plt.title("Projection 2D des Features Visuelles et Audio Projétées")
plt.xlabel("Dimension 1")
plt.ylabel("Dimension 2")
plt.legend()
plt.show()
```

Dans ce programme, nous traitons un fichier vidéo unique (*video.mp4*) qui contient à la fois le flux visuel et le flux audio. Les étapes majeures sont les suivantes :

1. Extraction des Frames Vidéo

Les frames sont extraites de manière équidistante avec **OpenCV**. Chaque frame est convertie en format RGB et redimensionnée à 224×224 pixels pour être compatible avec le modèle **MobileNetV2**.

2. Extraction des Embeddings Visuels

Le modèle MobileNetV2 pré-entraîné (avec pooling global) fournit un vecteur d'embedding de dimension 1280 pour chaque frame extraite.

3. Extraction de l'Audio

La commande ffmpeg est exécutée via le module **subprocess** pour extraire l'audio du fichier vidéo. Le signal audio est ensuite lu avec **soundfile** et segmenté en autant de parties que le nombre de frames. Pour chaque segment, les **MFCCs** sont calculés à l'aide de **librosa** et moyennés pour obtenir un vecteur de dimension 13.

4. Projection des Features Audio

Afin d'aligner les dimensions des embeddings visuels (1280) et des embeddings audio (13), nous utilisons une projection linéaire aléatoire fixe qui transforme les vecteurs audio dans un espace de dimension 1280. Cette étape est cruciale pour permettre le calcul de la similarité cosinus entre des vecteurs de même dimension.

5. Définition et Calcul de la Synergie

La synergie $S(i,j)$ est calculée comme la similarité cosinus entre un vecteur d'embedding visuel et le vecteur audio projeté correspondant, assurant ainsi une valeur dans l'intervalle $[0,1]$.

6. Mise à Jour des Pondérations du SCN

La règle de mise à jour utilisée est :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big],$$

ce qui correspond à un renforcement proportionnel à la synergie, combiné à une décroissance pour éviter une croissance illimitée. Après plusieurs itérations, la dynamique converge vers des valeurs reflétant la force de la synergie entre les entités.

7. Visualisation

La matrice finale des pondérations est affichée sous forme de heatmap, et une projection 2D des embeddings visuels ainsi que des embeddings audio projetés permet d'obtenir une intuition sur la structuration des clusters dans l'espace des features.

Ce programme fournit ainsi une **illustration concrète** de la reconnaissance audio-visuelle appliquée au Deep Synergy Learning, en intégrant directement l'audio depuis le fichier vidéo sans MoviePy, et en harmonisant les dimensions des données via une projection linéaire.

C. Systèmes Multimédias et Fusion Avancée

Lorsqu'un **SCN** (Synergistic Connection Network) s'étend à plusieurs sources multimédias – texte, audio, images ou encore métadonnées – le **Deep Synergy Learning (DSL)** offre un cadre général pour faire émerger des **macro-clusters** ou des **communautés** qui reflètent des thèmes globaux, des préférences utilisateur ou des types de contenus récurrents. L'idée directrice reprend la logique exposée en **section 2.2.1** (définition d'entités variées) et en **section 2.2.2** (mise à jour locale des liaisons), tout en l'adaptant à des couples d'entités relevant de modalités différentes.

Pour un **système** multimédia complexe, on définit plusieurs catégories d'entités :
– Des entités **texte** $\{\mathcal{T}_k\}$, regroupant par exemple des mots, des tags, des descriptions ou des résumés de contenu.

– Des entités **audio** $\{\mathcal{A}_j\}$, décrites via des embeddings sonores (spectrogrammes, MFCC ou modèles pré-entraînés) et potentiellement associées à des pistes musicales ou des bruitages.

– Des entités **visuelles** $\{\mathcal{V}_i\}$, allant d'images fixes à des extraits ou des embeddings vidéo résumant les frames d'un clip.

– Des entités **métadonnées** $\{\mathcal{M}_r\}$, comme la géolocalisation, les dates, les catégories, ou encore les profils d'utilisateur $\{\mathcal{U}_\ell\}$.

L'ensemble global de ces entités forme le réseau $\mathcal{E}$, où chaque nœud $\mathcal{E}_u$ provient d'une de ces modalités. On introduit alors une matrice de pondérations ω, de taille $|\mathcal{E}| \times |\mathcal{E}|$. Comme indiqué en **section 2.2.2**, la mise à jour de chaque liaison $\omega_{u,v}$ obéit à une équation de la forme

$$\omega_{u,v}(t+1) \;=\; \omega_{u,v}(t) \;+\; \eta\big[S(\mathcal{E}_u, \mathcal{E}_v) - \tau\, \omega_{u,v}(t)\big].$$

La **fonction de synergie** $S(\mathcal{E}_u, \mathcal{E}_v)$ s'adapte au type de couple (u, v). Entre un nœud **texte** et un nœud **image**, on pourra s'appuyer sur une similarité cosinus dans un espace latent commun. Entre un nœud **utilisateur** et un nœud **vidéo**, on peut modéliser la probabilité que cet utilisateur aime la vidéo en exploitant les liens qu'il partage déjà avec des contenus semblables. Entre un **profil** utilisateur et un **tag** textuel, on peut s'appuyer sur les préférences extraites par co-occurrence ou par historique d'utilisation.

De cette façon, la **dynamique** DSL autorise la fusion de multiples modalités dans un même **SCN**. Lorsqu'on applique cette méthode à une **plateforme** multimédia – par exemple, un site de diffusion de vidéos musicales – on obtient un graphe où :

– Les entités "vidéo $\mathcal{V}_i$" sont décrites par leurs embeddings visuels ou par des analyses de frames.
– Les entités "audio $\mathcal{A}_j$" reflètent les pistes sonores ou les morceaux musicaux sous forme de vecteurs caractéristiques.

– Les entités "texte $\mathcal{T}_k$" regroupent titres, tags, descriptions ou catégories associées.

– Les entités "utilisateurs $\mathcal{U}_\ell$" décrivent les profils (préférences, historique de navigation ou d'écoute).

La **synergie** $S(\mathcal{U}_\ell, \mathcal{V}_i)$ peut traduire la probabilité qu'un utilisateur $\mathcal{U}_\ell$ apprécie la vidéo $\mathcal{V}_i$. Cette probabilité peut reposer sur la similarité de $\mathcal{V}_i$ avec d'autres vidéos que $\mathcal{U}_\ell$ a aimées, sur des points communs textuels ($\mathcal{T}_k$) ou sur des corrélations audio ($\mathcal{A}_j$). Au fur et à mesure que le réseau s'**auto-organise**, les liens $\omega_{\ell,i}$ se renforcent si l'utilisateur $\mathcal{U}_\ell$ et la vidéo $\mathcal{V}_i$ présentent

une affinité, et décroissent dans le cas contraire, aboutissant à la formation de **communautés** de goût ou de **clusters** de contenus.

À mesure que le réseau grandit avec l'ajout de nouvelles vidéos, pistes audio, tags ou utilisateurs, la **dynamique DSL** s'adapte de façon **incrémentale**. Le **SCN** évalue la synergie entre la nouvelle entité et celles déjà présentes, met à jour les pondérations ω, et intègre progressivement cette entité dans le graphe, parfois en la rattachant à des **clusters** existants.

Pour éviter une explosion du nombre de liaisons, des **règles de parsimonie** (section 2.2.3) et d'**inhibition** (chap. 7.4) sont appliquées. Les poids $\omega_{u,v}$ trop faibles sont filtrés, et la croissance simultanée d'un trop grand nombre de connexions est limitée, garantissant ainsi que la structure du graphe reste **gérable** et **optimisée**.

L'**avantage** principal de cette approche réside dans son **unification auto-organisée**. Les **macro-clusters** qui émergent peuvent regrouper simultanément des éléments **sonores**, des **vidéos**, des **étiquettes textuelles** et des **profils utilisateur**. Ces ensembles multimodaux et multi-entités définissent ainsi des **thèmes** ou des **centres d'intérêt** partagés, sans nécessiter de modèle supervisé distinct pour chaque modalité.

Cette **fusion avancée** favorise plusieurs applications. La **recommandation de contenus** devient plus efficace, car un utilisateur $\mathcal{U}_\ell$ peut découvrir de nouvelles vidéos $\mathcal{V}_i$ associées à sa communauté. La **recherche d'information** bénéficie du croisement naturel entre **texte, image et audio**, tandis que la **scalabilité** est assurée par la nature **incrémentale et distribuée** du DSL.

Un **SCN** multimédia fonctionne ainsi comme un **réseau adaptatif** où chaque entité, quelle que soit sa modalité, établit des connexions pertinentes. Il en résulte des **macro-clusters** riches de sens, offrant des opportunités étendues pour l'**exploration**, l'**indexation** et la **recommandation** à grande échelle.

8.1.3. Plan du Chapitre

Afin de déployer clairement la logique du **DSL multimodal**, ce chapitre 8 s'organise en plusieurs sections, chacune dédiée à un **aspect** précis de la fusion entre différentes modalités (vision, langage, audio, etc.). L'objectif est d'étudier **comment** le **DSL** (Deep Synergy Learning) orchestre l'**intégration** de ces flux, tant sur les plans conceptuels (principes de fusion), qu'implémentaires (architectures SCN dédiées) ou mathématiques (définition de la synergie inter-modalités).

8.1.3.1. Présentation de la Structure : (8.2) Principes de la Fusion Multimodale en DSL, (8.3) Vision, (8.4) Langage, (8.5) Audio, etc.

À ce stade, il est pertinent de **structurer** la suite de l'**exposition** sur la **fusion multimodale** dans le cadre du **Deep Synergy Learning** (**DSL**). L'objectif est d'organiser la réflexion et les développements au sein du **Synergistic Connection Network** (**SCN**) lorsque plusieurs **modalités** comme la **vision**, le **langage** et l'**audio** interagissent simultanément.

La **section 8.2** explore les **principes fondamentaux** de la fusion multimodale. Les **sections 8.3, 8.4 et 8.5** analysent successivement les trois canaux principaux, à savoir la **vision**, le **langage** et l'**audio**. Enfin, une perspective élargie est envisagée, intégrant d'autres flux comme

les **capteurs divers**, les **signaux EEG** ou les **métadonnées**, ouvrant ainsi la voie à des applications encore plus riches et variées.

Section 8.2 (Principes de la Fusion Multimodale) établit tout d'abord le **fondement** théorique. Il y est discuté **pourquoi** fusionner plusieurs canaux (tels que des images, du texte ou des enregistrements sonores) à l'intérieur d'un même SCN ; la **dynamique** d'**auto-organisation** du DSL y est rappelée sous forme de l'équation

$$\omega_{i,j}(t+1) = \omega_{i,j}(t)\eta\left[\, S(i,j) \;-\; \tau\,\omega_{i,j}(t)\right],$$

où $\omega_{i,j}$ représente la **pondération** entre deux entités $\mathcal{E}_i$ et $\mathcal{E}_j$. L'enjeu majeur consiste à définir la **fonction** de **synergie** S entre des entités issues de modalités différentes, par exemple une **image** et un **segment** textuel, ou bien un **extrait** audio et un **objet** visuel. Les questions de **normalisation** des échelles, de répartition dans la matrice ω selon le type (intra-canal ou inter-canal) et de **gestion** de la **complexité** sont aussi abordées dans cette section 8.2, en lien avec les **optimisations** déjà évoquées au **chapitre 7**.

Dans la **section 8.3 (Vision)**, l'accent est mis sur l'exploitation de flux **visuels** (images ou vidéos) au sein du DSL. On considère comment un **embedding** d'image, noté $\mathbf{x}_i^{(\text{visuel})} \in \mathbb{R}^d$, peut nourrir la **similarité** $\text{sim}\left(\mathbf{x}_i^{(\text{visuel})}, \mathbf{x}_j^{(\text{visuel})}\right)$ et ainsi former la **synergie** $S(i,j)$ lorsqu'il s'agit de lier deux entités purement visuelles. L'enjeu est d'analyser comment, à travers la dynamique

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right],$$

les entités s'**agglomèrent** en **clusters** à différents niveaux. D'un côté, des **micro-patches** ou **régions locales** se forment, tandis que de l'autre, des **macro-objets** ou **catégories plus abstraites** émergent progressivement (cf. **section 8.3**).

Les liens avec la **multi-échelle**, développée dans le **chapitre 6**, apparaissent clairement. Le **DSL** permet en effet la formation de **super-nœuds**, comme des ensembles d'images partageant une même classe de scènes, favorisant ainsi une **organisation hiérarchique** des connexions dans le **SCN**.

La **section 8.4 (Langage)** se penche ensuite sur la façon dont le **DSL** s'applique à des entités **linguistiques** (mots, segments phraséologiques, etc.). Elle décrit la **construction** d'embeddings textuels, comme ceux issus de **word2vec**, **GloVe** ou **transformers** (BERT, GPT), et la manière dont la **synergie** $S\left(\mathcal{E}_i, \mathcal{E}_j\right)$ s'exprime par une **distance** ou une **similarité** cosinus entre ces vecteurs linguistiques. Les **ambiguïtés** lexicales ou la possibilité de regrouper plusieurs mots en un **macro-concept** (super-nœud) donnent lieu à un **SCN** linguistique potentiellement complexe, mais dont la dynamique demeure pilotée par la même équation de mise à jour $\omega_{i,j}(t+1) = \cdots$. Les liens inter-modaux, notamment image–texte, prennent alors corps dans la matrice ω, dès lors que $\phi_{\text{img}}\left(\mathbf{x}_i^{(\text{visuel})}\right)$ et $\phi_{\text{txt}}\left(\mathbf{x}_j^{(\text{texte})}\right)$ ont été projetés dans un espace latent permettant de définir S.

La **section 8.5 (Audio)** généralise encore plus l'approche à la **dimension acoustique**. Les entités y sont des **segments** sonores, décrits par des **features** telles que des **MFCC**, des **spectrogrammes** ou des embeddings audio (par exemple wave2vec). De nouveau, on retrouve la même formule DSL, et la question essentielle consiste à élaborer $S\left(\mathcal{E}_i^{(\text{audio})}, \mathcal{E}_j^{(\text{audio})}\right)$ pour des **paires** purement acoustiques, ou à formaliser $S(\text{audio,image})$ ou $S(\text{audio,texte})$ pour les liaisons **cross-modales**. Les **clusters** ainsi produits peuvent recouvrir un large éventail de

combinaisons comme, par exemple, un groupe de sons liés à des images cohérentes ou à des tokens textuels désignant ces mêmes sons.

Enfin, la fin de la **section 8.5** (et l'ouverture plus large du chapitre 8) mentionne la possibilité de traiter d'**autres modalités**, qu'il s'agisse de signaux physiologiques, de capteurs embarqués ou d'informations structurées. Le **DSL** conserve sa puissance d'**auto-organisation** tant que l'on parvient à définir, pour toute paire $\mathcal{E}_i, \mathcal{E}_j$, une **synergie** $S(i,j)$ traduisant la compatibilité ou la co-occurrence entre ces deux entités.

Cet enchaînement (cf. section 8.2 sur les principes de la fusion, section 8.3 sur la vision, section 8.4 sur le langage, section 8.5 sur l'audio) reprend la **philosophie** d'ensemble décrite dans la **section 8.1.3.1**. Chacune de ces parties approfondit la logique d'un **SCN** multimodal et insiste sur les **défis** mathématiques (normalisation, sélection parcimonieuse des liaisons, mécanismes d'inhibition pour éviter l'explosion du nombre de liens) et sur les **bénéfices** concrets (formation de **clusters** pertinents, émergence de macro-nœuds représentant des concepts multimédias). Les références directes aux **chapitres 5** (architecture SCN), **7** (optimisations avancées) et **6** (approche multi-échelle) garantissent une cohérence interne, montrant comment la construction pratique des poids $\omega_{i,j}$ s'intègre dans la **dynamique** du **DSL** pour gérer un large éventail de **flux** simultanés.

8.1.3.2. Rappel du lien avec Chap. 5 (architecture SCN) et Chap. 7 (optimisations)

Le présent **chapitre 8**, consacré à la **dimension multimodale** du **Deep Synergy Learning (DSL)**, s'inscrit dans la continuité des **fondations** établies aux **chapitres 5** et **7**. Il est donc indispensable d'en rappeler la **cohérence** et de souligner la façon dont les **modules** et la **dynamique** décrits auparavant constituent le **socle** permettant de gérer, avec efficacité, la **fusion** de plusieurs **modalités** (image, texte, audio, capteurs divers, etc.). Les deux grandes **références** sont d'une part le **Chap. 5**, qui expose l'**architecture** du **Synergistic Connection Network (SCN)**, et d'autre part le **Chap. 7**, qui introduit les **méthodes d'optimisation** et d'**adaptation** destinées à maîtriser la complexité du réseau et à éviter des minima locaux.

*A. Lien avec le Chap. 5 : l'**architecture SCN** comme fondement pour la multimodalité*

L'**architecture** SCN, telle qu'elle est décrite au **Chap. 5**, prévoit un **noyau** central où réside la matrice $\{\omega_{i,j}\}$, ainsi que des **modules** chargés, entre autres, du **calcul** de la **synergie**, de l'**inhibition** ou de la **distribution** des entités. Dans un contexte **multimodal**, cette modularité devient un levier essentiel. Chaque **modalité** (vision, texte, audio, capteurs, etc.) peut être traitée de manière spécialisée par un **module** apte à convertir les données brutes en **entités** internes $\mathcal{E}_i$ (vecteurs de caractéristiques, représentations symboliques), puis à appeler la fonction de **synergie** adaptée. Les **pondérations** $\omega_{i,j}$ sont alors mises à jour au sein du **noyau** unique, garantissant l'**unification** du **réseau**.

Mathématiquement, la dynamique demeure commandée par la relation

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big],$$

où la **fonction** $S(\mathcal{E}_i, \mathcal{E}_j)$ dépend de la **modalité** ou du **couple** de modalités concerné. Les dispositions décrites au **Chap. 5** (sections 5.4 et 5.7) permettent d'héberger, dans un **même SCN**, des entités visuelles, textuelles ou auditives, tout en conservant une **architecture** modulaire. Les sous-SCN (voir section 5.7.1.1) peuvent traiter localement chaque modalité, tandis qu'un **super-nœud** (5.7.2) orchestre la **fusion** en reliant ces sous-réseaux entre eux.

Cette **organisation** fournit un **socle** systématique pour déployer le **DSL** sur de grands volumes de données multimodales, en évitant qu'une seule et même matrice ω ne devienne ingérable. Le **Chap. 5** met en évidence cette **vision** modulaire qui se révèle fondamentale dès que l'on aborde, comme au **chapitre 8**, la question de la **synergie** entre plusieurs **canaux** (image, texte, audio) dans un **même SCN**.

*B. Lien avec le Chap. 7 : **méthodes d'optimisation** et **adaptation** pour la multimodalité*

Le **Chap. 7** propose diverses **méthodes** destinées à **optimiser** la configuration d'un SCN, à **éviter** une prolifération de liaisons « moyennes » et à **échapper** aux minima locaux. Cette problématique est d'autant plus **cruciale** en contexte **multimodal**, où le nombre total d'entités n peut être très important (car on accumule les entités issues de chaque modalité) et où la matrice ω peut compter $O(n^2)$ liens.

Les mécanismes décrits au **Chap. 7** sont particulièrement **pertinents** ici. Les techniques de **parsimonie** et d'**inhibition** avancée (voir 7.4) limitent le risque qu'une entité se connecte faiblement à trop de nœuds, ce qui serait encore plus probable lorsque l'on gère plusieurs canaux. Sur le plan **mathématique**, ces mécanismes se traduisent par l'ajout d'un **terme** de régulation dans la dynamique, par exemple une **pénalisation**

$$\gamma \sum_{k \neq j} \omega_{i,k}(t),$$

destinée à forcer la compétition entre les liaisons sortantes d'un même nœud $\mathcal{E}_i$. Sans une telle régulation, la dynamique

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta \left[S(i,j) - \tau\, \omega_{i,j}(t) \right]$$

risquerait de laisser trop de liens inter-canaux subsister, créant une **surcharge** et nuisant à la **lisibilité** des **clusters** émergents.

Les **techniques** de **recuit simulé** (7.3) ou d'**heuristiques globales** (7.5) jouent, en outre, un rôle important pour résorber les configurations localement bloquées. Par exemple, il se pourrait qu'une association image–texte se soit figée dans un cluster inapproprié ; l'injection d'un **bruit** contrôlé, fonction d'une « température » qui décroît au fil du temps, peut libérer la structure et lui permettre de se réorienter vers un alignement plus juste. Cette capacité d'**exploration** stochastique se révèle précieuse lorsque la dimension du problème (nombre d'entités multimodales) est très élevée.

Le **Chap. 7** aborde enfin l'**apprentissage continu** (7.6), qui autorise l'intégration incrémentale de nouveaux nœuds $\mathcal{E}_{\text{new}}$ au sein du SCN sans tout réentraîner depuis zéro. Cela s'avère décisif lorsque les flux multimodaux arrivent en continu (ajout progressif de nouvelles images, de nouveaux segments textuels, etc.). Le DSL peut ainsi élargir progressivement sa matrice $\{\omega_{i,j}\}$ et adapter les liens existants en suivant la même équation d'**auto-organisation**, sous réserve de quelques précautions (par exemple, une normalisation adaptative). Les méthodes décrites dans ce même chapitre (monitoring de la densité, insertion locale) forment ainsi un **complément** indispensable pour assurer la **flexibilité** et la **réactivité** d'un SCN multimodal.

8.2. Principes Généraux de la Fusion Multimodale dans le DSL

Le **DSL** (Deep Synergy Learning) ne se limite pas à agréger des entités d'une seule nature (par exemple, uniquement textuelles ou uniquement visuelles). Sa force réside dans la **capacité** à traiter différentes **modalités** simultanément (images, textes, sons, signaux capteurs, etc.) et à **auto-organiser** les pondérations $\omega_{i,j}$ en fonction de la **synergie inter-modale**. Dans ce cadre, la synergie $S(i,j)$ ne représente plus seulement une "similarité" habituelle (p. ex., cosinus entre vecteurs d'image), mais peut impliquer la **correspondance** ou la **complémentarité** entre deux modalités distinctes.

8.2.1. Notion de Synergie Inter-Modale

8.2.1.1. Définition : comment $S(i,j)$ relie une entité "visuelle" $\mathcal{E}_i$ et une entité "textuelle" $\mathcal{E}_j$?

Dans le cadre d'un **DSL** (Deep Synergy Learning) visant à prendre en compte plusieurs **modalités** (par exemple la vision et le texte), il est nécessaire de définir soigneusement la **synergie** entre une **entité** $\mathcal{E}_i$ issue du canal **visuel** et une **entité** $\mathcal{E}_j$ issue du canal **textuel**. De manière générale, le **Synergistic Connection Network** (SCN) regroupe un ensemble d'entités $\{\mathcal{E}_1, \mathcal{E}_2, \ldots\}$ provenant de différents canaux ; chacune $\mathcal{E}_i$ s'appuie sur une **représentation** (embedding, vecteur de caractéristiques, etc.). Lorsqu'on compare deux entités $\mathcal{E}_i \in$ Vision et $\mathcal{E}_j \in$ Texte, on cherche à construire une **fonction** $S(\mathcal{E}_i, \mathcal{E}_j)$ reflétant le **degré** de correspondance ou de compatibilité entre elles.

A. Entités Visuelles et Entités Textuelles

D'un côté, des **entités visuelles** peuvent correspondre à des *patchs* (régions extraites d'une image), à des *objets détectés* (boîtes englobantes fournies par un détecteur) ou à des *embeddings* (vecteurs de caractéristiques produits par un réseau convolutionnel type **ResNet**, **VGG** ou autre). On associe ainsi à chaque entité $\mathcal{E}_i^{(\text{visuel})}$ un vecteur $\mathbf{v}_i \in \mathbb{R}^d$, obtenu en traitant l'image (ou sa région) à l'aide d'un **CNN**.

De l'autre côté, des **entités textuelles** se déclinent sous forme de *mots*, de *tokens* de phrase, de syntagmes ou même de segments de texte plus longs (ex. clauses ou phrases entières). Il est usuel de les représenter par un **embedding** $\mathbf{t}_j \in \mathbb{R}^{d'}$, par exemple un vecteur Word2vec, GloVe ou BERT. L'entité $\mathcal{E}_j^{(\text{texte})}$ encapsule donc cette représentation vectorielle, sur laquelle on va opérer pour définir la similarité.

B. Construction de la Synergie Inter-Modale $S(\mathcal{E}_i, \mathcal{E}_j)$

Pour **lier** un vecteur visuel $\mathbf{v}_i \in \mathbb{R}^d$ à un vecteur textuel $\mathbf{t}_j \in \mathbb{R}^{d'}$, on doit concevoir une **fonction** de correspondance, notée $f(\mathbf{v}_i, \mathbf{t}_j)$, qui renvoie une **valeur** de similarité ou de complémentarité dans $\mathbb{R}^+$ ou $[0,1]$. Il existe plusieurs stratégies :

1. Approche par projection dans un espace commun :

Une méthode fréquente consiste à définir deux matrices (ou opérateurs) de projection W_v et W_t qui amènent respectivement $\mathbf{v}_i$ et $\mathbf{t}_j$ dans un même espace latent $\mathbb{R}^D$. On peut alors calculer un **produit scalaire** :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \sigma\langle W_v\, \mathbf{v}_i\, ,\, W_t\, \mathbf{t}_j \rangle,$$

où σ est une **activation** (par exemple une sigmoïde) garantissant un score dans $[0,1]$. Cette démarche s'apparente aux modèles "vision-language" qui apprennent un **espace commun** où images et textes se rapprochent quand ils désignent le même concept.

2. Similarité cosinus :

Une autre approche, plus directe, consiste à **normaliser** $\mathbf{v}_i$ et $\mathbf{t}_j$ puis à évaluer leur **cosinus** :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \cos\left(\phi_{\text{vis}}(\mathbf{v}_i),\ \phi_{\text{txt}}(\mathbf{t}_j)\right),$$

la fonction ϕ_{vis} et ϕ_{txt} pouvant être de simples normalisations ou des transformations apprises. Le score cos proche de 1 indique une forte corrélation (même direction), un score proche de 0 suggère une quasi-orthogonalité. Cette évaluation succincte est souvent suffisante pour générer un **degré** de synergie.

3. Autres fonctions :

On peut imaginer des **kernels** plus riches (RBF, polynomial) ou des **modèles** de co-occurrence statistique, si l'on dispose d'un corpus où images et textes sont co-présents. D'un point de vue **DSL**, toute fonction $S(\mathbf{v}_i, \mathbf{t}_j) \geq 0$ est recevable, dans la mesure où elle fournit un **signal** local pour renforcer ou affaiblir les pondérations $\omega_{i,j}$.

C. Complémentarité et Corrélation sémantique

Lorsque le **texte** "cat" et le **patch** d'image représentant un **chat** partagent des **features** sémantiques similaires (que ce soit via un produit scalaire ou un cosinus élevé), on obtient $S(i,j)$ important. Inversement, un mot "banana" en face d'une voiture visuelle génère $S(i,j) \approx 0$. Cette **quantification** détermine ensuite la mise à jour $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$, ce qui **auto-organise** le **SCN** de manière à **associer** des entités textuelles et visuelles vraiment liées.

D. Cas d'Exemple

Dans un scénario de **captioning d'image**, on peut disposer d'entités $\{\mathcal{E}_i^{(\text{image})}\}$ représentant différents objets (patchs) et d'entités $\{\mathcal{E}_j^{(\text{texte})}\}$ représentant des *mots* ou *n-grams*. Le **DSL** renforce les liens $\omega_{i,j}$ lorsque l'on détecte une haute **synergie** S. Les **clusters** (image + mots) qui en résultent indiquent la correspondance "ce patch de l'image est un chat, associé au mot 'cat'". Similairement, en **recherche cross-modale**, une requête "tower in Paris" sera reliée à l'image de la Tour Eiffel si leurs embeddings respectifs convergent vers un **score** S élevé.

E. Propriétés Mathématiques et Localité/Globalité

La **dimension** des espaces de représentation ($\mathbb{R}^d$ vs. $\mathbb{R}^{d'}$) varie selon la modalité. Pour **harmoniser**, on recourt à un **mapping** φ: $\mathbb{R}^d \times \mathbb{R}^{d'} \to \mathbb{R}$. De plus, la **localité** ou **globalité** de la comparaison est un choix d'implémentation. Il est possible de comparer un *patch* visuel à un *mot* précis (localité fine) ou bien la totalité de l'image à une **phrase** complète (globalité). Dans

un **SCN**, rien n'empêche de combiner ces deux approches si l'on souhaite modéliser plusieurs niveaux.

La **synergie** S peut en outre **évoluer** avec le temps, si un module interne affine les espaces d'embeddings. Le **DSL** s'ajuste alors naturellement, car la mise à jour $\omega_{i,j}$ dépend de la valeur courante de S.

F. Intérêt pour le DSL

L'introduction d'une **synergie inter-modale** est un levier clé pour conférer au **DSL** sa capacité de **fusion** multimédia. L'**auto-organisation** demeure pilotée par la même règle locale, mais la **fonction** S qui la sous-tend diffère selon les paires (visuel–textuel, texte–texte, visuel–visuel, etc.). Par ce biais, le SCN peut simultanément :

- **Former** des **clusters** multimodaux (image(s) + mot(s))
- **Relier** spontanément de nouvelles entités textuelles ou visuelles en fonction de leur score de similarité
- **Explorer** un large spectre d'arrangements possibles, s'adaptant ainsi à une grande diversité d'images, de mots et de combinaisons

Cela se traduit par un **réseau** (SCN) où, sans supervision explicite, les nœuds visuels et textuels s'**agrègent** selon leur proximité ou leur complémentarité.

8.2.1.2. Exemples : similarité d'un label texte avec un embedding d'image, correspondance audio-vidéo, etc.

La **fonction** de **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ peut relier des entités de différentes **modalités**, comme un **label textuel** et un **embedding** d'image, ou encore un **segment audio** et un **extrait vidéo**. Ces correspondances s'inscrivent dans le cadre du **DSL** (Deep Synergy Learning) appliqué à plusieurs canaux (texte, vision, audio, etc.) et permettent au **Synergistic Connection Network** (SCN) de gérer de manière **auto-organisée** des données hétérogènes. Les deux exemples présentés ci-après (texte–image et audio–vidéo) illustrent la façon dont se calcule la **synergie** cross-modale et comment cette valeur S est ensuite réinjectée dans la **dynamique** du DSL pour renforcer ou affaiblir les liaisons $\omega_{i,j}$.

A. Similarité Texte–Image : Label Textuel vs. Embedding d'Image

L'association entre un **mot** (ou un ensemble de mots) et la **représentation** vectorielle d'une image (qu'il s'agisse d'une image complète ou d'un patch visuel) illustre un premier cas d'usage du **Deep Synergy Learning (DSL)** dans un contexte multi-modal. Il s'agit de mettre en place un **SCN** (Synergistic Connection Network) où les entités d'intérêt, notées $\mathcal{E}_i^{(\text{img})}$ et $\mathcal{E}_j^{(\text{txt})}$, se trouvent reliées par des poids $\omega_{i,j}$ dont l'évolution dépend d'une **mesure** de similarité inter-modale.

Dans un cadre sub-symbolique, le **vecteur** visuel $\mathbf{v}_{\text{img}}$ provient généralement d'un **réseau** de neurones spécialisé (CNN, ResNet, ViT), qui transforme l'image initiale en un embedding $\mathbf{v}_{\text{img}} \in \mathbb{R}^d$. Ce vecteur capture des caractéristiques comme la forme, la texture ou la répartition d'objets, et constitue l'entité $\mathcal{E}_i^{(\text{img})}$. Parallèlement, on considère un **label** textuel, par exemple

“cat”, traduit lui aussi en vecteur $\mathbf{v}_{\text{txt}} \in \mathbb{R}^m$ grâce à un modèle comme Word2Vec, GloVe ou BERT. La dimension m peut diverger de d ; pour unifier les représentations, on applique souvent une **transformation** linéaire $\mathbf{W}: \mathbb{R}^m \to \mathbb{R}^d$, aboutissant au vecteur $\mathbf{v}_{\text{txt}}{}' = \mathbf{W}\,\mathbf{v}_{\text{txt}}$.

Dans un **SCN** multimodal, la **synergie** entre ces deux entités, notée $S(\mathcal{E}_{\text{img}}, \mathcal{E}_{\text{txt}})$, se définit usuellement à partir d’une **similarité** cosinus dans $\mathbb{R}^d$:

$$S(\mathcal{E}_{\text{img}}, \mathcal{E}_{\text{txt}}) \;=\; \frac{\mathbf{v}_{\text{img}} \cdot \mathbf{v}_{\text{txt}}{}'}{\| \mathbf{v}_{\text{img}} \| \, \| \mathbf{v}_{\text{txt}}{}' \|}.$$

Si cette valeur se montre élevée, alors la **pondération** $\omega_{\text{img,txt}}(t)$ tendra à croître selon la règle de mise à jour présentée en **section 2.2.2**, à savoir

$$\omega_{\text{img, txt}}(t+1) \;=\; \omega_{\text{img, txt}}(t) \;+\; \eta\big[S(\text{img, txt}) - \tau\,\omega_{\text{img, txt}}(t)\big].$$

Un **mot** comme “chat” verra donc son lien ω renforcer avec l’entité visuelle représentant effectivement un chat, tandis qu’il restera faible pour un patch d’image figurant plutôt une voiture. Au fil des itérations, cette dynamique **auto-organisée** amène le réseau à regrouper de façon cohérente les entités visuelles et textuelles.

B. Correspondance Audio–Vidéo : Aligner un Segment Sonore et un Flux Visuel

Un deuxième exemple typique concerne la mise en correspondance d’un **segment** audio avec un extrait **vidéo** au sein d’un SCN multimodal. Il s’agit ici de repérer si un certain $\mathcal{E}_{\text{audio}}$, qui peut symboliser un bruit (comme le moteur d’une voiture, le chant d’un oiseau ou un klaxon), s’aligne sur $\mathcal{E}_{\text{vid}}$, c’est-à-dire un lot de frames ou un embedding vidéo représentant la même réalité visuelle.

Du côté **audio**, on segmente le signal en extraits (frames temporelles ou segments) et l’on calcule un **embedding** $\mathbf{v}_{\text{audio}} \in \mathbb{R}^p$ à l’aide de méthodes comme les MFCC (coefficients cepstraux en fréquence mél), la transformation en spectrogrammes ou encore des modèles avancés de type Wav2Vec. Du côté **vidéo**, on peut recourir à un CNN 2D + temps ou à un transformeur vidéo pour extraire un vecteur $\mathbf{v}_{\text{vid}} \in \mathbb{R}^q$. Dans l’hypothèse d’une comparaison directe, on projette $\mathbf{v}_{\text{audio}}$ et $\mathbf{v}_{\text{vid}}$ dans un **espace latent** commun $\mathbb{R}^d$, puis on définit la synergie par une gaussienne ou un cosinus :

$$S(\mathcal{E}_{\text{audio}}, \mathcal{E}_{\text{vid}}) \;=\; \exp(-\alpha \;\| \mathbf{v}_{\text{audio}}{}' - \mathbf{v}_{\text{vid}}{}' \|^2), \quad \text{ou} \quad \cos(\mathbf{v}_{\text{audio}}{}', \mathbf{v}_{\text{vid}}{}').$$

On peut également inclure un **facteur temporel**, en considérant qu’un segment audio est d’autant plus apte à renforcer le lien $\omega_{\text{audio, vid}}$ qu’il se superpose temporellement à la séquence vidéo analysée. La règle de mise à jour DSL vient alors renforcer ou atténuer les liaisons en fonction de la pertinence détectée. Si dans une scène sportive, un bruit de foule coïncide avec la vue d’un stade, $\omega_{\text{audio, vid}}$ s’accroît et consolide un **cluster** rassemblant cet extrait vidéo et ce fragment sonore.

C. Généralisation et Exploitation dans le DSL

De façon plus générale, le **DSL** se prête à la **fusion** de plusieurs modalités, pas seulement l’image ou l’audio, mais aussi le texte, la métadonnée (localisation, date, etc.) ou encore la mesure capteur (température, mouvement, etc.). Chaque **modalité** se voit associée à un **embedding** adapté, et l’on définit une fonction de **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ pour chaque couple d’entités (i, j). Les modalités “intra-catégorie” (texte–texte, image–image, etc.) utilisent des métriques classiques (cosinus, distance exponentielle), tandis que les modalités “inter-

catégorie" (texte–image, audio–vidéo, etc.) font l'objet de projections ou de mécanismes d'alignement plus complexes.

La **mise à jour** de la pondération $\omega_{i,j}$ demeure, dans tous les cas, fidèle au schéma :

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big],$$

tel que prescrit en **section 2.2.2**. Ce processus **auto-organisé** entraîne une **polarisation** des liens en faveur des combinaisons $(\mathcal{E}_i, \mathcal{E}_j)$ qui révèlent une affinité notable. Cette convergence se traduit alors par la formation de **clusters** multimodaux, reliant différents types d'entités s'il existe une cohérence sémantique ou temporelle commune. Par exemple, un cluster peut englober un segment sonore, un patch visuel et un mot (ex. "car"), tous se trouvant "mutuellement cohérents". Les règles de **parsimonie** (coupes de liens faibles) et les **heuristiques** comme le recuit simulé (voir chap. 7.3) ou l'inhibition adaptative (chap. 7.4) contribuent à éviter la sur-densification du graphe et à franchir les minima locaux.

L'**exploitation** de ces clusters multimodaux concerne de nombreux cas d'usage, notamment l'**annotation automatique**, la **détection d'événements audio-visuels**, la **recherche cross-modale**, la **recommandation de contenus**, ainsi que la **construction d'un réseau sémantique généraliste**, où chaque modalité contribue à une représentation commune des mêmes objets. Le DSL agit comme un **socle** unificateur, puisqu'il régit la **dynamique** des pondérations ω d'une manière cohérente et distribuée, mettant en évidence les **associations** saillantes par renforcement local des synergies.

8.2.2. Avantages du DSL pour la Fusion

Dans la fusion multimodale, l'objectif est de **combiner** différents types de données (image, texte, audio, capteurs…) pour obtenir une **compréhension** ou une **représentation** plus riche du même concept. Le **Deep Synergy Learning (DSL)**, avec sa logique d'auto-organisation (chap. 1–7), se prête particulièrement bien à cette tâche. En effet, plutôt que de passer par un pipeline figé — où chaque modalité est traitée séparément puis "collée" en sortie —, le DSL propose un **réseau auto-organisé** $\{\omega_{i,j}\}$ qui relie les entités issues de différentes sources en fonction de leur **synergie** (cohérence). Nous présentons ici quelques **avantages** majeurs du DSL pour la fusion multimodale :

- **Auto-organisation**
- **Adaptation**
- **Clusters multimodaux**

8.2.2.1. Auto-organisation

La **force** du **DSL** (Deep Synergy Learning) réside dans sa capacité à opérer de manière **auto-organisée**, c'est-à-dire sans qu'un pipeline rigide ne soit imposé pour "fusionner" deux canaux comme l'image et le texte. Le **Synergistic Connection Network** (SCN) explore plutôt ses propres configurations en **renforçant** ou en **affaiblissant** les pondérations $\omega_{i,j}$ entre entités. Lorsque la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ s'avère élevée, un **lien** solide se crée **spontanément**. À l'inverse, si la synergie est jugée faible, la liaison $\omega_{i,j}$ décroît jusqu'à être négligeable. Cette section

illustre comment un flux d'images et un flux textuel se **coordonnent** d'eux-mêmes dans ce **contexte** auto-organisé, selon la règle dynamique du DSL.

A. Formation Spontanée de Liens

Il existe un cas très simple qui illustre l'idée de **formation** spontanée de liaisons dans un **SCN** (Synergistic Connection Network) multimodal. Imaginons qu'une entité $\mathcal{E}_{\text{img}}$ représente une **image** contenant un chat, tandis qu'une autre entité $\mathcal{E}_{\text{txt}}$ correspond au **mot** "cat". Supposons en outre que ces entités sont décrites par des **embeddings** vectoriels, l'un pour la partie visuelle, l'autre pour le label textuel. Quand les deux embeddings affichent une **similarité** notable, on peut formaliser cette correspondance via une **synergie** :

$$S(\mathcal{E}_{\text{img}}, \mathcal{E}_{\text{txt}}) \;=\; \cos\Big(\phi_{\text{vis}}(\mathbf{v}_{\text{img}}),\, \phi_{\text{txt}}(\mathbf{v}_{\text{txt}})\Big),$$

où ϕ_{vis} et ϕ_{txt} projettent respectivement l'image et le mot dans un espace commun ou au moins comparable. Si ce score dépasse un **seuil** θ, on infère qu'il existe une proximité sémantique marquée entre l'image "chat" et le label "cat". Dans la logique du **DSL**, la mise à jour de la pondération ω liant ces deux entités s'effectue suivant la règle de la **section 2.2.2**, c'est-à-dire :

$$\omega_{\text{chat, cat}}(t+1) = \omega_{\text{chat, cat}}(t) + \eta\big[S(\text{chat, cat}) - \tau\, \omega_{\text{chat, cat}}(t)\big].$$

Tant que la **synergie** $S(\text{chat}, \text{cat})$ demeure élevée, la valeur de $\omega_{\text{chat, cat}}$ se renforce d'elle-même au fil des itérations. Il n'est pas nécessaire de définir un pipeline rigide où l'image serait classée ou labellisée de façon programmée. C'est la **proximité** vectorielle (embedding visuel vs. embedding textuel) qui **guide** la convergence de $\omega_{\text{chat, cat}}$. Ainsi, l'apparition d'un cluster reliant "image de chat" et "mot cat" se produit **spontanément**, traduisant la correspondance sémantique identifiée.

B. Émergence sans Pipeline Figé

La spécificité du **DSL** réside dans le fait qu'il **s'affranchit** d'une architecture imposant un ordre de traitement, du type « image → réseau → label ». Au contraire, le **SCN** se présente comme un réseau plus général (graphe) où chaque entité – qu'elle soit une image, un mot, un segment sonore, etc. – peut potentiellement se connecter à toute autre entité. Les liaisons $\omega_{i,j}$ s'organisent localement en fonction de la **valeur** de synergie $S(i,j)$.

Sur le plan **mathématique**, un **processus** auto-organisé remplace l'enchaînement séquentiel d'un pipeline. À chaque itération, la mise à jour locale

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big]$$

se produit pour toutes les paires (i,j). Si un certain couple image–texte, comme "chat" / "cat", jouit d'une synergie supérieure, alors $\omega_{\text{chat, cat}}$ augmentera progressivement. Le résultat final est une **stabilisation** autour de valeurs élevées pour les liens cohérents, et de valeurs faibles (voire coupées) pour les autres. Aucun pipeline de classification explicite n'est donc imposé ; c'est le **DSL** qui, selon la **section 2.2.3** (parsimonie) et la **section 2.2.4** (états internes éventuels), régule localement la compétition entre liaisons.

C. Exemple : Images et Légendes

Un cas d'école correspond à un dataset contenant des **images** ainsi que des **phrases** de description (légendes). Chaque image reçoit un embedding, et chaque phrase est convertie en

un vecteur du même espace ou d'un espace comparable. Une fonction de **synergie** $S(\text{img}, \text{légende})$ mesure la similarité entre l'embedding global de l'image et celui de la légende. Le **DSL** renforce alors les liaisons $\omega_{\text{img, légende}}$ chaque fois que la correspondance apparaît réellement fondée. C'est ainsi qu'un petit ensemble de légendes adéquates se voit rapproché d'une image donnée, tandis qu'une image très proche se concentre sur les mêmes légendes.

En pratique, cela conduit à l'émergence de **clusters** regroupant des images quasi identiques ou apparentées, ainsi que leurs légendes communes ou synonymes. L'**auto-organisation** confère au système une extensibilité naturelle : l'ajout de nouvelles images et légendes entraîne une mise à jour de la dynamique de ω, ajustant les connexions en fonction du **score de synergie**.

Aucune étape d'alignement strict n'est requise, car la **cohérence locale** des associations suffit à façonner la structure globale du **SCN**, permettant une organisation fluide et adaptative.

E. Bénéfices et Gains

Une telle **auto-organisation** offre plusieurs **avantages**. Elle se révèle bien plus **flexible** qu'un pipeline dirigé, puisqu'il n'est pas nécessaire d'acheminer une image vers un module de classification pour obtenir un label. Chaque entité, qu'il s'agisse d'une image ou d'un mot, interagit directement avec les autres via les pondérations $\omega_{i,j}$. Si une proximité est détectée, les connexions se renforcent, sinon elles s'affaiblissent progressivement.

Cette approche garantit une certaine **robustesse**. Même si un label est imparfait ou qu'un embedding visuel est bruité, la **cohérence statistique collective** au sein du réseau peut compenser ces incertitudes. L'**évolutivité** constitue un autre atout majeur. Si un **nouveau canal**, comme l'audio, doit être intégré, il suffit d'introduire les entités audio $\{\mathcal{E}_{\text{aud}}\}$ dans le **SCN**, de définir les fonctions de synergie $S(\text{aud}, \text{img})$ et $S(\text{aud}, \text{txt})$, puis de poursuivre la mise à jour du réseau, sans nécessiter une reconstruction complète du modèle.

F. Perspectives pour la Fusion

Cette philosophie, détachée d'un pipeline séquentiel, se révèle particulièrement intéressante en contextes **multimodaux** ou multi-sources. Dès lors qu'une **fonction** $S(\mathcal{E}_i, \mathcal{E}_j)$ est définie pour chaque couple (i, j), il devient possible d'étendre la logique à de multiples canaux (texte, audio, image, vidéo, capteurs, etc.). Chaque entité $\mathcal{E}_i$ se contente de procéder à la mise à jour locale de $\omega_{i,j}$ selon la **section 2.2.2**, ce qui favorise la formation de **clusters** multimodaux de grande variété (par exemple, un patch visuel peut se lier à un mot, un segment sonore, voire un paramètre de localisation).

Sur le plan mathématique, tout ce mécanisme relève d'une **dynamique** de descente, éventuellement assortie de recuit simulé (chap. 7.3) ou d'inhibition adaptative (chap. 7.4). Au lieu de coder manuellement la façon dont un CNN se connecte à un embedding lexical, on laisse la **règle** $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta \left[S(i,j) - \tau\, \omega_{i,j}(t)\right]$ agir dans un **réseau**. Les **clusters** émergent donc de la **somme** des interactions, consolidant les liens forts et élaguant les liens faibles, selon la logique de parsimonie exposée en **section 2.2.3**. Cette absence de pipeline figé, couplée à la **capacité** d'auto-organisation, distingue nettement le **DSL** des approches traditionnelles de deep learning, tout en s'appuyant sur des fondements mathématiques et algorithmiques clairs.

8.2.2.2. Adaptation : si un flux est manquant (image floue, son corrompu), la synergie s'ajuste, pas de pipeline figé

Lorsque plusieurs **flux** comme la **vision**, l'**audio** ou le **texte** sont traités dans un **DSL** (Deep Synergy Learning) appliqué à un **Synergistic Connection Network** (**SCN**), il arrive qu'un **canal** soit inexploitable ou dégradé, par exemple une **image floue** ou une **bande sonore corrompue**. Dans les architectures **multimodales** classiques, la perte d'un flux peut compromettre l'ensemble du processus.

En revanche, la **dynamique auto-organisée** du **SCN** permet une **révision spontanée** des liaisons $\omega_{i,j}$, où la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ s'adapte à la situation. Cette propriété confère au **DSL** une **grande robustesse**. Même lorsqu'un flux fait défaut, le **réseau** continue d'évoluer et redistribue ses connexions sur les modalités restantes, maintenant ainsi une organisation stable et pertinente.

A. Absence ou Altération d'un Flux

Il est possible que certains canaux dans un système multimodal se trouvent momentanément inutilisables ou corrompus. Un exemple fréquent consiste en une image floue qui ne fournit plus d'information exploitable, un enregistrement audio pollué par un fort bruit de fond, ou un texte fortement tronqué. Dans une architecture de fusion classique, l'ensemble du pipeline peut être bloqué si le module correspondant à ce flux n'est plus opérationnel. Le **Deep Synergy Learning (DSL)** s'appuie sur un SCN (Synergistic Connection Network) qui ne dicte pas un enchaînement séquentiel fixe. Chaque entité associée à un flux forme un nœud du réseau, et les pondérations $\omega_{i,j}$ évoluent en fonction de la synergie $S(i,j)$.

Lorsque le flux d'une caméra devient inopérant, la **similarité** image–texte ou image–audio décroît, car l'embedding visuel se trouve altéré. Le SCN répercute cette détérioration dans la mise à jour locale, qui tend à réduire la valeur de $\omega_{i,j}$. La disparition de la synergie ne produit pas de rupture globale, le réseau continue de fonctionner en s'appuyant sur les canaux encore valides. Cette flexibilité permet un fonctionnement auto-adaptatif, en autorisant la **rétraction** des liaisons attachées aux flux défaillants et en laissant intacts les autres liens.

B. Ajustement Automatique de la Synergie

La **règle** de mise à jour du DSL demeure la même, à savoir

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(\mathcal{E}_i, \mathcal{E}_j) - \tau\,\omega_{i,j}(t)\right].$$

Si un flux est altéré, la fonction $S(\mathcal{E}_i, \mathcal{E}_j)$ chute parce que l'embedding n'est plus fiable. Cette diminution provoque une baisse progressive des pondérations $\omega_{i,j}$. En conséquence, toute entité $\mathcal{E}_i$ abîmée s'éloigne de ses connexions précédentes. Certains scénarios modélisent cette baisse de fiabilité par un coefficient $\gamma_{\mathcal{F}}(t)$ qui multiplie la synergie, de manière à refléter la perte de qualité. Le DSL intègre alors directement cette information dans la formule de mise à jour et élimine de fait les liens peu utiles.

C. Maintien d'un Réseau Cohérent sans Pipeline Figé

Dans un pipeline traditionnel, la panne d'un capteur peut entraîner un arrêt ou une sortie erronée à cause de la dépendance stricte entre les modules. Dans le cadre du DSL, les entités continuent d'interagir librement et le canal défaillant se trouve simplement écarté par la chute de ω. Les entités non défaillantes tissent toujours leurs liens normalement, ce qui garantit la persistance de l'auto-organisation. Si une image devient floue ou un microphone se met à produire un signal

inexploitable, le SCN réduit les connexions y afférentes, en maintenant intactes les autres liaisons.

Cette stratégie rend le DSL bien plus flexible. Le mécanisme ne repose pas sur l'exigence que tous les flux fonctionnent, mais laisse la dynamique régler de façon autonome la participation relative de chaque entité. Les canaux restés opérationnels conservent leurs ω élevés, tandis que ceux devenus temporairement inutiles s'éteignent.

D. Exemples et Intérêt Pratique

Dans un contexte de **robotique**, un robot équipé de caméras, micros et capteurs lidar peut se retrouver avec des images inexploitables par faible luminosité ou obstruction. Le DSL n'interrompt pas pour autant sa fusion de données. Il diminue les pondérations liées au flux visuel et concentre la mise à jour sur les canaux audio ou tactiles. À l'instant où l'image redevient nette, la synergie image–texte ou image–audio se rétablit et les liaisons ω se renforcent naturellement.

Dans le cas d'un **dialogue multimodal**, une plateforme conversationnelle exploite l'audio (reconnaissance vocale) et le texte (suggestions, retours). Si l'audio présente des coupures ou du brouillage, la valeur de $S(\text{audio}, \text{texte})$ se retrouve pénalisée, ce qui fait retomber les liens correspondants. Le système peut alors s'appuyer sur le flux textuel seul, sans bloquer l'intégralité de la session. Cette adaptation locale se produit au sein du SCN, évitant la nécessité d'une réingénierie complète du pipeline.

En **apprentissage continu**, un flux comme $\mathcal{E}_{\text{image}}$ peut fluctuer en qualité au cours du temps. Si un segment vidéo devient trop bruité, la pondération $\omega_{\text{image},\dots}$ chute. Lorsque la qualité revient à la normale, l'embedding visuel renforce de nouveau sa synergie et la règle DSL restaure une connexion appropriée. Cette gestion fluide de la variabilité illustre la robustesse du SCN face aux changements de conditions d'observation.

8.2.2.3. Clusters multimodaux : possibilité de créer des regroupements complexes {image + son + texte} autour d'un concept

L'une des **caractéristiques** notables d'un **DSL** (Deep Synergy Learning) consistant à gérer plusieurs **modalités** (image, audio, texte, etc.) est la faculté de **composer** ou de **fusionner** diverses mesures de **synergie** pour faire émerger des **clusters** intégrant plusieurs canaux à la fois. Plutôt que de traiter un flux visuel de manière isolée, ou de se limiter à la similarité audio–audio, le **Synergistic Connection Network** (SCN) peut, de manière **auto-organisée**, regrouper simultanément des **objets visuels**, des **segments sonores**, des **extraits textuels** (ou d'autres modalités encore), si la synergie cross-modale s'avère suffisamment élevée. Cette dynamique autorise la construction de **"macro-nœuds"** complexes, incarnant un "**concept**" ou un "**événement**" multimodal.

A. Combinaison des modalités pour former un concept

Un Synergistic Connection Network (SCN) manipule des entités issues de plusieurs canaux, comme l'image, l'audio, le texte ou tout autre flux sensoriel. Chaque entité $\mathcal{E}_i$ peut donc renvoyer à un patch d'image, un court segment sonore, un embedding lexical, ou d'autres formes de représentation. Il est possible de fusionner ces diverses modalités en définissant une synergie globale $S_{\text{multi}}(\mathcal{E}_i, \mathcal{E}_j)$. Une formule courante repose sur la somme de contributions partielles associées à chaque canal, par exemple

$$S_{\text{multi}}(\mathcal{E}_i, \mathcal{E}_j) \;=\; \alpha\, S_{\text{img}}(i,j) \;+\; \beta\, S_{\text{aud}}(i,j) \;+\; \gamma\, S_{\text{txt}}(i,j),$$

avec α, β, γ supérieurs à zéro et où chaque fonction $S_{\text{img}}, S_{\text{aud}}, S_{\text{txt}}$ quantifie une similarité intra-modale (entre embeddings visuels, embeddings audio ou embeddings textuels). L'intérêt de cette construction tient dans la possibilité de gérer un score global de synergie même lorsque plusieurs modalités sont présentes. Le mécanisme de mise à jour du DSL reste identique à celui de la section 2.2.2, ce qui signifie que si ce score $S_{\text{multi}}(\mathcal{E}_i, \mathcal{E}_j)$ est élevé, la pondération $\omega_{i,j}$ se renforce.

B. Constitution de clusters complexes $\{img + aud + txt\}$

L'agrégation de plusieurs canaux dans un SCN favorise la création de regroupements plus riches. Un triplet formé d'une entité $\mathcal{E}_{\text{image}}$, d'une entité $\mathcal{E}_{\text{audio}}$ et d'une entité $\mathcal{E}_{\text{texte}}$ peut correspondre à un concept unique, comme un "événement de concert" combinant un patch visuel montrant des musiciens sur scène, un segment sonore captant le morceau joué et une phrase textuelle décrivant la chanson. Les liens entre ces entités se voient renforcés par la règle

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\big[S_{\text{multi}}(i,j) - \tau\, \omega_{i,j}(t)\big].$$

Un exemple est un cluster formé par une image de voiture, un segment audio de moteur et un label textuel "car", donnant lieu à des liens élevés entre image–audio–texte dès lors que la synergie multimodale reste cohérente. L'émergence de ces macro-clusters dépasse la liaison de deux canaux isolés et illustre la capacité du DSL à tisser des associations complexes.

C. Bénéfices pour la Représentation

Lorsque trois canaux ou plus se regroupent dans un même cluster, le réseau acquiert une sémantique plus riche. La présence simultanée de plusieurs flux garantit une meilleure robustesse face à la défaillance éventuelle d'un canal, comme une image floue ou un son corrompu, car la synergie entre d'autres flux peut préserver la cohérence du regroupement. Un cluster associant un mot "beach", un motif acoustique d'océan et un patch montrant du sable informe fortement sur l'existence d'un concept de "plage", plus nettement que ne le ferait la prise en compte d'un seul flux.

D. Mise en œuvre pratique

Le calcul des scores inter-modal se révèle nécessairement plus coûteux, puisqu'il faut évaluer la similarité dans différentes combinaisons. Lorsqu'on manipule n entités réparties sur plusieurs canaux, on peut atteindre un nombre important de comparaisons. Dans la pratique, on adopte des heuristiques pour limiter la taille du réseau ou on applique des mécanismes de filtrage pour ne conserver que les candidats probables.

Le paramétrage de la fusion pose la question de la pondération relative de chaque modalité. Le choix d'une somme linéaire $\alpha + \beta + \gamma$ constitue un exemple simple. On peut également recourir à un modèle plus élaboré, comme un réseau neuronal capable de combiner non linéairement les valeurs de similarité venant de chaque canal. Dans tous les cas, la mise à jour locale

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S_{\text{multi}}(i,j) - \tau\, \omega_{i,j}(t)\big]$$

reste la même, mais l'expression de S_{multi} varie en fonction de la stratégie de fusion.

Un exemple concret peut s'observer dans le cas d'un vlog où chaque segment temporel aligne plusieurs frames vidéo, un extrait sonore et des sous-titres textuels. Une fonction de synergie

évalue leur cohérence multimodale et la règle DSL tisse des liens élevés si l'on constate un accord sémantique ou temporel. De ce fait, on repère des macro-clusters explicitant par exemple un passage de plage, avec des mots se rapportant à la mer, le son des vagues et les frames révélant une côte sablonneuse. Cette auto-organisation simultanée dans le réseau favorise un point de vue unifié sur les données, autorisant la classification, la recherche ou l'annotation multimodale sans pipeline figé.

8.2.3. Défis et Écueils

Dans la démarche **multimodale** du DSL (Deep Synergy Learning), où l'on traite simultanément images, textes, signaux audio, voire d'autres sources (capteurs, données symboliques, etc.), plusieurs **défis** surgissent. Cette section (8.2.3) en liste quelques-uns, parmi lesquels figurent (1) l'hétérogénéité forte des représentations, (2) le surcoût computationnel (du fait de la multiplication des entités) et (3) les problèmes de synchronisation temporelle lorsque plusieurs flux doivent être alignés. Avant de détailler les aspects (8.2.3.2, 8.2.3.3), focalisons-nous sur l'**hétérogénéité** des représentations (8.2.3.1).

8.2.3.1. Hétérogénéité forte des représentations (CNN pour images, Transformers pour textes, spectrogrammes pour audio...)

Il est fréquent qu'un Synergistic Connection Network (SCN) appliqué à la fusion multimodale doive gérer des représentations très différentes selon la modalité considérée. Une architecture CNN (ResNet, VGG ou ViT) peut décrire les entités visuelles, tandis qu'un Transformer (BERT, GPT ou similaire) produit des embeddings textuels contextuels, et qu'un spectrogramme ou un autoencodeur incarne l'information audio. Cette variété assure une exploitation spécialisée de chaque flux, mais complique la mise en place d'une synergie $S(\mathcal{E}_i, \mathcal{E}_j)$ unifiée et la dynamique auto-organisée qui en découle.

A. Différentes Dimensions et Structures de Données

Les entités issues de l'image proviennent souvent d'un vecteur ou d'un tenseur élevé en dimension, suite à l'application d'un CNN ou d'un ViT. Les entités textuelles se retrouvent sous forme de word embeddings (Word2Vec, GloVe) ou d'embeddings contextuels (BERT, GPT), de dimension potentiellement différente de celle des embeddings visuels. L'audio peut être encodé sous forme de spectrogrammes, de MFCC ou à travers un modèle de type Wav2Vec. Chaque flux présente donc des caractéristiques uniques, comme la spatialité pour la vision ou la contextualité pour le texte. Cette disparité implique la nécessité de comparer des embeddings qui ne partagent pas la même échelle ni la même signification.

B. Défi d'Alignement Conceptuel

Lorsque le SCN doit évaluer la synergie entre deux entités multimodales, par exemple une image et un texte, il est indispensable d'avoir une fonction de similarité capable de gérer deux représentations très différentes. Sur le plan mathématique, on peut opter pour un espace commun de dimension D où l'on projette successivement les embeddings issus de l'image et du texte, puis effectuer une comparaison par produit scalaire ou cosinus. Il est aussi possible d'utiliser une fonction plus complexe, comme une co-attention ou un module MLP dédié, qui reçoit les deux vecteurs et produit un score de compatibilité. Sans une telle cohérence conceptuelle, le DSL se trouverait en difficulté pour relier des entités hétérogènes au sein d'un même cluster.

C. Variabilité des Extracteurs

Chaque modalité peut se voir associée à un extracteur spécialisé. Un réseau CNN préentraîné sur ImageNet détecte les formes et les textures, un Transformer BERT s'appuie sur les contextes lexicaux, tandis qu'un encodeur audio ou un spectrogramme met en avant la structure fréquentielle du son. Ces modèles produisent des embeddings qui diffèrent dans leurs échelles, leurs dispersions et leurs biais. Le SCN doit composer avec ce foisonnement de dimensions et de signatures, sous peine d'obtenir des pondérations faussées ou incompatibles entre entités. Sur le plan pratique, on peut introduire autant de modules de synergie que de paires de modalités considérées, chacun étant responsable de comparer deux types d'embeddings de manière cohérente.

D. Interaction avec la Synergie dans le DSL

La formule de base du DSL se maintient. On répète la mise à jour $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(\mathcal{E}_i, \mathcal{E}_j) - \tau\,\omega_{i,j}(t)\right]$. Cependant, la fonction S dépend du couple de modalités en cause. Lorsqu'il s'agit de deux images, on applique une métrique visuelle de type distance euclidienne ou cosinus. Lorsqu'il s'agit d'un audio et d'un texte, on peut recourir à un alignement audio–texte. Le SCN doit donc étiqueter chaque entité selon sa modalité et appeler la bonne fonction de similarité. Cela revient à disposer d'une table recensant les combinaisons possibles (image–image, image–texte, texte–texte, audio–texte, audio–image, etc.) et le module spécifique à employer.

E. Complexité Opérationnelle

Lorsque plusieurs paires de modalités se côtoient, le volume de comparaisons à exécuter peut augmenter considérablement. Il faut évaluer la synergie pour de nombreuses entités issues de chaque canal, ce qui peut coûter cher en temps de calcul et en mémoire si chaque embedding compte plusieurs centaines ou milliers de dimensions. Le DSL peut alors nécessiter des heuristiques ou un filtrage sélectif pour éviter de réaliser la totalité des comparaisons, en particulier dans un scénario à grande échelle. Il faut aussi maintenir la cohérence des embeddings ou recalculer la similarité à la volée pour éviter un stockage excessif.

F. Conséquences pour la Convergence et la Qualité

Une synergie mal calibrée entre deux modalités peut amener le réseau à se fragmenter en sous-réseaux purement unimodaux. Un ajustement judicieux de la fonction S est souvent nécessaire pour encourager le rapprochement multimodal, par exemple via un préapprentissage de type CLIP ou ALIGN qui aligne déjà l'image et le texte. L'usage d'une phase de recuit (chap. 7) ou d'une inhibition adaptative peut également aider le réseau à sortir d'attracteurs locaux et à tisser des liaisons correctes entre flux très différents. L'aboutissement recherché est un ensemble de clusters exploitant pleinement la complémentarité des canaux, mais ce résultat dépend fortement de la façon dont on définit et normalise les fonctions de similarité multimodales.

8.2.3.2. Coût computationnel : multiplication du nombre d'entités

Lorsqu'un **DSL** (Deep Synergy Learning) est étendu à la **multimodalité** (image, audio, texte, capteurs…), le **Synergistic Connection Network** (SCN) devient rapidement volumineux. En effet, chaque modalité peut générer un nombre considérable de **descripteurs** ou **segments** (patchs d'image, frames audio, tokens textuels, etc.). La **somme** de ces entités n_{total} peut se

multiplier par rapport à une approche **monomodale**, accroissant lourdement le **coût** de calcul si l'on applique naïvement une mise à jour $\omega_{i,j}$ sur toutes les paires (i, j).

A. Formulation mathématique du coût naïf

Le **SCN** se définit sur un ensemble $\{\mathcal{E}_1, \ldots, \mathcal{E}_n\}$ d'entités. Dans une configuration **monomodale**, on peut avoir n de l'ordre de quelques centaines ou milliers. Dans un contexte **multimodal**, on additionne :

1. n_{vision} entités issues de l'analyse visuelle (ex. *patchs, feature maps* d'un CNN, keypoints SIFT, etc.),
2. n_{audio} entités extraites du spectrogramme ou d'un embedding type Wav2Vec,
3. n_{texte} entités correspondant à des tokens, n-grams, o
4. u phrases issues d'un embedding BERT/GPT,
5. éventuellement n_{capteurs} pour d'autres flux sensoriels.

La **somme**

$$n_{\text{total}} = n_{\text{vision}} + n_{\text{audio}} + n_{\text{texte}} + \cdots$$

peut atteindre plusieurs milliers ou dizaines de milliers. Or, la mise à jour naïve de la matrice $\{\omega_{i,j}\}$, si elle repose sur l'équation

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta \left[S(i,j) - \tau\, \omega_{i,j}(t)\right],$$

s'effectue potentiellement pour **tous** les couples (i, j). Cela représente $\mathrm{O}(n_{\text{total}}^2)$ calculs de $S(i, j)$ et autant de mises à jour $\Delta\omega$. D'un point de vue **big-O**, ce **carré** du nombre d'entités peut devenir prohibitif lorsque n_{total} s'avère important.

B. Détails sur l'impact du multimodal

Chaque **modalité** amène son **lot** d'entités :

- En **vision**, il est courant d'extraire une **multitude** de descriptors (patchs CNN, bounding boxes, keypoints…),
- En **audio**, on segmente la bande sonore en frames ou en fragments, chacun pouvant être répercuté dans le SCN,
- En **texte**, on peut gérer mots, tokens ou chunks, engendrant parfois des centaines de nœuds pour un seul document.

Au **total**, même si chaque modalité était en soi gérable (ex. 500 entités pour la vision, 500 pour l'audio, 1000 pour le texte), la **somme** peut rapidement s'envoler (ici 2000 entités). La complexité $\mathrm{O}(2000^2) \approx 4 \cdot 10^6$ pondérations est déjà non triviale, d'autant que le **calcul** de $S(i, j)$ peut lui-même nécessiter un **embedding** cross-modal.

En outre, le **SCN** inclut non seulement les liaisons $\omega_{i,j}$ entre entités de même modalité (ex. vision–vision), mais **aussi** les liaisons inter-modales (vision–audio, audio–texte, etc.). Cela multiplie encore les possibilités de couples (i, j). À chaque **itération**, on passe en revue

l'ensemble de la matrice ω, ce qui devient **lourd** si l'on conserve la formule classique $\omega_{i,j}(t+1) = \cdots$.

C. Perspective algorithmique

Dans la vision **naïve** où l'on met à jour toutes les paires (i,j), la charge $O(n_{\text{total}}^2)$ rend l'**itération** DSL extrêmement coûteuse. Cet état de fait **justifie** l'usage de méthodes d'**approximation** et d'**optimisation** (chap. 7). Parmi elles :

1. Sparsification

Ne conserver que les liens $\omega_{i,j}$ dépassant un certain seuil, ou limiter le calcul de $S(i,j)$ aux plus proches voisins (k-NN). Au lieu de mettre à jour $\omega_{i,j}$ pour chaque paire (i,j), on ne traite que celles dont la synergie potentielle semble prometteuse.

2. Heuristiques globales

Travailler par "blocs" ou "mini-batches" d'entités, réduire la dimension des embeddings via un autoencodeur commun, ou recourir à un **recuit** (voir section 7.3) appliqué sélectivement sur certaines parties du SCN.

3. Distribution sur plusieurs machines

Lorsque n_{total} est très grand, on peut **distribuer** les calculs, chaque nœud effectuant la mise à jour de sa section $\omega_{i,\cdot}$ ou $\omega_{\cdot,j}$. Ceci n'annule pas la complexité, mais la parallélise.

4. Filtrage amont

Avant même de construire le SCN, on peut filtrer les entités ou les paires improbables. Par exemple, on ne compare l'image et l'audio que si leurs timestamps coïncident dans une séquence vidéo, etc. On évite ainsi $O(n_{\text{img}} \times n_{\text{audio}})$ comparaisons inutiles.

D. Conséquences sur la dynamique et la qualité

En limitant $\omega_{i,j}$ aux seules paires "suspectées" de synergie (ou en imposant un "champ de vision" restreint), on sacrifie partiellement la possibilité de **découvrir** des correspondances imprévues. Cependant, en pratique, cet arbitrage est indispensable pour maîtriser le **coût** exponentiel. La **dynamique** DSL demeure valable sur la sous-partie "sparse" du graphe, et on obtient des **clusters** quasi équivalents à ceux du modèle complet, à condition de choisir un **filtrage** ou une **sparsification** judicieux.

8.2.3.3. Synchronisation Temporelle (Audio–Vidéo)

Lorsqu'il s'agit de **fusionner** des flux **audio** et **vidéo** dans un cadre **multimodal**, il ne suffit pas d'évaluer la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ sans tenir compte de la **dimension temporelle**. Un segment audio doit correspondre à la même portion de la séquence vidéo pour décrire un même événement, puisque les sons et les images doivent être **simultanés** pour être pertinents.

Cette problématique de **synchronisation** vient s'ajouter aux critères de **similarité** et de **compatibilité sémantique** déjà abordés (voir **section 8.2.1.2**). Il devient alors nécessaire d'intégrer la **variable temporelle** dans la fonction $S(\mathcal{E}_i, \mathcal{E}_j)$ ainsi que dans la mise à jour des pondérations $\omega_{i,j}$. Cette approche permet de former des **clusters multimodaux audio–vidéo**

réellement cohérents, en tenant compte à la fois des relations sémantiques et des alignements temporels.

A. Cadre Général de l'Alignement Temporel

Supposons que le flux **audio** soit découpé en trames $\{A_t\}$ d'une durée Δ_a chacune, tandis que le flux **vidéo** est découpé en images ou frames $\{V_\tau\}$ d'une durée Δ_v. Chaque trame audio A_t peut être considérée comme une **entité** $\mathcal{E}_{a,t}$ dans le **SCN**, et chaque frame vidéo V_τ comme une **entité** $\mathcal{E}_{v,\tau}$.

Pour définir la **synergie** $S(\mathcal{E}_{a,t}, \mathcal{E}_{v,\tau})$, il convient de tenir compte de la **proximité temporelle** : si $\tau\,\Delta_v$ et $t\,\Delta_a$ représentent des instants très éloignés, la probabilité que l'image et le son décrivent le même événement audiovisuel diminue. On peut formaliser :

$$S(\mathcal{E}_{a,t}, \mathcal{E}_{v,\tau}) = f\big(\text{similarité}_{\text{contenu}}(A_t, V_\tau),\ \text{décalage\_temps}(\tau\,\Delta_v - t\,\Delta_a)\big),$$

où $\text{similarité}_{\text{contenu}}(A_t, V_\tau)$ apprécie la **compatibilité sémantique** (par exemple, embedding audio vs. embedding vidéo), et décalage_temps pénalise les couples (t, τ) situés trop loin dans l'axe temporel.

B. Approche SCN : Pondérations et Décalage Temporel

Dans la **dynamique** DSL, chaque liaison $\omega_{(a,t),(v,\tau)}$ entre une trame audio $\mathcal{E}_{a,t}$ et une frame vidéo $\mathcal{E}_{v,\tau}$ subit la règle habituelle :

$$\omega_{(a,t),(v,\tau)}(k+1) = \omega_{(a,t),(v,\tau)}(k) + \eta\big[S(\mathcal{E}_{a,t}, \mathcal{E}_{v,\tau}) - \tau_d\ \omega_{(a,t),(v,\tau)}(k)\big],$$

où τ_d est la constante de décroissance (analogique à τ dans la formule DSL, à ne pas confondre avec l'index τ temporel de la vidéo). Le terme

$$S(\mathcal{E}_{a,t}, \mathcal{E}_{v,\tau})$$

se calcule en considérant :

- **La proximité temporelle** $\Delta T = |\tau\,\Delta_v - t\,\Delta_a|$. On peut pénaliser fortement les paires (t, τ) pour lesquelles ΔT dépasse un certain seuil.
- **La similarité des descripteurs** (spectrogramme vs. frame CNN, par exemple).

Une implémentation simple pourrait être :

$$S(A_t, V_\tau) = \cos\big(\phi_{\text{audio}}(A_t), \phi_{\text{vid}}(V_\tau)\big) \times \exp(-\alpha\ \Delta T),$$

avec un facteur gaussien $\exp(-\alpha\,\Delta T)$ minorant la synergie si A_t et V_τ sont trop décalés dans le temps.

C. Intégration d'Algorithmes d'Alignement plus Complexes

Lorsque l'audio et la vidéo ne sont pas strictement **échantillonnés** aux mêmes moments (ex. un flux vidéo à 30 FPS et un audio à 16 kHz), il existe plusieurs solutions :

- **Fenêtre glissante** : on recherche la **frame vidéo** la plus proche temporellement de la trame audio A_t. Les paires plus distantes sont ignorées ou reçoivent une synergie quasi nulle.

- **Dynamic Time Warping** (DTW) : si la vitesse d'enchaînement vidéo ou audio varie dans le temps (par ex. flux corrompu ou ralentissement), un algorithme DTW peut déterminer la "warp path" qui associe chaque trame audio à une frame vidéo la plus susceptible de correspondre, fournissant un **mapping** $(t \to \tau)$. Dans le SCN, on peut alors insérer un pénalty si (t, τ) s'écarte de cette warp path.

Le **DSL** s'appuie sur ces mécanismes pour **pondérer** la similarité purement sémantique ($\text{similarité}_{\text{contenu}}$) par un coefficient reflétant la **cohérence** temporelle. Ainsi, le **SCN** ne relie fortement un segment audio $\mathcal{E}_{a,t}$ qu'aux frames vidéo $\mathcal{E}_{v,\tau}$ qui cadrent en temps (et en contenu) avec lui.

D. Bénéfices pour l'Auto-Organisation

Un **SCN** multimodal où **audio** et **vidéo** s'alignent temporellement amène plusieurs avantages :

1. Clusters audio–vidéo de meilleure qualité

Chaque **événement** (bruit de voiture + vue de la voiture), chaque **action** (parole + mouvement labial) peut être regroupé, renforçant la synergie sur la période temporelle adéquate.

2. Désactivation automatique des liens hors synchronisation

Si un extrait audio se trouve 5 s avant la frame vidéo représentant la même scène, la synergie chute, donc ω ne se renforce pas. Au final, on obtient un **alignement** implicite des entités $(a, t) \leftrightarrow (v, \tau)$ partageant vraiment l'instant ou la courte fenêtre de temps.

3. Émergence de macro-clusters spatio-temporels

En combinant la notion de **similitude** de contenu et la **fenêtre** temporelle, le **DSL** peut construire un **super-nœud** représentant un événement multimédia. Celui-ci regroupe un ensemble de **frames vidéo**, un **segment audio** correspondant et, le cas échéant, un **bloc textuel** tel qu'un sous-titre ou une annotation insérée à la même période.

8.3. Modélisation des Entités Visuelles

Dans le cadre du **DSL multimodal**, la **composante visuelle** (images, vidéos) occupe une place essentielle. Les entités $\mathcal{E}_i$ peuvent être exclusivement visuelles ou associées à d'autres modalités comme le **texte** ou l'**audio**. Pour exploiter pleinement un **SCN** (Synergistic Connection Network) intégrant ces éléments, il est crucial de définir une **représentation** ou un **embedding** visuel adapté. Cette représentation doit garantir que la **synergie** $S\left(\text{image}_i, \text{image}_j\right)$ exprime fidèlement la **similarité** ou la **complémentarité** entre deux images.

8.3.1. Représentations d'Images

Au cours des dernières années, l'essor des **réseaux de neurones convolutifs** (CNN) a permis de générer des **embeddings** puissants et robustes pour décrire le **contenu** d'une image. Les sections suivantes détaillent comment on extrait ces vecteurs, comment ils se comparent à des approches plus classiques (points d'intérêt locaux, global pooling, etc.), et quels **formats** ou **dimensions** sont typiquement employés dans la littérature.

8.3.1.1. Embeddings issus de CNN (ex. ResNet, VGG)

Un **CNN** (Convolutional Neural Network) constitue, dans le cadre de la **vision par ordinateur**, l'un des moyens les plus éprouvés pour **extraire** des représentations vectorielles (ou *embeddings*) à partir d'une **image**. Ces embeddings se révèlent utiles pour un **DSL** (Deep Synergy Learning) appliqué à la modalité **image**, car ils fournissent un **vecteur** $\mathbf{f}_i \in \mathbb{R}^d$ permettant de comparer différentes images, de calculer la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ (voir chapitres précédents) et d'**auto-organiser** un **Synergistic Connection Network** (SCN) fondé sur la **similarité** ou la **distance** dans l'espace des **features**.

A. Principe des Convolutions et Extraction de Caractéristiques

Un **réseau** convolutif (CNN) Φ opère sur une **image** $\mathbf{x}$, typiquement un tenseur $\mathbf{x} \in \mathbb{R}^{H\times W\times 3}$ (RGB). Sur le plan **mathématique**, ce type d'architecture est une **composition** de :

- **Couches de convolution** : $\text{Conv}(\mathbf{x})$, appliquant des **noyaux** de taille $K \times K$ sur la carte de caractéristiques,
- **Fonctions d'activation** (ReLU, ELU, etc.),
- **Pooling** (max pooling, average pooling) pour réduire la dimension spatiale.

En notant σ la non-linéarité (ReLU, p. ex.) et Pool l'opération de pooling, on peut formaliser :

$$\mathbf{f} = \Phi(\mathbf{x}) = \text{Pool}\left(\sigma\left(\text{Conv}_K(\dots(\mathbf{x}))\right)\right).$$

Les convolutions successives extraient des **motifs visuels** (bords, textures, formes), et l'empilement des couches représente une **hiérarchie** de plus en plus abstraite. Ce processus aboutit à une **feature map** finale, ou un **vecteur** $\mathbf{f} \in \mathbb{R}^d$ lorsqu'on "aplatit" les dimensions spatiales.

B. Réseaux Pré-entraînés

Pour obtenir un **embedding** de qualité, la méthode la plus courante consiste à utiliser un **CNN** pré-entraîné sur un large corpus d'images. **ImageNet** (1.2 million d'images réparties en 1000 catégories) est un exemple classique. Des architectures comme **ResNet**, **VGG** ou **MobileNet** ont été **entraînées** à reconnaître ces classes, leur permettant d'extraire des **caractéristiques** génériques telles que les **textures**, les **contours** et les **détails d'objets**.

On récupère alors, dans le **DSL**, cette fonction Φ déjà apprise. Chaque **image x** se voit convertie en un **vecteur** $\mathbf{f} \in \mathbb{R}^d$:

$$\mathbf{f} = \Phi(\mathbf{x}),$$

souvent en extrayant la **dernière couche** avant la classification, notée par exemple "pool5" ou "fc7" selon l'architecture. Pour ResNet-50, la dimension $d \approx 2048$. Pour VGG16 (fc7), $d = 4096$. D'autres modèles (EfficientNet, MobileNet) ont des dimensions plus modestes (ex. 1280).

Ces **embeddings** se montrent relativement robustes et "généralistes", car ils capturent des **motifs** utiles pour reconnaître un large éventail de catégories. Dans un **SCN**, deux images similaires (même classe ou sujet visuel) reçoivent des vecteurs $\mathbf{f}_i$, $\mathbf{f}_j$ proches, conduisant à un **score** $S(i,j)$ élevé et favorisant la **cohésion** dans un cluster.

C. Opérations Complémentaires

Bien que ce vecteur $\mathbf{f}_i$ soit déjà une représentation, il peut nécessiter des **opérations** supplémentaires.

Réduction de dimension

Pour VGG16, le vecteur 4096-d peut être jugé trop grand. On applique une **PCA**, un **autoencodeur** ou une simple projection linéaire $\mathbf{U} \in \mathbb{R}^{d' \times d}$ pour ramener $\mathbf{f}_i$ en $\mathbf{g}_i \in \mathbb{R}^{d'}$. Cela **allège** aussi le calcul de $S(i,j)$.

Normalisation

En DSL, il est très courant de **normaliser** chaque embedding à la norme Euclidienne 1 :

$$\mathbf{f}_i \leftarrow \frac{\mathbf{f}_i}{\| \mathbf{f}_i \|_2}$$

afin de faciliter une **similarité cosinus** (également interprétée comme un **produit scalaire**). Un vecteur normalisé $\mathbf{f}_i$ aide à stabiliser les mesures de distance ou les kernels RBF, et simplifie la **dynamique** de la synergie.

8.3.1.2. Caractéristiques locales (SIFT, SURF) vs. globales (pooling)

Dans le **Deep Synergy Learning (DSL)** appliqué à la modalité **visuelle**, le choix des **descripteurs** associés à chaque image est une question centrale. Faut-il privilégier des

caractéristiques locales, extraites autour de points d'intérêt et codées par des vecteurs comme **SIFT** ou **SURF**, ou bien résumer l'image en un **vecteur global** obtenu par un *pooling* (moyen ou max) sur l'ensemble de la carte de caractéristiques ?

Cette décision ne relève pas uniquement d'un choix pratique. Elle impacte directement la **précision** de la mesure de **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ et influe sur la **complexité** du **Synergistic Connection Network (SCN)**.

Les sections précédentes (8.3.1.1) ont montré comment un **CNN** (ResNet, VGG, etc.) pouvait extraire des embeddings globaux. Il reste néanmoins pertinent de discuter la représentation **locale** (points clés) qui peut, dans un cadre DSL, coexister ou être combinée à ces embeddings globaux.

A. Caractéristiques locales : SIFT, SURF, etc.

Les descripteurs dits *locaux* se concentrent sur des **points d'intérêt** détectés dans l'image, chacun aboutissant à un vecteur décrivant sa texture et son orientation dans un voisinage restreint. Les méthodes **SIFT** (Scale-Invariant Feature Transform) et **SURF** (Speeded-Up Robust Features) sont parmi les plus connues.

SIFT, par exemple, cherche des **extrema d'échelle** et de position dans l'espace de différences de gaussiennes, ce qui permet de localiser un point clé (x_k, y_k) approximativement invariant aux changements d'échelle et de rotation. Autour de ce point clé, on calcule un **descripteur** $\mathbf{d}_k \in \mathbb{R}^{128}$ en subdivisant un patch en cellules. Sur le plan mathématique, si l'on note $\nabla I(x, y)$ le gradient du niveau de gris au pixel (x, y) et $\theta_{x,y}$ son orientation, alors, pour chaque cellule $c \subset \Omega$ (où Ω est le voisinage du point clé), on accumule un histogramme $\mathrm{Hist}_c(\theta)$:

$$\mathrm{Hist}_c(\theta) = \sum_{(x,y)\in c} w\,(x, y)\,\delta\big(\theta_{x,y},\,\theta\big),$$

où $w(x, y)$ est un poids (généralement une gaussienne centrée au point clé) et θ parcourt une discrétisation en 8 directions principales. L'assemblage de ces histogrammes sur 4×4 cellules fournit un vecteur final de taille 128. SURF emploie un principe similaire mais recourt à des ondelettes de Haar intégrales pour constituer un vecteur plus compact (64 ou 128 dimensions).

Dans un **DSL**, on peut considérer chaque point clé $\mathbf{d}_{i,k}$ de l'image $\mathcal{I}_i$ comme une entité $\mathcal{E}_{i,k}$. La **synergie** $S(\mathcal{I}_i, \mathcal{I}_j)$ entre deux images peut alors s'obtenir via un *matching* local. Une manière de la formaliser est de prendre le max sur tous les couples (k, ℓ) :

$$S(\mathcal{I}_i, \mathcal{I}_j) = \max_{k,\ell} \mathrm{Sim}\big(\mathbf{d}_{i,k},\,\mathbf{d}_{j,\ell}\big),$$

où Sim peut être un produit scalaire ou un ℓ_2 inversé, en notant que la normalisation de $\mathbf{d}_{i,k}$ peut simplifier la comparaison. Alternativement, on peut additionner ces similarités si l'on veut un score plus global.

L'un des principaux **avantages** des **descripteurs locaux** est leur **robustesse** aux transformations partielles, notamment lorsque seule une portion d'image correspond à une autre. Leur **invariance** aux changements d'**échelle** et aux **rotations légères** les rend adaptés aux tâches de reconnaissance.

En revanche, le **coût** devient significatif. Si chaque image comporte un grand nombre de **points clés**, la dimension du **SCN** augmente, et la dynamique $O(n^2)$ (où n est le nombre total d'entités) peut être difficile à gérer, comme l'illustrent les considérations du **chapitre 7**.

B. Caractéristiques globales : pooling ou représentation agrégée

À l'opposé, on peut vouloir un **vecteur** unique décrivant l'ensemble de l'image. Lorsque l'on exploite un réseau de neurones convolutifs (ResNet, VGG), chaque pixel aboutit, dans les couches profondes, à une **feature map** en (W, H, C). On peut alors appliquer un *global average pooling* (ou un *global max pooling*) qui, pour chaque canal c :

$$g_c = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} f_c\,(x, y),$$

où $f_c(x, y)$ sont les valeurs de la feature map dans le canal c. On aboutit alors à un vecteur $\mathbf{g} \in \mathbb{R}^C$ qui constitue une **représentation globale** de l'image $\mathcal{I}$. Cette approche se traduit, dans un SCN, par la création d'une entité $\mathcal{E}_i$ associée au vecteur $\mathbf{g}_i$. La **similarité** se mesure facilement (cosinus, RBF) et la mise à jour $\omega_{i,j}$ se fait par la formule DSL usuelle.

L'**avantage** est une **grande simplicité** : un seul vecteur par image, des comparaisons rapides et une robustesse face à de légères variations. L'**inconvénient** est que si deux images ne coïncident que localement, par exemple avec un petit objet identique sur des fonds différents, le *pooling* global peut atténuer ou masquer cette différence ou cette similarité.

C. Comparaison locale vs. globale dans un DSL

Le **DSL** peut aisément intégrer ces deux échelles de description, permettant un ajustement par la **dynamique** de la mise à jour ω. On peut ainsi associer, pour chaque image $\mathcal{I}_i$, un ensemble $\{\mathbf{d}_{i,k}\}_{k=1\ldots K_i}$ de *keypoints* (nœuds "locaux") et un *embedding* global $\mathbf{g}_i$ (nœud "global"). La **synergie** $S(\mathcal{I}_i, \mathcal{I}_j)$ peut alors se définir comme une combinaison :

$$S(\mathcal{I}_i, \mathcal{I}_j) = \alpha \max_{k,\ell} \mathrm{Sim}(\mathbf{d}_{i,k}, \mathbf{d}_{j,\ell}) + (1 - \alpha)\,\mathrm{Sim}(\mathbf{g}_i, \mathbf{g}_j),$$

ou toute autre fonction similaire, où $\alpha \in [0,1]$ pèse l'importance relative du *matching* local et du *matching* global. Cette souplesse s'intègre bien dans la logique d'un **SCN** multi-échelle (cf. chap. 6), où l'**auto-organisation** peut pointer, dans certains cas, vers des correspondances précises de patchs, et dans d'autres, vers un accord global d'images entières.

8.3.1.3. Avantages de normaliser, dimension typique (256, 512, etc.)

Dans le cadre du **Deep Synergy Learning** (DSL) appliqué aux données visuelles, la représentation de chaque entité par un **vecteur** (embedding) soulève deux interrogations. La première concerne le **choix** de la **dimension** de ce vecteur (256, 512, voire 1024). La seconde se rapporte à la **nécessité** de normaliser lesdits vecteurs (c'est-à-dire leur imposer une norme ou un écart-type constant). Ces considérations influencent à la fois la **richesse** de la représentation et la **stabilité** de la dynamique auto-organisée au sein du **Synergistic Connection Network** (SCN).

A. Dimension typique : 256, 512, etc.

Une dimension de quelques centaines, comme 256 ou 512, s'avère souvent un **compromis** pertinent. Il est souhaitable que le vecteur $\mathbf{v}_i \in \mathbb{R}^d$ soit suffisamment grand pour encoder la **variété** des caractéristiques de l'entité visuelle (couleurs, formes, textures, etc.), mais pas au point de rendre le calcul $O(n^2\, d)$ inabordable pour un SCN de grande taille. Sur le plan

mathématique, si l'on considère deux entités $\mathcal{E}_i$ et $\mathcal{E}_j$ dotées de vecteurs $\mathbf{v}_i, \mathbf{v}_j \in \mathbb{R}^d$, le **produit scalaire** $\mathbf{v}_i \cdot \mathbf{v}_j$ (ou la distance euclidienne $\| \mathbf{v}_i - \mathbf{v}_j \|$) gagne en **précision** lorsqu'on dispose d'un espace de dimension plus grande. On peut, par exemple, définir la **synergie**

$$S(i,j) = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\| \mathbf{v}_i \| \, \| \mathbf{v}_j \|}$$

dans le cas d'une similarité cosinus. Une dimension de l'ordre de 512 peut ainsi mieux capturer les multiples facettes de l'image, en opposition à un vecteur de très faible taille (ex. 16 ou 32) qui risquerait de manquer de **discriminabilité**. À l'inverse, des dimensions trop élevées (1024 ou plus) peuvent conduire à un surcoût de stockage et de calcul, et la "malédiction de la dimension" peut diluer la pertinence des distances.

B. Avantages de normaliser les vecteurs

La seconde préoccupation porte sur la **normalisation** de $\mathbf{v}_i$. Il est fréquent, dans un **SCN**, de chercher à aligner la norme de chaque embedding à une valeur fixe, par exemple 1, ce qui place $\mathbf{v}_i$ sur la **sphère unité** $\mathcal{S}^{d-1}$.

Lorsque les vecteurs sont tous normalisés, la similarité cosinus

$$S(i,j) = \mathbf{v}_i \cdot \mathbf{v}_j \quad \text{(puisque } \| \mathbf{v}_i \| = \| \mathbf{v}_j \| = 1\text{)},$$

se simplifie et présente une **échelle** de valeurs plus stable. On évite ainsi que certaines entités, possédant un embedding de norme très élevée, dominent le calcul. Dans la dynamique DSL $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big]$, cette borne entre -1 et +1 facilite le réglage des paramètres η et τ.

L'usage de la sphère unité prévient des problèmes liés à des vecteurs de norme extrême, et la distance ou le produit scalaire entre vecteurs normalisés exhibent un comportement plus régulier. On peut, par ailleurs, décider d'un petit offset pour forcer la positivité (entre 0 et 1), par exemple en appliquant $1 + \mathbf{v}_i \cdot \mathbf{v}_j/2$, mais la démarche la plus courante reste de s'en tenir à $[-1, +1]$.

On peut réécrire la mise à jour de $\omega_{i,j}$ lorsque l'on s'appuie sur la normalisation. Si $\| \mathbf{v}_i \| = 1$ et $\| \mathbf{v}_j \| = 1$, alors

$$S(i,j) = \mathbf{v}_i \cdot \mathbf{v}_j.$$

La suite $\{\omega_{i,j}(t)\}$ évolue dans un intervalle restreint selon la valeur du produit scalaire, consolidant ainsi la **stabilité** de la dynamique.

8.3.2. Synergie Visuelle-Visuelle

La **synergie** entre deux **images** (par ex. image_i et image_j) peut se définir via une mesure de **similarité** ou de **distance** au sein d'un espace d'embrayage visuel. Dans la perspective du **DSL** (Deep Synergy Learning), on attribue une pondération $\omega_{i,j}$ reflétant la force de l'association entre image_i et image_j. Au préalable, on doit donc **calculer** la synergie

$$S\left(\text{image}_i, \text{image}_j\right) \in \mathbb{R}$$

qui servira de “signal” pour le renforcement ou l’inhibition des liens visuels dans le **SCN**.

8.3.2.1. Calcul de $S\left(\text{image}_i, \text{image}_j\right)$: cosinus, distance euclidienne

Les entités **visuelles** dans un **Deep Synergy Learning** (DSL) sont souvent codées par des **vecteurs** extraits d’images, appelés *embeddings*, qu’ils proviennent de descripteurs classiques (SIFT, SURF, HOG) ou de représentations profondes (CNN, ViT, etc.). Une fois ces vecteurs définis, le **Synergistic Connection Network** (SCN) dispose d’un outil pour mesurer la **synergie** entre deux images image_i et image_j via une fonction $S\left(\text{image}_i, \text{image}_j\right)$. Le plus souvent, cette synergie repose sur une **distance** ou une **similarité** dans l’espace des embeddings.

Lorsque l’on compare deux images image_i et image_j, on commence par les **représenter** chacune sous forme d’un **vecteur** $\mathbf{v}_i, \mathbf{v}_j \in \mathbb{R}^d$. Selon le scénario, ces vecteurs peuvent provenir de méthodes variées. Certains environnements privilégient des *features* “classiques” comme SIFT, SURF, ou HOG, d’autres exploitent un réseau de neurones convolutionnel (ResNet, VGG…) dont on récupère la dernière couche “fully-connected” ou un “global average pooling” (GAP) pour aboutir à un vecteur $\mathbf{v}$. Il est également possible de combiner plusieurs sources, par exemple des canaux de couleur et de texture. Sur le plan **mathématique**, la conclusion est que chaque image image_i est associée à un vecteur $\mathbf{v}_i \in \mathbb{R}^d$. Pour tout couple (i, j), on doit donc définir une fonction

$$S\left(\text{image}_i, \text{image}_j\right) = f\left(\mathbf{v}_i, \mathbf{v}_j\right),$$

et la plus grande liberté est laissée au concepteur pour choisir la forme de cette fonction.

La méthode la plus directe pour quantifier la **similarité** entre deux vecteurs consiste à recourir à la **similarité cosinus** ou à la **distance euclidienne**.

Dans ce cas, on pose

$$S\left(\text{image}_i, \text{image}_j\right) = \cos\left(\mathbf{v}_i, \mathbf{v}_j\right) = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\| \mathbf{v}_i \| \, \| \mathbf{v}_j \|}.$$

Si les embeddings sont préalablement normalisés (voir section 8.3.1.3) afin que $\| \mathbf{v}_i \| = 1$, alors la similarité cosinus s’identifie simplement au produit scalaire $\mathbf{v}_i \cdot \mathbf{v}_j$. Un score voisin de 1 signale deux images jugées très proches, tandis qu’un score proche de -1 (pour un cosinus brut) ou 0 (pour un cosinus borné positivement) indique une faible correspondance.

Une autre possibilité est de recourir à la **distance** $\mathrm{d}\left(\mathbf{v}_i, \mathbf{v}_j\right)$, où la plus courante est la norme ℓ_2. On peut ensuite définir

$$S\left(\text{image}_i, \text{image}_j\right) = \frac{1}{1 + \mathrm{d}\left(\mathbf{v}_i, \mathbf{v}_j\right)},$$

ou l’on peut préférer un noyau gaussien

$$S\left(\text{image}_i, \text{image}_j\right) = \exp\left(-\alpha \, \| \mathbf{v}_i - \mathbf{v}_j \|^2\right),$$

selon l'esprit d'un **RBF-kernel**. L'idée est similaire. Plus la distance est petite, plus le score S est grand, ce qui signale une plus grande proximité entre les deux images.

Dans bien des applications, il est bénéfique de **normaliser** chaque $\mathbf{v}_i$. En appliquant $\mathbf{v}_i \leftarrow \mathbf{v}_i / \| \mathbf{v}_i \|$, on simplifie la formule du cosinus en un simple produit scalaire. Cette pratique unifie l'échelle des embeddings et empêche qu'un vecteur de très forte norme "fausse" les calculs de similarité, ce qui est important lorsque la mise à jour $\omega_{i,j}$ dépend du score S. D'un point de vue purement **mathématique**, cette approche revient à projeter $\mathbf{v}_i$ sur la sphère unité $\mathcal{S}^{d-1}$, où les distances ou produits scalaires peuvent être plus aisément comparés. Il est également courant de borner la valeur de S à $[0,1]$ en appliquant un offset, par exemple

$$\tilde{S}(i,j) = \frac{1 + \cos(\mathbf{v}_i, \mathbf{v}_j)}{2}.$$

Cette transformation s'intègre parfaitement dans la logique du DSL, où l'on aime manipuler un score positif.

Le **DSL** ne fixe pas de norme unique. Il n'exige qu'une fonction S renvoyant un score localement "cohérent" avec la notion d'entités semblables ou dissemblables. En pratique, l'on retrouve principalement le cosinus ou l'inverse d'une distance euclidienne, mais rien n'empêche de définir une mesure plus sophistiquée (ex. un **kernel** non linéaire ou une **information mutuelle** si l'on traite des distributions). L'essentiel est d'obtenir un **score** qui pointe vers "haute synergie" si deux images sont estimées proches, et "faible synergie" sinon, ce qui oriente la dynamique $\omega_{i,j}$ pour renforcer ou diminuer les liens.

A. Rôle du score : de la similarité à la synergie

Une fois la fonction $S\left(\text{image}_i, \text{image}_j\right)$ définie, on l'incorpore dans la **règle** de mise à jour DSL :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta \left[S\left(\text{image}_i, \text{image}_j\right) - \tau\, \omega_{i,j}(t) \right].$$

Cela signifie qu'un **score** élevé pousse $\omega_{i,j}$ à croître, reflétant la proximité entre les deux images. Au fil du temps, un **SCN** voit se former des **clusters** de liaisons robustes parmi les entités jugées proches, tandis que d'autres liens décroissent. Ce mécanisme **auto-organisé** crée une partition ou une structuration reflétant la **cohérence** visuelle.

B. Détails mathématiques et normalisation

Il existe différentes façons de paramétrer cette fonction. Par exemple, on peut introduire un facteur α dans le RBF-kernel :

$$S\left(\text{image}_i, \text{image}_j\right) = \exp\left(-\alpha \, \| \mathbf{v}_i - \mathbf{v}_j \|^2\right).$$

Avec une telle fonction, les images très proches (faible $\| \mathbf{v}_i - \mathbf{v}_j \|$) reçoivent un score proche de 1, tandis que les images distantes ont un score décroissant exponentiellement. Dans le cas d'un cosinus, on peut corriger la fourchette de valeur $[-1, +1]$ pour la replacer dans $[0,1]$. D'un point de vue **numérique**, il est souvent pratique de **normaliser** $\mathbf{v}_i$ pour aboutir à un intervalle fixe, ce qui évite tout réglage supplémentaire dans l'équation DSL.

C. Cas d'images brutes vs. embeddings profonds

Historiquement, on aurait pu comparer deux images en traitant directement les pixels (par exemple, distance L2 entre leur contenu brut), mais cela se heurte rapidement à des problèmes d'invariance (lumière, rotation, etc.), et la dimension devient énorme (un vecteur 128×128×3 = 49152 pour une image en 128×128 couleur). Dans la pratique contemporaine, on exploite davantage des **embeddings** issus d'un réseau profond (cf. chap. 8.3.1.1) pour aboutir à un vecteur $\mathbf{v}_i \in \mathbb{R}^d$ de taille plus raisonnable (128, 256, 512, etc.) et plus **représentatif** sémantiquement. Cette évolution mathématique facilite le calcul de S et rend la convergence du SCN plus stable.

8.3.2.2. Approche par hashage (LSH) pour accélérer la similarité d'images

Dans les applications **multimodales** du **Deep Synergy Learning** (DSL) où l'on doit traiter un nombre conséquent d'entités visuelles (pouvant se chiffrer en milliers ou en millions), la comparaison exhaustive de toutes les paires $(\mathcal{E}_i, \mathcal{E}_j)$ mène à une complexité en $O(n^2)$. Cette explosion pose un problème majeur lorsqu'on désire construire un **Synergistic Connection Network** (SCN) sur un large ensemble d'images. Pour remédier à cela, on recourt souvent à des **techniques de hashage** localement sensible, dites **LSH** (*Locality-Sensitive Hashing*), qui visent à **rapprocher** les entités potentiellement semblables dans des "buckets" ou codes communs, et à s'abstenir de comparer explicitement toutes les paires.

A. Principe général du LSH

Le **LSH** repose sur l'idée qu'il existe des **fonctions** de hachage, notées $h: \mathbb{R}^d \to \mathcal{C}$, ayant la propriété suivante. Si $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^d$ sont proches, par exemple selon une norme $\| \mathbf{x}_i - \mathbf{x}_j \|$ ou un angle cosinus, alors la probabilité que $h(\mathbf{x}_i) = h(\mathbf{x}_j)$ est élevée. L'objectif mathématique est :

$$\| \mathbf{x}_i - \mathbf{x}_j \| \text{ petite} \quad \Rightarrow \quad \text{Prob}\left[h(\mathbf{x}_i) = h(\mathbf{x}_j)\right] \approx 1,$$

et inversement, si $\| \mathbf{x}_i - \mathbf{x}_j \|$ est grande, la collision de leurs codes devient improbable.

Dans la pratique, on ne se contente pas d'une unique fonction de hachage, mais on en définit plusieurs, notées $h_1, h_2, \ldots, h_L$. Les résultats de ces fonctions sont concaténés en un **code** ou un **bucket**. Deux entités $\mathbf{x}_i, \mathbf{x}_j$ se retrouvent alors rangées dans le même bucket si $\left(h_1(\mathbf{x}_i), \ldots, h_L(\mathbf{x}_i)\right) = \left(h_1(\mathbf{x}_j), \ldots, h_L(\mathbf{x}_j)\right)$. L'effet combiné est de **discriminer** plus finement les entités proches de celles qui ne le sont pas.

Lorsque l'on souhaite calculer la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ dans un **SCN**, on peut ne se concentrer que sur les paires dont les codes LSH sont identiques ou suffisamment proches. On passe ainsi d'une complexité naïve $O(n^2)$ à un processus plus ciblé, qui compare bien moins de paires.

B. Exemples de fonctions LSH pour images

Les algorithmes de LSH diffèrent selon la **distance** ou la **mesure** qu'on veut préserver.

Lorsqu'on emploie la **similarité cosinus**, on peut définir des hyperplans aléatoires $\{\mathbf{r}_k\}_{k=1}^{L} \subset \mathbb{R}^d$. Pour un vecteur $\mathbf{x}$, on prend un bit $b_k = \text{sgn}(\mathbf{r}_k \cdot \mathbf{x})$. Les L bits forment alors un code binaire de longueur L. Deux vecteurs $\mathbf{x}_i, \mathbf{x}_j$ proches en angle ont de grandes chances de donner la même suite $(b_1, \ldots, b_L)$. On retrouve ainsi la propriété de hachage localement sensible.

Dans les applications de “hachage perceptuel” d’images, on peut redimensionner et convertir chaque image (ou son embedding) via une transformée en cosinus discrète (DCT), puis binariser les coefficients les plus bas ou la médiane. Cela donne un code binaire (ex. 64 bits), où deux images similaires (au sens perceptuel) se retrouveront souvent avec le même code.

Pour représenter une image comme un ensemble (ou un multi-ensemble) de *features* (par exemple, un sac de mots visuels), on peut appliquer des permutations aléatoires et prendre le plus petit index, un concept appelé **minwise hashing**. Les ensembles présentant une grande intersection partagent plus fréquemment le même hash.

C. Intégration dans un DSL

Dans un **SCN** gérant une foule d’images $\{\mathcal{E}_i\}_{i=1\dots n}$, on souhaite limiter le calcul explicite de la fonction $S(\mathcal{E}_i, \mathcal{E}_j)$ à des paires “pertinentes”. On peut alors indexer chaque embedding $\mathbf{x}_i \in \mathbb{R}^d$ dans une structure LSH. Lorsqu’on insère la nouvelle entité $\mathbf{x}_i$, on détermine son code LSH et on ne compare $\mathbf{x}_i$ qu’aux entités déjà présentes dans le même bucket (ou un voisinage restreint en code). De cette façon, on se concentre sur des **candidats** de haute similarité potentielle, passant la complexité d’un $O(n^2)$ à quelque chose plus proche de $O(n \cdot \text{avgBucketSize})$, ce qui peut s’avérer beaucoup plus tractable.

Dans la **dynamique** du DSL, cela signifie qu’au lieu de parcourir exhaustivement tous les index j pour une entité i et de mettre à jour $\omega_{i,j}(t+1)$, on parcourt seulement les paires (i,j) dont les codes LSH s’avèrent proches (potentiellement identiques). On résout ainsi la mise à jour $\omega_{i,j}(t+1) = \cdots$ uniquement sur ce sous-ensemble réduit, ce qui accélère la formation d’un **SCN** de grande échelle.

D. Aspects mathématiques et erreurs LSH

Le **LSH** procure un **contrôle** probabiliste. Deux entités réellement proches, au sens de $\| \mathbf{x}_i - \mathbf{x}_j \|$ ou $\mathbf{x}_i \cdot \mathbf{x}_j$, ont une forte probabilité d’**être** placées dans le même code, tandis que deux entités très distantes ne le sont qu’avec une probabilité faible.

Cependant, des **faux négatifs** peuvent survenir lorsque des entités proches sont assignées à des codes différents, et des **faux positifs** apparaissent lorsque des entités peu similaires se retrouvent dans le même code. Dans un **DSL**, cela peut entraîner soit un **manque** dans la mise à jour de $\omega_{i,j}$ pour certaines paires pertinentes, soit une **comparaison superflue** de paires non significatives. Tant que ces probabilités restent faibles, la réduction de complexité due à un **moindre nombre de paires testées** compense ces inconvénients.

8.3.2.3. Cas d’un flux vidéo : segmentation en frames ou analyse spatio-temporelle ?

La **vidéo** constitue un exemple clé de la dimension **multimodale** examinée dans la section 8.3.2, où la notion de **temps** s’ajoute à l’espace purement visuel de chaque image. Lorsqu’on souhaite construire un **Synergistic Connection Network** (SCN) pour un **flux vidéo**, on peut adopter un modèle où l’on segmente la séquence en *frames* indépendantes (comme autant de “photos” séparées) ou, au contraire, essayer de capturer la **dynamique spatio-temporelle** qui relie ces images consécutives. Le **Deep Synergy Learning** (DSL) reste flexible quant aux choix de représentation, mais la complexité mathématique et la pertinence des **clusters** ou **entités** créés peuvent différer considérablement.

A. Segmentation en frames simples

Un choix classique et simplifié consiste à traiter la vidéo "frame par frame". Dans ce cas, chaque image $\mathcal{E}_t$ correspond à la vue au temps $t\Delta t$. Sur le plan **mathématique**, on dispose d'un vecteur $\mathbf{v}_t \in \mathbb{R}^d$ pour chaque frame, obtenu typiquement par un **CNN** (ResNet, VGG...) ou un autre extracteur visuel. Le **DSL** manipule alors un ensemble $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$, où n est le nombre de frames retenues. La **synergie** $S(\mathcal{E}_t, \mathcal{E}_{t'})$ peut être définie, par exemple, comme une **similarité** cosinus ou l'inverse d'une **distance** $\| \mathbf{v}_t - \mathbf{v}_{t'} \|$.

Le principal intérêt de cette segmentation réside dans sa **simplicité de mise en œuvre**. On applique les techniques de traitement d'images statiques pour extraire des **embeddings** à chaque frame de la vidéo, puis on alimente le **SCN**. Le **DSL** ne traite alors que des images considérées comme indépendantes. Cette approche est adaptée lorsque l'objectif est un **échantillonnage léger** du flux, par exemple en sélectionnant une image toutes les dix frames pour obtenir un aperçu global.

Sur le plan mathématique, la charge de calcul dépend du nombre total de **frames**. Chaque itération du **DSL** s'exécute en $O(n^2)$ si toutes les paires (t, t') sont comparées. Cette multiplication du nombre de frames peut entraîner des problèmes de **complexité** (chapitre 7), nécessitant l'emploi d'**heuristiques** pour limiter la comparaison exhaustive.

Cette approche *frame par frame* ne prend pas en compte la **continuité temporelle**. Deux frames successives partagent souvent des motifs quasi identiques, ce qui peut entraîner la formation de **clusters dominants** entre frames proches, sans exploiter explicitement l'information de mouvement. Elle devient donc inadaptée lorsque l'objectif est d'identifier des **événements dynamiques** tels que des gestes ou des actions qui s'étendent sur plusieurs frames.

B. Analyse spatio-temporelle

Une seconde option plus riche modélise la vidéo de manière spatio-temporelle. Au lieu de ne considérer que l'espace (x, y) pour un instant t, on tient compte également de la dimension temps. On peut alors découper la vidéo en "volumes" (blocs (x, y, t)) ou **patchs 3D**. Sur le plan **mathématique**, chaque entité $\mathcal{E}_\alpha$ correspond à un sous-cube de la séquence, par exemple 16 frames contiguës sur 112×112 pixels, pour lequel on calcule un **embedding** 3D (réseau CNN 3D, S3D, I3D, etc.) ou un **modèle** spatio-temporel plus complexe.

La **synergie** $S(\mathcal{E}_\alpha, \mathcal{E}_\beta)$ devient une mesure de similarité entre deux volumes, par exemple un cosinus entre leurs vecteurs $\mathbf{u}_\alpha, \mathbf{u}_\beta \in \mathbb{R}^d$. Cela permet de capturer le **mouvement** et la **dynamique** au sein de la séquence. Deux actions similaires, comme un saut ou un coup de raquette, partagent un **embedding spatio-temporel** plus élevé.

Cette approche est particulièrement efficace pour reconnaître des **événements** se déroulant sur plusieurs frames. Toutefois, elle entraîne une **augmentation de la dimension** du descripteur, car chaque entité devient un vecteur 3D plus volumineux. Le nombre total d'entités considérées s'accroît également, puisque tous les blocs potentiels α doivent être pris en compte. Cela amplifie la **complexité** de la mise à jour DSL, qui suit une dynamique $O(n^2)$.

C. Conséquences mathématiques

Le passage à un espace spatio-temporel modifie la définition de S. Par exemple, on peut écrire

$$S(\mathcal{E}_\alpha, \mathcal{E}_\beta) = \lambda_1 \,\mathrm{sim}\left(\mathbf{v}_\alpha^{(\text{apparence})},\ \mathbf{v}_\beta^{(\text{apparence})}\right) + \lambda_2 \,\mathrm{sim}\left(\mathbf{v}_\alpha^{(\text{mouvement})},\ \mathbf{v}_\beta^{(\text{mouvement})}\right),$$

afin de coupler l'aspect *apparence* (les frames) et l'aspect *dynamique* (le flot optique, l'embedding 3D). Le **DSL** manipule ensuite la règle

$$\omega_{\alpha,\beta}(t+1) = \omega_{\alpha,\beta}(t) + \eta\big[S(\mathcal{E}_\alpha, \mathcal{E}_\beta) - \tau\,\omega_{\alpha,\beta}(t)\big],$$

ce qui construit un **SCN** reflétant non seulement la ressemblance spatiale, mais également la **cohérence** temporelle. Les coûts croissent, car la quantité d'entités (blocs spatio-temporels) peut être très importante, invitant à la prudence (éventuelles méthodes heuristiques ou approximations, cf. chap. 7).

D. Choix entre segmentation en frames ou spatio-temporel

Le choix de l'approche dépend de l'**objectif** visé. Pour des tâches de repérage ou de détection de scènes globales, la segmentation *frame par frame* peut suffire. Cette méthode ignore l'information de mouvement mais est plus simple et rapide à mettre en place. En revanche, pour identifier des **actions dynamiques** qui s'étendent sur plusieurs instants, il devient nécessaire d'intégrer la **dimension temporelle** dans la représentation. Cela implique soit un **embedding spatio-temporel**, soit une **agrégation de frames successives** pour capter les dynamiques sous-jacentes.

Une approche **hybride** peut aussi être envisagée. On segmente d'abord en frames ou en micro-blocs, puis une **analyse plus approfondie** est déclenchée uniquement sur les segments jugés pertinents. Sur le plan **mathématique**, cela revient à faire coexister, au sein du **SCN**, des **entités "frame"** et des **entités "blocs 3D"**, avec une définition cohérente de la **synergie** entre ces différentes représentations.

8.3.3. Liens avec d'Autres Modalités

Lorsque l'on parle de **DSL Multimodal**, on ne se limite pas à un seul couplage (vision + audio, par exemple) ; au contraire, il est fréquent d'**enchaîner** ou de **composer** plusieurs modalités. La **synergie** peut alors s'étendre entre la **vision**, le **texte**, l'**audio**, ou d'autres flux (capteurs, données symboliques), chacun interagissant pour renforcer la **compréhension** globale. Dans cette section 8.3.3, nous mettons l'accent sur la **visée** du DSL à l'heure de **concilier** (ou de relier) la vision avec d'autres flux, en particulier le **texte** et l'**audio**, afin de clarifier comment le **SCN** (Synergistic Connection Network) fusionne des informations potentiellement très différentes (images, séquences textuelles, signaux sonores).

8.3.3.1. Vision–Texte : ex. l'étiquette textuelle d'une image (caption)

L'association entre **vision** et **texte** constitue l'une des illustrations les plus abouties de la **multimodalité** évoquée dans les sections précédentes. Dans un **Deep Synergy Learning** (DSL) appliqué à un **Synergistic Connection Network** (SCN), il est fréquent d'introduire deux **ensembles** d'entités, l'un relatif aux **descripteurs** (embeddings) issus d'**images**, l'autre associé aux **segments textuels** (mots, étiquettes, phrases). Les liaisons $\omega_{i,j}$ entre $\mathcal{V}_i$ (vision) et $\mathcal{T}_j$ (texte) se renforcent dès que la **synergie** $S(\mathcal{V}_i, \mathcal{T}_j)$ révèle une adéquation marquée. Ce mécanisme permet, par exemple, de faire émerger un **couplage** entre une image et son label ("cat"), ou encore une légende plus détaillée.

A. Définition mathématique de la synergie Vision–Texte

Pour relier une **image** (ou un patch) $\mathcal{V}_i$ à un **extrait textuel** $\mathcal{T}_j$, il s'avère utile de disposer de deux **embeddings** dans un espace commun ou partiellement commun. Dans des approches récentes (type CLIP), on apprend des projections $\phi_{\text{img}}(\mathbf{v}_i)$ et $\phi_{\text{txt}}(\mathbf{t}_j)$ qui convergent vers un espace latent $\mathbb{R}^d$. Le score de **similarité** peut alors se définir comme :

$$S(\mathcal{V}_i, \mathcal{T}_j) = \cos\left(\phi_{\text{img}}(\mathbf{v}_i),\ \phi_{\text{txt}}(\mathbf{t}_j)\right) = \frac{\phi_{\text{img}}(\mathbf{v}_i)\ \cdot\ \phi_{\text{txt}}(\mathbf{t}_j)}{\|\ \phi_{\text{img}}(\mathbf{v}_i)\ \|\ \|\ \phi_{\text{txt}}(\mathbf{t}_j)\ \|}.$$

Plus ce cosinus est élevé (proche de 1), plus le modèle estime que le texte correspond visuellement (et sémantiquement) à l'image.

Le **Synergistic Connection Network** peut aussi manipuler plusieurs *patchs* pour une image et plusieurs *mots* pour un texte. Chaque entité visuelle $\mathcal{V}_{i,k}$ décrit un patch ou un objet détecté, chaque entité textuelle $\mathcal{T}_{j,\ell}$ décrit un token ou un segment. La **synergie** S entre un patch $\mathcal{V}_{i,k}$ et un mot $\mathcal{T}_{j,\ell}$ peut se formaliser via un cosinus, un RBF-kernel, ou une fonction quelconque de distance. On obtient ensuite un **cluster** combinant plusieurs patchs et mots si la dynamique auto-organisée du DSL accroît fortement les liaisons $\omega_{(i,k),(j,\ell)}$.

B. Processus de mise à jour ω et formation de clusters Vision–Texte

La règle fondamentale reste :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(\mathcal{V}_i, \mathcal{T}_j) - \tau\,\omega_{i,j}(t)\right].$$

Si la synergie S entre $\mathcal{V}_i$ et $\mathcal{T}_j$ est élevée (i.e. forte similarité dans l'espace latent ou cosinus élevé), $\omega_{i,j}(t+1)$ augmente, traduisant une correspondance solide (par exemple, l'étiquette textuelle "dog" apparaît juste pour l'image représentant un chien). Lorsque l'étiquette ne convient pas (score ≈ 0), la pondération $\omega_{i,j}$ diminue peu à peu.

Après plusieurs itérations, on constate que certaines entités visuelles $\{\mathcal{V}_i\}$ et certains descripteurs textuels $\{\mathcal{T}_j\}$ présentent des $\omega_{i,j}$ fortement établis, signant l'**appariement** entre l'image et son label ou sa légende. On observe alors des **clusters** multimodaux, où un ensemble d'images (ou de patchs) se raccorde à un mot, ou une phrase plus ou moins longue. Cet effet de **renforcement** est la version auto-organisée d'une annotation, ou **captioning**, guidée par la logique du DSL.

C. Applications et bénéfices

Ce mécanisme de **synergie** Vision–Texte fonctionne comme une construction de légende, où l'image $\mathcal{V}_i$ "attire" les mots $\mathcal{T}_j$ les plus pertinents pour la décrire. Sur le plan **mathématique**, le réseau renforce $\omega_{i,j}$ pour les paires ayant une forte similarité cosinus ou une faible distance, ce qui permet au **cluster final** d'associer un sous-ensemble de mots descriptifs à chaque image. Cette approche se rapproche d'un processus de **captioning** ou d'**annotation** non supervisée, avec la possibilité d'intégrer des contraintes ou une **inhibition** (chapitre 7) pour éviter un appariement excessif.

Lorsqu'un SCN relie chaque image à des mots, une requête textuelle comme "cat" se traduit directement par la recherche des **entités textuelles** $\mathcal{T}_j$ associées à ce mot, puis des **images** qui leur sont connectées dans le réseau. Les **pondérations** ω et la **synergie** S jouent alors un rôle

d'index sémantique, permettant de retrouver quasi instantanément les images dont la légende ou le label contient "cat".

Sur le plan théorique, l'**auto-organisation** opère si la dimension des embeddings (image, texte) est suffisante et si la fonction S reflète la proximité sémantique. La descente implicite du DSL accroît $\omega_{i,j}$ pour les paires qui se co-entretiennent (mots et images vraiment voisins dans l'espace latent), conduisant à une **cohérence** à l'échelle du réseau.

8.3.3.2. Vision–Audio : correspondance entre la scène visuelle et l'environnement sonore (ex. oiseaux, etc.)

Lorsque l'on fusionne la **vision** et l'**audio** dans un **Deep Synergy Learning** (DSL) multimodal, la scène visuelle contient des informations sur la présence d'entités (oiseaux, véhicules, objets), tandis que l'audio apporte un **contexte complémentaire** (chants, bruits ambiants). Le **Synergistic Connection Network** (SCN) établit alors des **liens** entre ces entités si la **synergie** mesurée $S(\mathcal{E}_{\text{vis}}, \mathcal{E}_{\text{aud}})$ est suffisamment forte.

Cette dynamique illustre la capacité du SCN à **auto-organiser** les éléments multimodaux en fonction de leur compatibilité. Les pondérations ω évoluent de manière à renforcer les connexions entre les éléments **visuels** et **auditifs** qui se corroborent, formant ainsi des **clusters** cohérents. Cela permet une représentation intégrée de la scène et de son environnement sonore, où chaque modalité enrichit l'interprétation globale.

A. Entités visuelles vs. entités sonores

D'un côté, on dispose de **descripteurs visuels** $\mathbf{v}_i \in \mathbb{R}^d$, que l'on peut considérer soit comme un **embedding** unique de l'ensemble de l'image (ou d'une frame vidéo), soit comme un ensemble de **patchs** ou de bounding boxes détectées (plusieurs $\mathbf{v}_{i,k}$). D'un autre côté, l'**audio** est quant à lui découpé en **trames** ou en **segments**, par exemple via des MFCC ou un spectrogramme. Chaque segment audio $\mathbf{a}_j \in \mathbb{R}^m$ correspond à un court intervalle temporel (quelques millisecondes à une seconde), permettant une reconnaissance potentielle d'événements sonores (ex. chant d'oiseau, bruissement de feuilles, klaxon, etc.).

Dans un **SCN** multimodal, on introduit alors des **nœuds** $\mathcal{E}_{\text{vis},i}$ pour le canal visuel et $\mathcal{E}_{\text{aud},j}$ pour le canal audio. Les pondérations $\omega_{(vis,i),(aud,j)}$ traduisent la **compatibilité** ou la **corrélation** entre ces entités. Le **DSL** se charge de mettre à jour ces pondérations par la formule :

$$\omega_{(vis,i),(aud,j)}(t+1) = \omega_{(vis,i),(aud,j)}(t) + \eta\big[S\big(\mathcal{E}_{\text{vis},i},\ \mathcal{E}_{\text{aud},j}\big) - \tau\, \omega_{(vis,i),(aud,j)}(t)\big].$$

B. Calcul de la synergie $S\big(\mathcal{E}_{vis,i}, \mathcal{E}_{aud,j}\big)$

La **synergie** $S\big(\mathbf{v}_i, \mathbf{a}_j\big)$ repose sur plusieurs propriétés cross-modales, adaptées à l'objectif poursuivi. Une approche intuitive consiste à s'appuyer sur la **coïncidence temporelle** : si la trame audio $\mathbf{a}_j$ est capturée à l'instant t_j et que le segment visuel $\mathbf{v}_i$ correspond à l'instant t_i (ou à un intervalle $[t_i, t_i + \Delta]$), alors plus l'écart $|t_i - t_j|$ est faible, plus la synergie est élevée. Pour refléter cette dépendance, on peut définir :

$$S\big(\mathbf{v}_i, \mathbf{a}_j\big) = \rho\big(\Delta t_{i,j}\big)\ \cdot\ D\big(\mathbf{v}_i, \mathbf{a}_j\big),$$

où $\rho\big(\Delta t_{i,j}\big)$ est une fonction (par exemple gaussienne ou tophat) qui vaut 1 si $\Delta t_{i,j} < \Delta_{\max}$ et décroît à mesure que l'écart temporel croît. Le terme $D\big(\mathbf{v}_i, \mathbf{a}_j\big)$ exprime une **compatibilité de**

contenu entre les modalités visuelle et auditive. Par exemple, un modèle de détection d'objets visuels identifiant un **oiseau** dans $\mathbf{v}_i$ et un classifieur audio reconnaissant un **chant d'oiseau** dans $\mathbf{a}_j$ peuvent générer une **probabilité conjointe de correspondance**. Cette mesure permet d'évaluer dans quelle mesure les éléments des deux canaux partagent une cohérence sémantique, renforçant ainsi la **synergie** entre l'image et le son. Un schéma plus élaboré consiste à projeter $\mathbf{v}_i$ et $\mathbf{a}_j$ dans un **espace latent** commun (type CLIP multimodal) et d'utiliser un cosinus ou un RBF-kernel entre $\phi(\mathbf{v}_i)$ et $\phi(\mathbf{a}_j)$. L'intégration de la variable temporelle (décalage) peut alors se formaliser par une pénalisation $\exp(-\alpha\,|t_i - t_j|)$.

C. Correspondance scène–environnement : exemple oiseaux

Un exemple emblématique est celui d'une **scène** filmant la nature (un bosquet, un marécage) et d'une **piste audio** où l'on entend des chants d'oiseaux. Sur le plan mathématique, chaque entité visuelle $\mathbf{v}_i$ décrit un patch où l'on détecte un oiseau, tandis que chaque segment audio $\mathbf{a}_j$ correspond à un intervalle temporel où un algorithme de reconnaissance identifie un chant d'oiseau. La **synergie** $S(\mathbf{v}_i, \mathbf{a}_j)$ est alors élevée si l'on repère, d'une part, la proximité temporelle (le chant survient exactement quand l'oiseau est visible) et, d'autre part, la proximité sémantique (oiseaux de la même espèce, par exemple).

Dans le **SCN**, la pondération $\omega_{(vis,i),(aud,j)}$ se renforce lorsqu'un chant d'oiseau coïncide dans le temps et la sémantique avec la vision d'un oiseau. Au fur et à mesure que la mise à jour

$$\omega_{(vis,i),(aud,j)}(t+1) = \omega_{(vis,i),(aud,j)}(t) + \eta\left[S(\mathbf{v}_i, \mathbf{a}_j) - \tau\,\omega_{(vis,i),(aud,j)}(t)\right]$$

accumule ces signaux, un cluster inter-modal émerge en reliant "patchs d'oiseaux" et "segments audio de chants". Cela illustre la cohérence du flux multimodal où la vue et l'ouïe se soutiennent mutuellement.

D. Structure multimodale et gains

Le **SCN** ainsi formé offre plusieurs avantages. Il peut découvrir des correspondances entre des entités visuelles mal identifiées et des signaux sonores explicites. Par exemple, un chant d'oiseau peut confirmer qu'un patch visuel flou correspond bien à un oiseau. Il permet aussi de former des clusters plus riches que dans un mode unimodal en créant des macro-nœuds associant un ensemble de segments audio et un ensemble de patchs visuels. Sur le plan mathématique, la dynamique en $O(n^2)$ peut limiter la taille du dataset, à moins d'utiliser des heuristiques comme LSH ou k-NN (voir chap. 7). La philosophie du **DSL** reste cependant inchangée en auto-organisant un réseau où les liaisons trans-modales se renforcent lorsqu'une coïncidence est détectée.

8.3.3.3. Exemples concrets (images de chats accompagnées de miaulements)

Les exemples les plus parlants pour comprendre la **synergie** multimodale d'un **Deep Synergy Learning** (DSL) se manifestent souvent par de petites scènes associant **visuel** et **audio** de manière évidente, comme le cas d'un **chat** et de son **miaulement**. Lorsque l'on combine ces données au sein d'un **Synergistic Connection Network** (SCN), on obtient une représentation qui auto-organise non seulement les entités **visuelles** relatives au chat, mais aussi les entités **auditives** correspondant à des miaulements, et fait émerger leurs correspondances par la dynamique de la mise à jour ω.

A. Contexte et mise en place

Dans un **SCN** multimodal, deux familles d'entités sont introduites. La première regroupe les images de chats, notées $\{\mathcal{E}_i^{(\text{vis})}\}$, où chaque image $\mathcal{E}_i$ est représentée par un vecteur d'**embedding** $\mathbf{x}_i \in \mathbb{R}^d$. La seconde comprend les extraits audio contenant des miaulements, notés $\{\mathcal{E}_j^{(\text{aud})}\}$, chacun étant décrit par un vecteur $\mathbf{y}_j \in \mathbb{R}^{d'}$, obtenu par des **MFCC** ou un **embedding** appris.

L'objectif est d'étudier la **synergie** $S(\mathbf{x}_i, \mathbf{y}_j)$ entre un embedding d'image $\mathbf{x}_i$ et un embedding audio $\mathbf{y}_j$. Une **haute** synergie peut signifier qu'un miaulement particulier correspond à l'image d'un certain chat, ou qu'ils appartiennent au même contexte (ex. le même chat filmé et entendu simultanément).

B. Construction de la synergie multimodale

Le calcul de $S(\mathbf{x}_i, \mathbf{y}_j)$ dépend de la modalité visée. On peut se limiter à la "coïncidence" temporelle et au fait qu'une photo d'un chat noir correspond à un enregistrement audio réputé être un miaulement de chat. Sur un plan **mathématique**, un score simple se définit en repérant, par exemple, un classifieur audio qui détecte "chat" et un classifieur vision qui détecte "chat", et en pondérant leur accord :

$$S(\mathbf{x}_i, \mathbf{y}_j) = \mathbf{1}(\text{label}_{\text{vis}}(\mathbf{x}_i) = \text{"cat"}) \cdot \mathbf{1}(\text{label}_{\text{aud}}(\mathbf{y}_j) = \text{"cat meow"}).$$

Une formule plus avancée peut projeter $\mathbf{x}_i$ et $\mathbf{y}_j$ dans un **espace latent** Φ (typiquement appris pour associer images et sons), puis utiliser une distance ou une similarité :

$$S(\mathbf{x}_i, \mathbf{y}_j) = \exp(-\alpha \parallel \Phi(\mathbf{x}_i) - \Phi(\mathbf{y}_j) \parallel^2).$$

Cette fonction de synergie, insérée dans la mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(\mathbf{x}_i, \mathbf{y}_j) - \tau\, \omega_{i,j}(t)],$$

renforce les liaisons $\omega_{i,j}$ entre l'image i et le son j quand ils s'avèrent pertinents.

C. Synergie émergente : clusters "image + miaulement"

Dans un **SCN** multimodal, la dynamique $\omega_{(vis,i),(aud,j)}(t)$ finit par **souligner** les correspondances stables entre certains groupes de photos de chats et certains types de miaulements. Ainsi, un certain "chat tigré" peut se coupler à un "miaulement strident", ou un "chaton" se voir relier à un "miaulement aigu". On voit donc émerger des **clusters** mixtes contenant à la fois des entités visuelles et des entités audios.

Supposons un petit dataset de 10 images de chat (différentes races, postures) et 5 enregistrements de miaulements. Au départ, $\omega_{i,j} \approx 0$. Si un classifieur ou un module de similarité multi-modal signale que l'image $\mathbf{x}_3$ (chat siamois) coïncide souvent en contexte avec le miaulement $\mathbf{y}_2$ (miaulement "siamois"), alors $\omega_{3,2}(t)$ va croître. D'autres associations resteront faibles. Sur le plan **mathématique**, ce renforcement dérive de la valeur $S(\mathbf{x}_3, \mathbf{y}_2) \approx 0.8$ par exemple, alors que $S(\mathbf{x}_1, \mathbf{y}_2)$ serait autour de 0.1. Après plusieurs itérations, un **cluster** se solidifie autour de {images siamoises} ∪ {miaulement siamois}.

D. Observations et conclusions

Ces scénarios “images de chats + miaulements” démontrent à petite échelle la **puissance** du DSL multimodal. La **dynamique** auto-organisée :

- Crée des **liens** plus forts (pondérations ω) entre des images et des sons compatibles,
- Constitue des **clusters** multicanaux, par exemple un macro-nœud contenant un même chat vu sous différents angles et le même miaulement sur plusieurs enregistrements,
- Permet de **trouver** ou de **valider** (sans supervision explicite) quelles entités audio s’alignent le plus avec quelles entités visuelles.

Cette **approche** conserve toute la flexibilité du DSL. On peut y intégrer d’autres attributs comme les couleurs, les textures ou les types de miaulements, ainsi que différents timescales entre frames vidéo et segments audio temporels. La mise à jour des pondérations $\omega_{i,j}(t+1)$ converge alors vers un réseau cohérent associant chaque chat à son miaulement. Ce simple exemple “chat + meow” illustre la **capacité** du **SCN** à faire émerger des correspondances **cross-modales** plus complexes dans des applications réelles.

8.4. Modélisation des Entités Linguistiques

Lorsque l'on traite des données textuelles au sein d'un **SCN** (Synergistic Connection Network), la question de la **représentation** des mots, segments, phrases ou documents se révèle cruciale. La **synergie** $S(i,j)$ entre deux entités linguistiques dépend fortement de la manière dont on les encode. La section 8.4 aborde la **modélisation** des informations **linguistiques** sous forme sub-symbolique avec des embeddings vectoriels ou sous forme symbolique avec des ensembles de mots et des règles. Le **DSL** (Deep Synergy Learning) utilise ces représentations pour calculer la synergie entre entités textuelles ou entre le texte et d'autres modalités comme l'image ou l'audio.

8.4.1. Représentations Textuelles

Les entités linguistiques d'un **SCN** peuvent être des **tokens** sous forme de mots ou sous-mots, des **phrases** ou des **documents** complets. Au niveau micro, on manipule souvent des **embeddings** vectoriels tandis qu'au niveau macro, plusieurs tokens ou phrases peuvent être agrégés en un super-nœud correspondant à un thème ou un concept. Cette organisation s'appuie sur l'apprentissage multi-échelle abordé au **chapitre 6**. Deux grandes **catégories** de représentations se distinguent avec les **embeddings** sub-symboliques comme Word2Vec ou BERT et les **représentations** symboliques sous forme d'ensembles de mots-clés ou de règles logiques..

8.4.1.1. Word embeddings (Word2Vec, GloVe), contextualisés (BERT, GPT)

L'une des pièces fondamentales du **Deep Synergy Learning** (DSL) appliqué aux données **textuelles** est la manière de représenter chaque **mot** ou **segment** de phrase sous forme d'un **vecteur** dans un espace latent. Historiquement, des méthodes dites *statiques* comme **Word2Vec** ou **GloVe** ont été introduites, pour ensuite évoluer vers des méthodes *contextualisées* comme **BERT** ou **GPT**. Cette évolution mathématique reflète une sophistication croissante de la notion de "similarité sémantique" entre mots.

Embeddings statiques : Word2Vec, GloVe

Les approches de type **Word2Vec** (Skip-gram, CBOW) ou **GloVe** (Global Vectors) produisent une **projection vectorielle** de dimension d pour chaque mot du vocabulaire, notée $\mathbf{v}_w \in \mathbb{R}^d$. L'idée mathématique repose sur l'observation que des mots apparaissant fréquemment dans le même contexte (voisinage) se rapprochent dans l'espace vectoriel. Dans le cas de **Word2Vec (Skip-gram)**, l'objectif d'entraînement s'écrit

$$\max \prod_{(w,c)\in\mathcal{D}} P\,(c \mid w),$$

où $\mathcal{D}$ est un ensemble de paires (mot w, mot-contexte c) tirées d'une large collection de textes, et où la probabilité $P(c \mid w)$ se modélise via un petit réseau de neurones. La **similarité** cosinus

$$\cos(\mathbf{v}_w, \mathbf{v}_{w'}) = \frac{\mathbf{v}_w \cdot \mathbf{v}_{w'}}{\| \mathbf{v}_w \| \| \mathbf{v}_{w'} \|}$$

mesure la proximité sémantique entre deux mots w et w'. **GloVe**, quant à lui, opère sur la **matrice** de co-occurrences globales. Il s'appuie sur un tableau $X_{w,w'}$ qui comptabilise le nombre d'occurrences simultanées des mots w et w'. L'objectif est alors d'ajuster les vecteurs de telle sorte que $\mathbf{v}_w^\top \mathbf{v}_{w'}$ soit proportionnel à $\log(X_{w,w'})$. Dans les deux cas, chaque mot w hérite donc d'un unique **vecteur** $\mathbf{v}_w$. Ces embeddings sont dits *statiques* car un mot polysémique conserve la même représentation, indépendamment du contexte réel de la phrase. Les dimensions courantes vont de 50 à 300, ce qui assure un compromis entre la richesse sémantique et la facilité de manipulation.

Embeddings contextualisés : BERT, GPT

Les progrès récents en **traitement du langage** ont introduit la notion de **contexte**. un mot comme "bank" en anglais (rive ou banque) ne saurait être réduit à un unique vecteur. Les modèles **BERT**, **GPT**, **RoBERTa**, etc., s'appuient sur l'architecture **Transformer** pour produire des vecteurs qui varient selon la phrase. Mathématiquement, un Transformer recourt à des **couches** d'attention multi-têtes, formellement :

$$\text{Attention}(Q, K, V) = \text{softmax}\left(QK^\top/\sqrt{d}\right)V,$$

où Q, K, V sont des projections linéaires de l'entrée (chaque token), et d en est la dimension. En **BERT**, on entraîne l'encodeur Transformer via des objectifs tels que "masked language modeling" (prédire les tokens masqués) et "next sentence prediction". En **GPT**, on adopte une modélisation auto-régressive (chaîne de Markov cachée, où l'on prédit chaque token après les précédents). Les vecteurs résultants $\mathbf{v}_{w,t} \in \mathbb{R}^{768}$ (par exemple) reflètent la **signification** d'un mot w en position t, tenant compte de toute la phrase. Chaque occurrence du même mot dispose alors d'un embedding potentiellement différent, levant la limite des approches statiques. D'un point de vue mathématique, on obtient un nuage de vecteurs plus précis, aux dimensions parfois élevées (768, 1024...), mieux adaptés aux ambiguïtés lexicales.

Dimension et variation

Les modèles statiques (Word2Vec, GloVe) proposent des dimensions typiques de l'ordre de 100 à 300, tandis que **BERT base** en comporte souvent 768, GPT-2 large peut s'étendre jusqu'à 1024, et d'autres modèles extralarges (p. ex. GPT-3) vont encore plus loin. Manipuler des vecteurs en dimension 768 ou 1024 dans un **DSL** accroît la **richesse** potentielle de $S(\cdot,\cdot)$, mais alourdit le calcul $O(n^2 d)$. Des techniques de normalisation (section 8.3.1.3) ou d'approximate nearest neighbor (chap. 7.2.3) sont alors invoquées pour gérer la complexité.

A. Synergie et DSL

Dans le cadre du DSL, chaque token ou mot $\mathcal{E}_i$ devient un **nœud** dans le **SCN**. Son embedding $\mathbf{v}_i \in \mathbb{R}^d$ (issu de Word2Vec, BERT, etc.) permet de définir la **synergie** :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \max\{0,\ \cos(\mathbf{v}_i, \mathbf{v}_j)\} \quad \text{ou} \quad \exp(-\lambda \parallel \mathbf{v}_i - \mathbf{v}_j \parallel),$$

de sorte que deux **tokens** textuels reçoivent un score élevé s'ils sont jugés proches dans l'espace sémantique. La règle de mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$$

entraîne un **renforcement** $\omega_{i,j}$ si les embeddings se ressemblent, formant ainsi des **clusters** de mots sémantiquement proches. Dans un DSL multimodal, ce même principe peut lier des tokens

textuels à d'autres types d'entités (images, audio), d'où la dimension centrale du choix d'un bon embedding.

B. Retombées dans le SCN

Le fait de travailler avec des embeddings **contextualisés** (BERT, GPT) offre la capacité de distinguer, au sein du **SCN**, plusieurs occurrences d'un même mot dans des contextes différents. Chacune se voit attribuer un embedding $\mathbf{v}_{w,t}$. Les nœuds correspondant à "bank" dans un texte financier se rapprochent entre eux, tandis que ceux qui indiquent "bank" en tant que "rive de fleuve" forment un autre **cluster**. En outre, si l'on agrège les tokens d'une phrase ou d'un document, on peut obtenir un vecteur global document **d** (par moyenne ou attention pooling). Il devient alors possible de constituer des **macro-nœuds** pour des paragraphes, des chapitres, etc., toujours selon le même modèle de synergie et de mise à jour ω.

Le **coût** algorithmique des comparaisons cosinus $O(n^2 d)$ peut nécessiter des heuristiques (LSH, k-NN) si le nombre de tokens n et la dimension d sont grands. Mais le gain en **précision sémantique** est considérable, surtout lorsque l'on applique le DSL à des corpus textuels riches ou hétérogènes.

8.4.1.2. Forme vectorielle sub-symbolique, dimension variable (300, 768...)

La **représentation** d'entités linguistiques ou multimédias par des **vecteurs** réels est au cœur de l'approche sub-symbolique. Dans un **Deep Synergy Learning** (DSL) appliqué à un **Synergistic Connection Network** (SCN), il est fréquent de disposer d'**embeddings** (Word2Vec, GloVe, BERT, GPT, CNN, etc.) dont la dimension s'élève à 300, 512, 768, voire plus. Le présent segment met en lumière les implications mathématiques et conceptuelles de l'usage de tels vecteurs à la fois riches et complexes.

A. Vectorisation : concept et dimensionnalité

Le **principe** consiste à associer, à chaque entité $\mathcal{E}_i$ (un mot, un token BERT, un patch d'image, etc.), un vecteur $\mathbf{v}_i \in \mathbb{R}^d$. Ces vecteurs se qualifient de *sub-symboliques* puisqu'ils n'expriment pas directement une structure logique ou des symboles explicites, mais condensent l'information (ex. sémantique, contextuelle) en des coordonnées réelles. Les tailles les plus fréquentes dans la pratique se situent entre 300 et 1024, selon que l'on emploie des modèles statiques (Word2Vec/GloVe) ou des architectures transformer (BERT, GPT) plus profondes.

Sur le plan **mathématique**, ce choix de dimension d représente un compromis entre la **capacité** à capturer des nuances (un vecteur trop faible ne peut encoder suffisamment de variabilité) et la **complexité** algébrique (coût en $O(n^2 d)$ si on compare tous les vecteurs). Certains modèles de traitement de texte se limitent à 300 dimensions (Word2Vec/GloVe), quand d'autres comme BERT base utilisent 768, GPT-2 large peut monter à 1024, etc. Dans un **SCN**, rien n'interdit d'avoir des vecteurs de dimensions différentes selon les modalités, quitte à définir un **mapping** ou des formules S distinctes.

B. Calcul de la synergie $S(i,j)$ à partir de vecteurs

Dans un **DSL**, la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ doit rendre compte de la **distance** ou de la **similarité** entre $\mathbf{v}_i$ et $\mathbf{v}_j$. Les fonctions les plus usuelles incluent :

- **La similarité cosinus** $\cos(\mathbf{v}_i, \mathbf{v}_j)$, définie par

$$\cos(\mathbf{v}_i, \mathbf{v}_j) = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\| \mathbf{v}_i \| \, \| \mathbf{v}_j \|}.$$

- **La distance euclidienne** $\| \mathbf{v}_i - \mathbf{v}_j \|$, puis on inverse ou exponentie cette distance afin d'obtenir un score de proximité (ex. RBF-kernel $\exp(-\alpha \| \mathbf{v}_i - \mathbf{v}_j \|^2)$).

En appliquant le **DSL**, on associe à chaque couple (i, j) une pondération $\omega_{i,j}$, mise à jour par la règle

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(\mathbf{v}_i, \mathbf{v}_j) - \tau\, \omega_{i,j}(t)\right].$$

Deux entités à vecteurs très proches (cosinus élevé, distance faible) voient $\omega_{i,j}$ s'accentuer, trahissant leur "affinité" dans le **SCN**. Ce mécanisme engendre des **clusters** ou des "communautés" de vecteurs similaires.

C. Intégration dans le SCN

Sur le plan **architectural** (chap. 5), on réserve un "Module Synergie" pour traiter les vecteurs sub-symboliques. Ce module calcule $S(i, j)$ à partir des embeddings $\mathbf{v}_i, \mathbf{v}_j$. En cas de **multimodalité** hétérogène (texte vs. image), on peut projeter chaque vecteur dans un espace commun ou définir plusieurs synergies spécialisées.

Les **dimensions** typiques 300, 768, 1024 s'accompagnent de techniques de normalisation (on pose $\mathbf{v} \leftarrow \mathbf{v}/\| \mathbf{v} \|$) afin de mieux stabiliser les valeurs $S(i, j)$ et d'empêcher qu'une entité à très grande norme ne domine le calcul. Dans un **SCN** de grande envergure (n élevé), la comparaison $O(n^2 d)$ s'avère coûteuse, d'où l'introduction de méthodes d'approximation (LSH, k-NN) évoquées en chap. 7 pour éviter la combinatoire complète.

D. Considérations mathématiques autour de la grande dimension

La "**malédiction de la dimension**" fait qu'en très haute dimension, la distance $\| \mathbf{v}_i - \mathbf{v}_j \|$ perd parfois en contraste, comme les points peuvent tous se retrouver (en termes relatifs) assez espacés. C'est pourquoi la **similarité cosinus** ou un **kernel** (RBF) adéquat est souvent plus indiquée, accompagnée d'une étape de normalisation. De plus, le DSL n'exige pas un usage unique de cosinus ou euclidienne. D'autres formules $S(\mathbf{v}_i, \mathbf{v}_j)$ sont admissibles, pourvu qu'elles reflètent la proximité souhaitée.

8.4.1.3. Approches symboliques (ensembles de mots, règles logiques) vs. embeddings (et exemples d'overlap)

Les **représentations** dans un **Deep Synergy Learning** (DSL) peuvent s'inspirer tantôt d'un **mode symbolique** (ex. ensembles de mots, règles logiques, ontologies), tantôt d'un **mode sub-symbolique** (embeddings vectoriels issus de réseaux neuronaux). Cette coexistence illustre une dualité classique entre la **transparence** et la **rigueur** du symbolique d'une part, et la **flexibilité** ainsi que la **robustesse** des vecteurs appris d'autre part. Le **Synergistic Connection Network** (SCN) du DSL, loin d'en privilégier exclusivement un, peut exploiter **les deux** et définir la synergie $S(\mathcal{E}_i, \mathcal{E}_j)$ comme une combinaison ou un choix adaptatif.

A. Représentations symboliques : ensembles de mots, règles logiques

Un cas courant consiste à modéliser une entité $\mathcal{E}_i$ comme un **ensemble** de mots (ou de concepts) $\{w_1, \dots, w_k\} \subseteq V$. On peut alors définir la **synergie** $S_{\text{symb}}(\mathcal{E}_i, \mathcal{E}_j)$ selon une **mesure** d'**overlap**. L'exemple le plus cité demeure l'**indice de Jaccard**, qui se formule :

$$S_{\text{symb}}(\mathcal{E}_i, \mathcal{E}_j) = \frac{|\mathcal{E}_i \cap \mathcal{E}_j|}{|\mathcal{E}_i \cup \mathcal{E}_j|}.$$

Si $\mathcal{E}_i$ et $\mathcal{E}_j$ partagent de nombreux mots, leur Jaccard est élevé. Dans un **SCN**, une **pondération** $\omega_{i,j}$ plus forte apparaît lorsque les ensembles sont proches. Cette approche symbolique est **explicable** puisqu'il est possible de lister les mots communs justifiant la synergie et d'intégrer des variations plus raffinées comme une pondération TF–IDF plutôt qu'un simple ensemble.

Au-delà des simples ensembles de mots, on peut disposer de **règles** (ex. "si X alors Y"), ou d'une **ontologie** plus structurée (graphes conceptuels, taxonomies). Pour comparer deux entités symboliques $\mathcal{E}_i, \mathcal{E}_j$ — chacune étant un mini-ensemble de règles ou un segment ontologique — on définit un score S_{symb} via la fraction de clauses communes, la **compatibilité logique**, ou l'exemple d'unification (combien de substitutions rendent les deux bases cohérentes). Une forme simplifiée serait :

$$S_{\text{symb}}(\mathcal{E}_i, \mathcal{E}_j) = \text{overlap}(\mathcal{R}_i, \mathcal{R}_j) = \frac{|\mathcal{R}_i \cap \mathcal{R}_j|}{|\mathcal{R}_i \cup \mathcal{R}_j|}.$$

Cette expression d'overlap, proche du Jaccard, fonctionne tant qu'on voit les ensembles logiques $\mathcal{R}_i, \mathcal{R}_j$ comme un listing de formules. D'un point de vue **DSL**, l'inconvénient réside dans un éventuel **coût** combinatoire de comparaison si $\mathcal{R}_i, \mathcal{R}_j$ sont volumineux ou si la logique est complexe, mais on y gagne en **lisibilité** et en **fondement** théorique.

B. Embeddings sub-symboliques

Les approches sub-symboliques reposent sur la **vectorisation** de chaque entité $\mathcal{E}_i$ en $\mathbf{v}_i \in \mathbb{R}^d$. On peut tirer $\mathbf{v}_i$ d'une méthode Word2Vec, GloVe, BERT ou GPT (dimension 300, 768, 1024, etc.). Sur le plan **mathématique**, deux vecteurs $\mathbf{v}_i, \mathbf{v}_j$ se comparent via la **similarité** cosinus ou la **distance** euclidienne, puis on en dérive un score :

$$S_{\text{embed}}(\mathbf{v}_i, \mathbf{v}_j) = \max\{0,\ \mathbf{v}_i \cdot \mathbf{v}_j\} \quad \text{(si normalisés, par ex.)},$$

ou un RBF-kernel $\exp(-\| \mathbf{v}_i - \mathbf{v}_j \|^2/\sigma^2)$.

La sous-représentation sub-symbolique se montre **robuste** aux variations lexicales, gère la synonymie et la polysémie (surtout pour BERT). Elle permet des comparaisons en produit scalaire, nettement moins complexes qu'une unification de clauses logiques. L'inconvénient réside dans la **boîte noire**, car il devient moins évident de déterminer quels "mots" ou "règles" expliquent la proximité entre deux embeddings. D'un point de vue **DSL**, cela favorise toutefois la construction d'un **SCN** où l'on compare rapidement un grand nombre d'entités.

C. Concilier Symbolique et Embeddings dans le DSL

Un **SCN** multimodal ou multi-représentation peut intégrer :

1. Une composante symbolique : ensemble de mots $\mathcal{W}_i$ ou ensemble de règles $\mathcal{R}_i$. On définit une synergie $S_{\text{symb}}(i,j)$ via un *overlap* (ex. Jaccard).

2. Une composante embedding : vecteur $\mathbf{v}_i$. On définit un autre score $S_{\text{embed}}(i,j)$ via cosinus ou RBF.

3. Fusion :

$$S(i,j) = \alpha\, S_{\text{symb}}(i,j) + (1-\alpha)\, S_{\text{embed}}(i,j),$$

ou toute autre forme de combinaison. La mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big]$$

tient alors compte simultanément de la portion symbolique et de la portion sub-symbolique.

Exemple :
Si $\mathcal{E}_i$ est un document associant $\mathcal{W}_i \subseteq V$ et un embedding BERT $\mathbf{v}_i \in \mathbb{R}^{768}$, alors la synergie avec un autre document $\mathcal{E}_j$ combine $\text{overlap}(\mathcal{W}_i, \mathcal{W}_j)$ et $\cos(\mathbf{v}_i, \mathbf{v}_j)$. On peut ainsi mettre en avant la **convergence** symbolique (des mots identiques) et la **proximité** sémantique sub-symbolique (mêmes thèmes, même usage lexical).

D. Implications et choix de conception

On peut concrètement choisir l'**overlap** Jaccard pour l'ensemble de mots $\mathcal{W}_i$, par exemple :

$$S_{\text{symb}}(\mathcal{W}_i, \mathcal{W}_j) = \frac{|\mathcal{W}_i \cap \mathcal{W}_j|}{|\mathcal{W}_i \cup \mathcal{W}_j|}.$$

Dans un **DSL** textuel, cela se traduit par un haut score si deux entités partagent la plupart de leurs mots ou concepts. À l'inverse, si elles sont disjointes lexicalement, le score tombe à 0 (ou proche).

La **partie symbolique** demeure plus "interprétable" (on sait quels mots recouvrent l'entité) et s'intègre mieux à des modules logiques (on peut raisonner sur les règles). Par contre, la **partie embedding** apporte un surcroît de robustesse envers la synonymie, l'orthographe, la flexion, etc. En combinant les deux, le SCN devient plus complet, mais le calcul du score $S(i,j)$ s'alourdit.

Cette hybridation se révèle précieuse dans des systèmes cognitifs ou multi-agents, où la **logique** (symbolique) et la **similarité** (sub-symbolique) doivent coexister. Un **SCN** opérant la mise à jour $\omega \leftarrow \omega + \eta[S - \tau\,\omega]$ sur un score mixte parvient à organiser les entités selon les *deux* critères, délimitant des clusters où l'**overlap** lexical est fort *et/ou* la **distance** embedding est faible.

8.4.2. Synergie Langage-Langage

Lorsqu'il s'agit de mesurer la **synergie** entre deux entités purement **linguistiques** (ex. phrases, documents, segments de texte), le **SCN** (Synergistic Connection Network) doit disposer de **fonctions** capables d'évaluer la **similarité textuelle** ou le **partage sémantique**. Cette **section** (8.4.2) met l'accent sur la manière dont le **DSL** (Deep Synergy Learning) gère la synergie entre deux **blocs** de langage, qu'il s'agisse de phrases, de paragraphes ou de textes entiers.

8.4.2.1. Mesures de similarité textuelle (cosinus, tokens communs, etc.)

Pour **évaluer** la **synergie** entre deux entités textuelles dans un **Deep Synergy Learning** (DSL), il est essentiel de disposer d'une **mesure** de similarité adaptée. Les méthodes classiques reposent sur des représentations vectorielles ou sur le comptage de tokens (mots, n-grammes), et elles s'intègrent parfaitement dans un **Synergistic Connection Network** (SCN). Le DSL se contente d'une fonction $S(\mathcal{E}_i, \mathcal{E}_j)$ en sortie, quel que soit le détail de son calcul.

Une approche omniprésente consiste à convertir chaque entité textuelle $\mathcal{E}_i$ en un **vecteur** $\mathbf{v}_i \in \mathbb{R}^d$. Ce vecteur peut correspondre aux **embeddings** d'un mot, d'une phrase ou d'un document, issus de Word2Vec, GloVe, BERT, GPT, ou encore d'une analyse TF–IDF. Une fois ces vecteurs produits, la mesure la plus straightforward demeure la **similarité cosinus** :

$$\mathrm{SimCos}(\mathbf{v}_i, \mathbf{v}_j) = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\| \mathbf{v}_i \| \, \| \mathbf{v}_j \|}.$$

Si $\mathrm{SimCos} \approx 1$, cela indique que $\mathbf{v}_i$ et $\mathbf{v}_j$ pointent dans des directions similaires, gage d'une forte ressemblance sémantique entre les entités textuelles $\mathcal{E}_i$ et $\mathcal{E}_j$. Dans un **SCN**, cette valeur de similarité peut alimenter la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ de manière directe, par exemple :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \max\left(0,\ \mathrm{SimCos}(\mathbf{v}_i, \mathbf{v}_j)\right),$$

afin de garantir un score positif. Sur le plan **mathématique**, le calcul du cosinus s'avère $O(d)$, ce qui reste relativement accessible tant que le nombre d'entités n et la dimension d ne deviennent pas trop grands (cf. chap. 7 pour la question de la complexité $O(n^2 d)$).

Une seconde famille de mesures s'appuie sur le **comptage** de tokens (mots, n-grammes). Si $\mathcal{E}_i$ et $\mathcal{E}_j$ sont modélisés comme des **ensembles** de termes T_i et T_j, on peut définir un score d'**overlap**. Un exemple typique est l'**indice de Jaccard** :

$$\mathrm{Jaccard}(\mathcal{E}_i, \mathcal{E}_j) = \frac{|T_i \cap T_j|}{|T_i \cup T_j|}.$$

Une version différente introduit un **facteur** de normalisation basé sur la longueur, ou des **poids** TF–IDF pour accentuer l'importance des mots rares et réduire celle des mots très fréquents. Par exemple, pour un vecteur TF–IDF $\mathbf{v}_i$ (de dimension d), la **similarité cosinus** :

$$\mathrm{Sim}_{\mathrm{tfidf}}(\mathbf{v}_i, \mathbf{v}_j) = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\| \mathbf{v}_i \| \, \| \mathbf{v}_j \|}$$

renvoie un score élevé si $\mathcal{E}_i$ et $\mathcal{E}_j$ partagent des mots importants, identifiés par la TF–IDF. Ce principe est souvent utilisé pour comparer rapidement des documents textuels dans le cadre d'un SCN.

Dans la **mise en œuvre** concrète, on peut mixer **un score** Jaccard (ou overlap) pour souligner l'identité exacte de certains mots-clés et **une similarité cosinus** sur embeddings afin de gérer la proximité sémantique plus fine.

On peut alors pondérer ces deux volets :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \alpha \, \mathrm{Jaccard}(T_i, T_j) + (1 - \alpha) \, \mathrm{SimCos}(\mathbf{v}_i, \mathbf{v}_j).$$

A. Intégration au DSL : synergie $S(i,j)$

Au sein d'un **SCN**, la règle de mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(\mathcal{E}_i, \mathcal{E}_j) - \tau\, \omega_{i,j}(t)\right]$$

applique la **dynamique** auto-organisée. Un texte $\mathcal{E}_i$ voit son lien $\omega_{i,j}$ avec $\mathcal{E}_j$ renforcé si $S(\mathcal{E}_i, \mathcal{E}_j)$ est assez grand. Ainsi, des textes (phrases, documents) partageant de nombreux tokens ou jouissant d'une parenté sémantique dans l'espace embedding reçoivent une synergie plus élevée, ce qui les fait converger dans un **cluster** commun.

B. Ajustements et normalisations

Certaines variantes imposent un **seuil** pour éviter d'affecter un lien inutilement. En pratique,

$$S'(i,j) = \max(0,\ S(i,j) - \delta)$$

peut garantir que seules les paires avec $S(i,j) > \delta$ méritent d'entrer dans la mise à jour. Il est également coutumier de **normaliser** la similarité dans $[0,1]$, par exemple en réécrivant :

$$\mathrm{SimCos}(\mathbf{v}_i, \mathbf{v}_j) \leftarrow \frac{1 + \mathrm{SimCos}(\mathbf{v}_i, \mathbf{v}_j)}{2}$$

si SimCos peut devenir négatif.

8.4.2.2. "Topic synergy" : si deux documents ou phrases partagent un thème, la pondération se renforce

Dans l'analyse textuelle, il est fréquent de repérer, pour chaque document ou phrase, les **thèmes** (ou *topics*) qui y sont prédominants. Au sein d'un **Deep Synergy Learning** (DSL), cette notion de **thème** peut être exploitée pour définir une **synergie** entre deux entités textuelles $\mathcal{E}_i$ et $\mathcal{E}_j$. Si ces deux entités abordent des sujets proches, leur **pondration** $\omega_{i,j}$ tend à se renforcer, formant des **clusters** de textes thématiquement similaires. Cette logique dite "topic synergy" enrichit le **Synergistic Connection Network** (SCN) en permettant une organisation auto-émergente par **thèmes**.

A. Évaluation de la "topic synergy"

Pour estimer la proximité thématique, on modélise chaque entité textuelle $\mathcal{E}_i$ par un **vecteur de topics** $\mathbf{v}_i \in \mathbb{R}^K$. Souvent, on emploie un algorithme de topic modeling comme **LDA** (Latent Dirichlet Allocation) ou **NMF** (Nonnegative Matrix Factorization) pour extraire K grandes "thématiques" dans un corpus. Le vecteur $\mathbf{v}_i = (v_{i,1}, \ldots, v_{i,K})$ décrit la distribution de l'entité i sur les K thèmes. Chaque composante $v_{i,k}$ indique le **poids** ou la **probabilité** du thème k dans $\mathcal{E}_i$.

Pour deux entités $\mathcal{E}_i, \mathcal{E}_j$ on définit la **synergie** par la similarité entre $\mathbf{v}_i$ et $\mathbf{v}_j$. Une formule répandue demeure la **similarité cosinus** :

$$S_{\text{topic}}(i,j) = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\|\,\mathbf{v}_i\,\|\,\|\,\mathbf{v}_j\,\|}.$$

Si $\text{cosinus}(\mathbf{v}_i, \mathbf{v}_j) \approx 1$, cela signale un fort recouvrement thématique, c'est-à-dire que les **topiques** dominants de $\mathcal{E}_i$ coïncident avec ceux de $\mathcal{E}_j$. On peut encore opter pour la distance euclidienne inversée $1/1+\| \mathbf{v}_i - \mathbf{v}_j \|$ ou d'autres kernels. Sur le plan **mathématique**, ce n'est qu'une **fonction** $S_{\text{topic}}: \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}^+$; le DSL en tire un score de synergie entre entités.

La règle de mise à jour (cf. chap. 4)

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(\mathcal{E}_i, \mathcal{E}_j) - \tau\, \omega_{i,j}(t)\right]$$

utilise $S(\mathcal{E}_i, \mathcal{E}_j)$. Si on s'appuie sur la "topic synergy" comme critère, cela signifie que deux documents partageant fortement un même topique se voient **renforcer** leur liaison $\omega_{i,j}$. Progressivement, des **clusters** se dessinent autour de textes **thématiquement** convergents.

B. Pertinence pour la collaboration de documents ou de phrases

Au niveau **macro**, la distribution de topiques $\mathbf{v}_i \in \mathbb{R}^K$ encode quel pourcentage de chaque document $\mathcal{E}_i$ se rattache aux différents thèmes. Deux documents abordant le thème "Finance" (haute composante sur le topic "Finance") et "Économie internationale" se rapprochent, alors qu'un document traitant de "Biologie marine" sera éloigné. Ce **SCN** se spécialise alors en regroupant les documents par **thèmes**, révélant des clusters sémantiques.

À un niveau plus fin, on peut découper un texte en **phrases** ou **paragraphes**, puis leur attribuer des **distributions** de thèmes locales. Les clusters du SCN refléteront alors non pas l'appartenance globale d'un document, mais la portion thématique précise où la synergie s'avère forte (par ex. deux paragraphes décrivant la même technologie ou le même concept).

Dans un système plus complexe, un document fortement relié à un "Topic A" peut "coopérer" avec un autre document partageant ce Topic A, renforçant la connexion $\omega_{i,j}$. Cette *auto-organisation* favorise la **recommandation** d'éléments semblables (ou l'extraction de chapitres sur un thème commun).

C. Mathématiques avancées : combiner "topic synergy" avec d'autres synergies

On peut aisément combiner la "topic synergy" à d'autres mesures, par exemple la similarité lexicale brute ou une similarité cosinus sur des embeddings contextuels (BERT). Mathématiquement, on définit

$$S(i,j) = \alpha\, S_{\text{topic}}(i,j) + (1-\alpha)\, \text{SimCos}(\mathbf{u}_i, \mathbf{u}_j).$$

Le **DSL** intègre ce score dans la mise à jour $\omega_{i,j}$.

Si la distribution de topics $\mathbf{v}_i$ se modifie (ex. on entraîne un modèle de topic modeling plus élaboré), alors la fonction $S_{\text{topic}}(i,j)$ change, déclenchant une reconfiguration des liaisons $\omega_{i,j}$. Ceci illustre la flexibilité d'un **SCN** où la synergie s'ajuste dès que la représentation sous-jacente évolue.

8.4.2.3. Inhibition possible si on veut éviter la fusion de textes trop divergents

Dans un **Synergistic Connection Network** (SCN) appliqué à la **fusion** ou la **comparaison** de segments textuels (documents, paragraphes, phrases), on peut rechercher une **cohérence** interne au sein des clusters de textes. Il arrive cependant qu'une mesure de similarité (cosinus, overlap lexical, etc.) attribue un score modéré à deux entités très différentes, risquant alors de les

“fusionner” artificiellement et de gâcher la structure thématique. Pour éviter ce phénomène, il est fréquent d’introduire des mécanismes d’**inhibition** dans le **Deep Synergy Learning** (DSL). Ce terme “inhibition” renvoie à une **compétition** empêchant une entité de se lier simultanément à trop d’autres entités, ou de se lier à des entités trop éloignées, renforçant ainsi la cohérence globale.

A. Contexte : fusion de textes et divergence thématique

Dans un scénario où l’on compare des entités textuelles $\mathcal{E}_i$ et $\mathcal{E}_j$, la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ peut prendre la forme d’une **similarité** cosinus sur des embeddings ou d’un **overlap** de tokens (section 8.4.2.1). Si S s’avère légèrement non nulle pour deux textes très différents, la règle de mise à jour DSL

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big]$$

peut finir par **entretenir** un lien $\omega_{i,j}$ trop important, conduisant à une “fusion” excessive ou non souhaitée. On obtient alors des **clusters** hétérogènes où les textes se mélangent alors même qu’ils traitent de thématiques opposées. C’est là qu’intervient l’**inhibition**, un moyen de contenir la prolifération de liens “moyennement justifiés”.

B. Rôle de l’inhibition pour éviter une fusion inappropriée

Le principe d’inhibition indique qu’une entité $\mathcal{E}_i$ dispose d’une **capacité limitée** à établir des connexions fortes. Formulé mathématiquement, on ajoute un **terme** d’inhibition dans la mise à jour :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big] - \gamma \sum_{k \neq j} \omega_{i,k}\,(t),$$

où $\gamma > 0$ décrit la force de l’inhibition. De fait, si $\mathcal{E}_i$ cherche déjà à entretenir des liaisons $\omega_{i,k}$ non négligeables avec plusieurs textes, le nouveau lien $\omega_{i,j}$ se voit pénalisé. Les entités entrent en **compétition**, un segment textuel refuse de multiplier les liaisons moyennes, privilégiant les **quelques** plus pertinentes.

Supposons qu’un texte i ait déjà un fort lien $\omega_{i,k}$ avec un segment k très proche thématiquement. S’il tente d’en créer un second $\omega_{i,j}$ vers un texte j assez différent, la somme $\sum_k \omega_{i,k}$ devient élevée, provoquant une inhibition qui maintient $\omega_{i,j}$ à un niveau bas. Ainsi, on évite la “fusion” artificielle de textes hétérogènes.

C. Formules d’inhibition “anti-fusion”

Au-delà du schéma général, on peut concevoir des pénalisations spécifiques, par exemple :

$$\Delta_{\text{inhib}}(i,j) = -\gamma\big(\omega_{i,j}(t)\,\mathbf{1}_{S(i,j)<\delta}\big),$$

où $\mathbf{1}_{S(i,j)<\delta}$ indique que la similarité S est jugée trop faible, mais pas encore nulle. Une telle fonction punit explicitement le cas de liaisons “moyennes”, les incitant à décroître rapidement. On peut aussi incrémenter un compteur de liaisons “moyennes” et imposer un maximum par entité $\mathcal{E}_i$. Sur un plan **mathématique**, cela se traduit par un “cut-off” ou un “softmax” restreignant la somme $\sum_j \omega_{i,j}$.

8.4.3. Liens Texte–Autres Modalités

Lors du traitement de données multimodales, la **composante textuelle** occupe souvent un rôle central en fournissant des **descripteurs sémantiques** tels que mots-clés, légendes ou tags pour les autres modalités, tout en s'appuyant sur celles-ci pour enrichir le contexte. Dans le cadre du **DSL** (Deep Synergy Learning), on cherche à **relier** les entités textuelles (phrases, segments, mots) aux entités d'autres modes (images, audio, etc.) via une **synergie** $S(\mathcal{E}_{\text{texte}}, \mathcal{E}_{\text{autre}})$. Le chapitre 8.4.3 propose donc d'examiner plusieurs **dimensions** de ce lien texte–autres modalités, en insistant sur la manière dont le SCN (Synergistic Connection Network) tisse des **associations** ou clusters entre ces différents flux d'information.

8.4.3.1. Texte–Image : association sémantique (caption, tags)

Un **Synergistic Connection Network** (SCN) appliqué à un contexte **multimodal** peut relier du **texte** (mots, phrases, descriptions) à des **images** (embeddings visuels, patchs, régions d'intérêt). L'idée est d'évaluer pour chaque couple $(\mathcal{E}_{\text{txt}}, \mathcal{E}_{\text{img}})$ une **synergie** $S(\mathcal{E}_{\text{txt}}, \mathcal{E}_{\text{img}})$ qui reflète la correspondance ou la compatibilité entre le texte et l'image. Lorsqu'on s'intéresse aux associations sémantiques, on peut utiliser ce **SCN** pour des tâches comme le **captioning** (génération de légendes) et le **tagging** (assignation de mots-clés) d'images, sans recourir à un pipeline dédié.

A. Représentation des entités texte et image

Chaque unité textuelle $\mathcal{E}_{\text{txt}}$ (mot, phrase, document) peut être convertie en un vecteur $\mathbf{v}_{\text{txt}} \in \mathbb{R}^{d_t}$, via, par exemple, Word2Vec, GloVe, BERT, GPT, ou TF–IDF. Cette étape confère à l'entité textuelle une position dans un espace sémantique. On comparera ensuite deux vecteurs $\mathbf{v}_{\text{txt}}$ par un produit scalaire ou une distance.

Pour chaque image $\mathcal{E}_{\text{img}}$, on recourt à un **réseau convolutionnel** (ex. ResNet, VGG) ou un **transformer** (ViT) pour extraire un embedding $\mathbf{v}_{\text{img}} \in \mathbb{R}^{d_i}$. L'analyse peut se faire globalement sur l'image entière ou localement sur des bounding boxes ou patchs.

Dans un **DSL** multimodal, la **synergie** $S(\mathcal{E}_{\text{txt}}, \mathcal{E}_{\text{img}})$ découle d'une **fonction** $\kappa(\mathbf{v}_{\text{txt}}, \mathbf{v}_{\text{img}})$. Selon la mise en œuvre, on peut

$$\kappa(\mathbf{v}_{\text{txt}}, \mathbf{v}_{\text{img}}) = \mathbf{v}_{\text{txt}}^{\top}\, \mathbf{W}\, \mathbf{v}_{\text{img}},$$

ou plus simplement projeter $\mathbf{v}_{\text{txt}}, \mathbf{v}_{\text{img}}$ dans un espace commun (style CLIP) et faire un **cosinus**. Le **SCN** se contente d'une **valeur** positive décrivant la proximité sémantique entre le texte et l'image.

B. Association sémantique (captions, tags)

Dans un **SCN**, si un segment textuel $\mathcal{E}_{\text{txt}}$ affiche une forte synergie $S(\mathcal{E}_{\text{txt}}, \mathcal{E}_{\text{img}})$ avec une image $\mathcal{E}_{\text{img}}$, la règle de mise à jour DSL

$$\omega_{(\text{img,txt})}(t+1) = \omega_{(\text{img,txt})}(t) + \eta\left[S(\mathbf{v}_{\text{img}}, \mathbf{v}_{\text{txt}}) - \tau\, \omega_{(\text{img,txt})}(t)\right]$$

renforce ce lien ω. Au terme de l'auto-organisation, on verra **clusters** contenant l'image et un ensemble de tokens/phrases très liés, interprétable comme une légende. Par exemple, une image de chat se retrouvera fortement associée aux mots "cat", "feline", "grey fur", etc.

Sur un plan plus discret, on peut assigner des *tags* (mots-clés) à une image si la synergie cross-modale est au-dessus d'un certain seuil. En pratique, cela veut dire qu'un vecteur pour "dog" se rapprochera fortement de l'embedding visuel détectant un canidé. L'**auto-organisation** conduit la pondération $\omega_{(\text{img},\text{“}dog\text{”})}$ à s'élever, signifiant que l'image reçoit le tag "dog". D'autres tags moins compatibles déclinent.

C. Avantages mathématiques : couplage embeddings–synergie

D'un point de vue **algébrique**, on peut chercher à projeter $\mathbf{v}_{\text{txt}}$ et $\mathbf{v}_{\text{img}}$ vers un espace commun $\mathbb{R}^d$ grâce à des matrices $\mathbf{W}_{\text{txt}}$ et $\mathbf{W}_{\text{img}}$. Ensuite, la **similarité** cross-modal $\kappa\left(\mathbf{v}_{\text{txt}}, \mathbf{v}_{\text{img}}\right)$ devient un simple produit scalaire. L'approche **CLIP** (OpenAI) opère un apprentissage supervisé sur un gigantesque corpus d'images-légendes pour aligner $\mathbf{v}_{\text{img}}$ et $\mathbf{v}_{\text{txt}}$.

Le grand apport du DSL est qu'il **renforce** de façon auto-organisée un lien $\omega_{i,j}$ si la synergie demeure élevée. Ainsi, aucune instruction explicite n'est requise pour dire "ce texte décrit cette image" ; la dynamique ω le **découvre** si, en pratique, κ reste haut pour un couple image–texte. On aboutit alors à un **cluster** d'images et un cluster de mots/phrases partiellement confondus, traduisant la correspondance sémantique.

D. Cas d'utilisation dans un SCN global

Une fois un **SCN** construit (images + segments textuels), on peut, pour un nouveau mot, repérer quelles images possèdent déjà une liaison élevée ω avec ce mot (ou un mot synonyme). Le **DSL** agit comme un **index** sémantique en exploitant les arcs ω qui relient des textes proches du nouveau mot à telle ou telle image.

En sens inverse, on peut découvrir quels mots ou légendes s'agglomèrent autour d'un groupe d'images semblables, ce qui vaut étiquetage ou classification. La structure **auto-organisée** permet également de "fusionner" l'information visuelle et textuelle pour générer ou compléter des légendes de manière plus souple.

8.4.3.2. Texte–Audio : ex. transcript, reconnaissance vocale

Lorsque l'on aborde un **contexte** multimodal (Chap. 8), la **fusion** entre flux **audio** et **texte** constitue une application de choix. Dans de nombreux cas, il s'agit de **reconnaissance vocale** ou de **transcription** où le **signal** acoustique, qu'il soit continu (discours, conversation) ou segmenté (clips audio), est mis en correspondance avec des entités textuelles plus ou moins élaborées. Le **Deep Synergy Learning** (DSL) fournit un **Synergistic Connection Network** (SCN) où l'on peut simultanément gérer des entités **audio** et **texte**, et définir une **synergie** qui reflète la **cohérence** entre ces deux modalités.

A. Configuration Générale : Audio + Texte

Le flux **audio** se décompose en segments $\{\mathcal{A}_k\}$, chacun pouvant être caractérisé par des **features** acoustiques $\mathbf{v}_{a,k} \in \mathbb{R}^{d_a}$. Ces descripteurs incluent souvent des coefficients MFCC, un spectrogramme réduit, ou des indices de prosodie (intensité, pitch). Parallèlement, on dispose d'un ensemble d'entités **texte**, par exemple $\{\mathcal{T}_i\}$, qui peuvent être des **mots**, des **tokens** (BERT/GPT) ou des **segments** plus larges (phrases, paragraphes). Chaque entité textuelle se voit associée à un **vecteur** $\mathbf{v}_{t,i} \in \mathbb{R}^{d_t}$, obtenu via un embedding (Word2Vec, GloVe, transformeur) ou via un simple TF–IDF.

Le **SCN** (Synergistic Connection Network) regroupe dans un **même** graphe les entités audio et texte, et définit une pondération $\omega_{(a,k),(t,i)}$ entre le segment audio $\mathcal{A}_k$ et le segment texte $\mathcal{T}_i$. La **synergie** $S(\mathcal{A}_k, \mathcal{T}_i)$ doit alors rendre compte d'une **compatibilité** audio–texte. Si l'on vise la **reconnaissance vocale**, on souhaite repérer lequel des mots $\{\mathcal{T}_i\}$ s'aligne le mieux avec le segment acoustique $\mathcal{A}_k$. Dans un cadre plus général, on peut chercher à mesurer la "correspondance sémantique" entre un contenu verbal (texte) et un indice acoustique (ton, style).

B. Cas d'Usage : Transcript et Reconnaissance Vocale

Le cas de la **reconnaissance vocale** ou de la **transcription** se prête particulièrement bien à la logique DSL. Une mise en correspondance classique se présenterait ainsi :

$$\omega_{(a,k),(t,i}(t+1) = \omega_{(a,k),(t,i)}(t) + \eta\big[S(\mathcal{A}_k, \mathcal{T}_i) - \tau\, \omega_{(a,k),(t,i)}(t)\big].$$

Plus la **similarité** ou la **cohérence** $S(\mathcal{A}_k, \mathcal{T}_i)$ est élevée, plus le lien ω se renforce au fil des itérations.

Dans un système disposant à la fois d'un **flux audio** et d'une **transcription** (manuelle ou partielle), on peut prendre chaque segment $\mathcal{A}_k$ en regard de chaque portion de texte $\mathcal{T}_i$. La **synergie** $S(\mathcal{A}_k, \mathcal{T}_i)$ repose par exemple sur un **score** phonémique (ressemblance entre le spectre détecté et la chaîne phonétique du mot) ou sur un **embedding** commun (par exemple, un vecteur acoustique vs. un vecteur BERT), voire les deux :

$$S(\mathcal{A}_k, \mathcal{T}_i) = \alpha\, \mathrm{sim}_{\mathrm{phon}}\big(\mathbf{v}_{a,k},\, \mathbf{v}_{t,i}\big) + \beta\, \mathrm{sim}_{\mathrm{sem}}\big(\mathbf{v}_{t,i},\, \text{contexte lexical}\big),$$

où $\mathrm{sim}_{\mathrm{phon}}$ évalue la proximité acoustique (indicateurs de phonèmes) et $\mathrm{sim}_{\mathrm{sem}}$ jauge la cohérence lexicale. Le **DSL** renforce alors $\omega_{(a,k),(t,i)}$ si ce couple audio–texte demeure pertinent, ce qui affine l'alignement final entre la séquence vocale et les mots transcrits.

Dans un pipeline habituel de speech-to-text, le système produit déjà un alignement hypothétique. Un **SCN** multimodal peut raffiner ce couplage par la **dynamique** auto-organisée. Chaque segment audio peut tester plusieurs hypothèses textuelles. Si un couple $(\mathcal{A}_k, \mathcal{T}_i)$ obtient un score $S(\mathcal{A}_k, \mathcal{T}_i)$ supérieur, il sera **renforcé**, tandis que les liaisons concurrentes se verront "repoussées". On obtient in fine une structure ω où chaque segment audio lie fortement le texte le plus vraisemblable.

C. Bénéfices d'une Approche DSL pour Texte–Audio

Le **DSL** fournit un cadre unique pour gérer la **cohérence** linguistique et acoustique. Contrairement à un pipeline strictement séquentiel, un SCN peut rassembler en un graphe global divers segments $\{\mathcal{A}_k\}$ et $\{\mathcal{T}_i\}$, qu'il peut **associer** ou **désassocier** selon la synergie calculée. Sur le plan **mathématique**, la mise à jour ω se base uniquement sur $S(\mathcal{A}_k, \mathcal{T}_i)$ et un terme d'amortissement $\tau\, \omega_{(a,k),(t,i)}$.

Plutôt que d'imposer un modèle de Markov caché ou un alignement rigide, le DSL favorise une **résonance** locale de sorte que si un bloc audio correspond au mot $\mathcal{T}_i$, on renforce $\omega_{(a,k),(t,i)}$. Cela facilite la **correction** de transcriptions si une hypothèse initiale se révèle incohérente (score en baisse) ou la **découverte** de nouveaux mots si l'on introduit de nouveaux segments textuels.

On peut ajouter dynamiquement de nouveaux extraits audios ou de nouveaux tokens textuels (vocabulaire mis à jour). Le SCN recalcule alors la synergie entre chaque couple $(\mathcal{A}_k, \mathcal{T}_i)$ et

réitère les mises à jour ω. Un système d'analyse **temps-réel** peut boucler régulièrement, incorporant de plus en plus de segments audio ou lexical, permettant à l'**auto-organisation** de suivre l'évolution de la conversation.

8.4.3.3. Cas d'usage : un SCN où mots, phrases, images s'agglomèrent autour de concepts communs

Il est souvent instructif de disposer d'un **Synergistic Connection Network** (SCN) qui manipule à la fois des **entités textuelles** (mots, phrases) et des **entités visuelles** (images ou extraits d'images). On souhaite observer la formation de **clusters** ou de **macro-nœuds** où, de façon auto-organisée, on regroupe mots et images décrivant le même **concept** ou la même **idée**. Dans le **Deep Synergy Learning** (DSL), ce phénomène émerge du fait que la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ valorise les associations sémantiquement ou perceptuellement proches, puis la mise à jour ω concrétise ces proximités en liaisons fortes.

A. Structure Générale du SCN

Un **SCN** peut contenir plusieurs catégories d'entités. D'un côté, il y a les **mots** (ex. "chat", "arbre", "voler") et des **phrases** plus étendues (des segments, des énoncés entiers). Chacune de ces entités textuelles $\mathcal{E}_i^{(\text{txt})}$ se voit associer un **vecteur** $\mathbf{v}_i \in \mathbb{R}^{d_t}$, obtenu par Word2Vec, GloVe, BERT ou tout autre embedding neuronal. De l'autre côté, il y a des **images** (ou des parties d'images) $\mathcal{E}_j^{(\text{img})}$, encodées par un CNN ou un ViT, ce qui donne $\mathbf{u}_j \in \mathbb{R}^{d_v}$. Dans un **DSL** multimodal, le **SCN** contient ainsi un ensemble hétérogène de nœuds (texte, image), reliant chaque paire (i, j) par une **pondération** $\omega_{i,j}$.

Pour deux entités de la **même** modalité (texte–texte ou image–image), on peut définir une similarité standard (cosinus, distance inverse). Pour un **couple** texte–image, on recourt à une **fonction** $f(\mathbf{v}_i, \mathbf{u}_j)$ s'apparentant à un alignement cross-modal (ex. un modèle "image to text" appris type CLIP, ou un simple produit scalaire si l'on dispose d'espaces déjà projetés). On obtient :

$$S\left(\mathcal{E}_i^{(\text{txt})}, \mathcal{E}_j^{(\text{img})}\right) = f(\mathbf{v}_i,\ \mathbf{u}_j),$$

puis la **règle** DSL

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big].$$

B. Émergence de Concepts Communs

Supposons qu'un **mot** "chat", une **phrase** "Le chat dort sur un coussin" et une **image** représentant un chat partagent des embeddings proches. Le calcul de S("chat",image) sera élevé, de même que S("Le chat dort…",image). La **dynamique** DSL renforce alors les liaisons $\omega_{(\text{mot="chat"}),(\text{image})}$ et $\omega_{(\text{phrase="Le chat…" }),(\text{image})}$. Au fil des itérations, on voit émerger un **micro-cluster** text–image centrant sur le **concept** "chat". Le **SCN** ne requiert pas qu'on pré-spécifie ce concept car c'est la **haute synergie** qui unifie ces entités dans un nœud macro, aboutissant à un **cluster** text–image.

Dans un **SCN** de grande envergure, de nombreux mots, phrases et images s'organisent en différents **clusters**. Un premier groupe se forme autour de "chat / feline", un second autour de "chien / dog", tandis qu'un troisième regroupe "Paris / Tour Eiffel" associé à des images de la

capitale. Cette auto-organisation survient sans supervision puisque le **DSL** applique la mise à jour $\omega_{i,j} \leftarrow \omega_{i,j} + \eta\big[S(i,j) - \tau\,\omega_{i,j}\big]$, la synergie se développant naturellement selon la proximité sémantique ou perceptuelle.

C. Modélisation Mathématique de la Synergie Multimodale

Pour comparer un mot $\mathbf{v}_i \in \mathbb{R}^{d_t}$ et une image $\mathbf{u}_j \in \mathbb{R}^{d_v}$, on définit :

$$S_{\text{cross}}\big(\mathbf{v}_i, \mathbf{u}_j\big) = \cos\big(\mathbf{W}_{\text{txt}}\,\mathbf{v}_i,\ \mathbf{W}_{\text{img}}\,\mathbf{u}_j\big),$$

où $\mathbf{W}_{\text{txt}}$ et $\mathbf{W}_{\text{img}}$ sont des **matrices** (ou des réseaux) projetant dans un espace commun, puis on applique la **similarité cosinus**. Si le score est élevé, on conclut que l'image et le mot décrivent un même concept. On peut se fier à des approches pré-entraînées, telles que CLIP, pour disposer déjà de vecteurs $\mathbf{v}_i, \mathbf{u}_j$ alignés, et recourir au **produit scalaire** direct.

Lorsqu'on compare texte–texte, on choisit S_{txt} ; quand on compare image–image, on dispose de S_{img}. Le **DSL** peut agréger :

$$S(i,j) = \begin{cases} S_{\text{txt}}\big(\mathbf{v}_i, \mathbf{v}_j\big), & \text{(texte–texte)}, \\ S_{\text{img}}\big(\mathbf{u}_i, \mathbf{u}_j\big), & \text{(image–image)}, \\ S_{\text{cross}}\big(\mathbf{v}_i, \mathbf{u}_j\big), & \text{(texte–image)}. \end{cases}$$

Le **SCN** met alors à jour toutes les $\omega_{i,j}$ en conséquence, générant une structure globale où certains arcs correspondent à similitudes text–text et d'autres à cross-modal text–image.

D. Processus d'Auto-Organisation

À chaque **itération**, la formule

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S\big(\mathcal{E}_i, \mathcal{E}_j\big) - \tau\,\omega_{i,j}(t)\big]$$

applique l'**auto-organisation**. Lorsque $S\big(\mathcal{E}_i, \mathcal{E}_j\big)$ est élevé de façon récurrente, $\omega_{i,j}$ grandit, traduisant un lien stable au sein du **SCN**. D'un point de vue **analytique**, on peut prouver que si η et τ restent constants et S fixe, on tend vers un état stationnaire où

$$\omega_{i,j}^{*} = \frac{S(i,j)}{\tau},$$

pour chaque couple (i,j). Dans les cas plus complexes, l'existence de termes d'inhibition ou la variabilité de S suscite des bifurcations, mais la logique de regroupement demeure.

Si plusieurs mots, phrases et images sont tous fortement interconnectés (liens ω élevés), ils forment un **macro-nœud** (chap. 6). Cette entité plus large représente un **concept** commun ("sports de balle", "animaux de ferme", "vacances estivales"), que le DSL identifie sans supervision explicite. C'est la **philosophie** d'**auto-organisation** où la **dynamique** localisée sur ω produit spontanément une hiérarchie de regroupements text–image.

E. Gains Opérationnels

Un **SCN** réunissant les **mots**, **phrases** et **images** autour de **concepts** facilite des applications telles que la **recherche** multimodale (trouver une image via un mot), le **tag** automatique d'images par le vocabulaire textuel, ou l'**annotation** textuelle (proposer des légendes). La **mise**

à jour ω confère à ce système une capacité d'évolution où de nouveaux mots ou images peuvent s’intégrer aux clusters existants dès lors qu’ils présentent une synergie suffisante.

8.5. Modélisation des Entités Sonores

Dans un contexte **multimodal**, la dimension **audio** occupe une place cruciale. Elle renseigne sur les **caractéristiques** d'un signal acoustique (voix, musique, bruitages, etc.) et peut s'intégrer au **SCN** (Synergistic Connection Network) en tant qu'entités d'information. Pour que le **DSL** (Deep Synergy Learning) puisse exploiter la **synergie** entre sources audio et autres modalités (vision, texte, etc.), il est indispensable de **représenter** chaque segment audio de façon adéquate.

8.5.1. Représentations Audio

Le choix de la **représentation** (features ou embeddings) appliquée à une **trame** (ou un bloc) de signal sonore conditionne grandement la **qualité** du calcul de synergie $S(\mathcal{E}_i, \mathcal{E}_j)$. On peut distinguer les **features classiques** (MFCC, spectrogrammes) des **embeddings profonds** plus récents (Audio2Vec, etc.).

8.5.1.1. Features classiques (MFCC, spectrogrammes) vs. embeddings profonds (Audio2Vec)

Il existe plusieurs **méthodes** pour représenter un **signal audio** sous forme de **vecteurs** ou de **descripteurs** exploitables par le **Deep Synergy Learning (DSL)** dans le cadre d'un **Synergistic Connection Network** (SCN). Certains choix relèvent d'**approches classiques**, telles que les **MFCC** ou les **spectrogrammes**, tandis que d'autres recourent à des **embeddings profonds** (ex. Audio2Vec). Chaque famille de techniques possède ses avantages et contraintes, influençant la **synergie** calculée entre segments audio et la structure finale du réseau.

A. Caractéristiques Classiques

Les **MFCC** constituent une représentation historique et largement utilisée en reconnaissance de parole et analyse audio. Le signal $\mathbf{x}(t)$ est découpé en trames temporelles de durée courte (souvent 20 à 40 ms). Pour chaque trame, on calcule un **spectre** à l'aide de la transformée de Fourier discrète :

$$X(\omega) = \sum_{\ell=0}^{L-1} x\,(\ell)\, e^{-i\,\omega\,\ell},$$

où L représente le nombre d'échantillons dans la trame et ω la variable fréquentielle. On projette ensuite ce spectre sur une **échelle Mel** (partition non linéaire de la bande de fréquences), puis on applique un log de la puissance et, enfin, une **Discrete Cosine Transform** (DCT) pour obtenir les **coefficients cepstraux c**. En pratique, on retient souvent les 13 premiers coefficients, éventuellement complétés de leurs dérivées ($\Delta\mathbf{c}, \Delta^2\mathbf{c}$). Sur le plan mathématique, la représentation MFCC $\mathbf{c} \in \mathbb{R}^{13}$ condense la forme globale du spectre de parole en tenant compte de l'échelle psychoacoustique Mel. Le principal avantage se situe dans la **compacité** (très faible dimension) et la robustesse en reconnaissance de parole. L'inconvénient est une capacité plus limitée à représenter des nuances complexes (timbres musicaux, émotions, etc.).

Les **spectrogrammes** constituent une approche plus détaillée. Le signal $\mathbf{x}(t)$ est segmenté en fenêtres glissantes, et pour chacune, on calcule la **Short-Time Fourier Transform** (STFT). On obtient ainsi une matrice $\mathbf{S}(t, f)$ représentant l'amplitude (ou la puissance) en fonction du temps et de la fréquence :

$$\mathbf{S}(t,f) \approx \left| \sum_{\ell} x\,(\ell)\, w(\ell - t)\, e^{-2\,i\,\pi\,f\,\ell} \right|,$$

où $w(\ell - t)$ désigne une fonction fenêtre localisant la portion temporelle. Sur le plan **mathématique**, cela donne un tableau 2D plus volumineux que les MFCC, mais contenant davantage d'information sur les harmoniques et la structure fréquentielle. Les similarités entre spectrogrammes de segments audio peuvent se calculer par cosinus, distance euclidienne ou kernels RBF. Le principal intérêt réside dans la **richesse** de la représentation ; l'inconvénient tient à la **dimension** plus élevée et au coût de calcul dans un **SCN** volumineux.

B. Embeddings Profonds (Audio2Vec, etc.)

Certaines méthodes s'inspirent du succès de **Word2Vec** dans le monde textuel pour proposer des **embeddings** spécifiques à l'audio. L'objectif est d'apprendre un **réseau** Φ_θ qui, pour tout segment audio $\mathbf{x}$, produit un vecteur $\mathbf{z} = \Phi_\theta(\mathbf{x}) \in \mathbb{R}^d$. L'apprentissage peut se faire par supervision (classer des locuteurs, reconnaître des mots-clés) ou par approches auto-supervisées (contraste entre segments positifs/négatifs). Une fois entraîné, ce **réseau** Φ fournit un **embedding** audio plus abstrait, capturant des **patterns** variés (intonation, spectre, timing). Mathématiquement, la représentation finale :

$$\mathbf{z} = \Phi_\theta(\mathbf{x}) \in \mathbb{R}^d$$

s'obtient en passant l'onde audio ou son spectrogramme dans un CNN, un RNN ou un transformeur audio. Les **embeddings profonds** (type Audio2Vec) améliorent la généralisation et la robustesse, car ils s'appuient sur des couches profondes. Cependant, ils exigent souvent un **dataset** conséquent et un **entraînement** onéreux (réseau large, multiples époques).

C. Forme mathématique d'un embedding audio

Quelle que soit la méthode, on aboutit finalement à un **vecteur** $\mathbf{z}_i \in \mathbb{R}^d$ pour l'entité audio $\mathcal{E}_i$. Dans un cadre plus formel, on pose

$$\Phi_\theta(\mathbf{x}) = \mathbf{z} \in \mathbb{R}^d,$$

avec θ ajusté par minimisation d'une perte

$$\mathcal{L}(\theta) = \sum_{\mathbf{x},\mathbf{y}} \ell(\Phi_\theta(\mathbf{x}),\, \mathbf{y}).$$

L'apprentissage détermine θ pour classifier ou regrouper des signaux. Une fois θ fixés, le vecteur $\mathbf{z}$ devient la **représentation** sub-symbolique du segment audio.

D. Implication pour le SCN

Dans un **SCN**, le **Deep Synergy Learning** évalue la **distance** (ou la **similarité**) entre deux vecteurs $\mathbf{z}_i$ et $\mathbf{z}_j$. On peut définir

$$S(i,j) = \exp\left(-\alpha \parallel \mathbf{z}_i - \mathbf{z}_j \parallel^2\right) \quad \text{ou} \quad \frac{\mathbf{z}_i \cdot \mathbf{z}_j}{\parallel \mathbf{z}_i \parallel \parallel \mathbf{z}_j \parallel},$$

selon l'option RBF (Gaussienne) ou la similarité cosinus. Cette **synergie** gouverne la mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\, \omega_{i,j}(t)\right].$$

Les segments audio s'**agglomèrent** en **clusters** s'ils présentent des vecteurs $\mathbf{z}_i$ voisins. Cette **auto-organisation** acoustique peut également s'étendre à d'autres modalités (texte, image), ce qui fonde la multi-modalité (voir Chap. 8.5.2, 8.5.3).

8.5.1.2. Normalisation : amplitude, échelle log, etc.

Lorsque l'on souhaite traiter plusieurs **modalités** (audio, vision, texte, capteurs, etc.) ou simplement différentes **sources** de données (descripteurs MFCC vs. spectrogrammes, embeddings, signaux bruts), il s'avère souvent indispensable de recourir à des **techniques de normalisation**. L'objectif est de **mettre à l'échelle** les variables ou les vecteurs de représentation, afin d'éviter qu'une modalité ne "domine" artificiellement les calculs de **synergie** $S(i,j)$ dans le **Deep Synergy Learning** (DSL).

A. Normalisation d'Amplitude

Les données collectées, qu'elles proviennent de l'audio (amplitude ou puissance spectrale), du texte (fréquences de mots, valeurs TF–IDF), ou d'autres modalités, peuvent se situer sur des échelles très diverses. On peut alors se retrouver avec des **vecteurs** $\mathbf{x}_i$ dont la norme $\parallel \mathbf{x}_i \parallel$ est très grande (par exemple, une intensité audio ou un histogramme massif), tandis qu'un autre vecteur $\mathbf{x}_j$ possède une norme plus réduite. Pour éviter que l'on compare des grandeurs incomparables, on pratique une **division** ou une **mise à l'échelle** afin de les **ramener** à des amplitudes similaires.

Une forme simple est la **mise à l'échelle min–max**, définie composante par composante :

$$\mathbf{x'}_i = \frac{\mathbf{x}_i - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})}$$

ce qui ramène les valeurs dans l'intervalle $[0,1]$. Pour des représentations vectorielles plus classiques, on recourt souvent à la **normalisation** ℓ_2 :

$$\mathbf{x'}_i = \frac{\mathbf{x}_i}{\parallel \mathbf{x}_i \parallel}.$$

Cette dernière est très prisée lorsque la **synergie** se base sur la similarité cosinus, puisqu'on ne regarde alors plus que l'**angle** entre deux vecteurs, et pas leur taille.

La normalisation d'amplitude garantit que la **magnitude** de chaque vecteur reste bornée. Dans un **SCN** multimodal, cela évite qu'une modalité, dont les valeurs numériques seraient élevées, impose systématiquement une **forte** synergie malgré l'absence de vrai lien sémantique ou perceptuel. En définissant, par exemple, $\mathbf{x'}_i = \mathbf{x}_i / \parallel \mathbf{x}_i \parallel$, on homogénéise les directions, ce qui rend les similarités plus interprétables.

B. Échelle Logarithmique (Rescaling log)

Il arrive que certaines modalités (certains signaux) varient sur des **ordres de grandeur** très différents, comme 1, 10, 1000. Dans ce contexte, un simple min–max ou division par la norme ne suffit pas toujours. Une **transformation logarithmique** atténue les valeurs extrêmes, ce qui est souvent pertinent pour des grandeurs exponentielles (ex. volume sonore, intensité lumineuse, fréquence de mots en TF–IDF). En pratique, on définit :

$$x'_i = \log(1 + \alpha\, x_i),$$

avec $\alpha > 0$ choisie pour adapter l'échelle.

Si l'on note $\mathbf{x}_i$ un vecteur de valeurs non négatives, la version "log-scale" s'écrit :

$$\mathbf{x}'_i = \log(1 + \alpha\, \mathbf{x}_i).$$

Cela **réduit** considérablement l'influence d'un vecteur très large sur la similarité si son écart n'est qu'exponentiel.

En ramenant les valeurs par le **log**, on privilégie les **rapports** plutôt que les **différences** absolues. Dans le **Deep Synergy Learning**, cela revient à ce que la **synergie** S se comporte de façon plus stable quand il y a des valeurs très distinctes. C'est un choix adapté aux distributions sur de multiples échelles (comme l'audio intensité, ou les histogrammes de fréquences pour certains tokens rares/fréquents).

C. Normalisation par Statistiques Globales

On peut aussi opter pour une **standardisation** classique par la moyenne μ et l'écart-type σ. Cela consiste à recentrer et réduire les valeurs :

$$\mathbf{x}'_i = \frac{\mathbf{x}_i - \mu}{\sigma}.$$

Lorsque l'on compare différents flux (audio, vision, texte), on peut effectuer cette opération flux par flux, pour s'assurer qu'ils possèdent tous une moyenne proche de 0 et un écart-type proche de 1.

La standardisation basée sur la médiane et la MAD (Median Absolute Deviation) s'emploie pour minimiser l'effet des outliers. On substitue à la place de μ, σ la **médiane** et la médiane des écarts absolus, puis on ramène $\mathbf{x}_i$ à des valeurs autour de 0 avec une échelle unitaire. Cette méthode confère une plus grande **robustesse** lorsque quelques valeurs extrêmes pourraient biaiser les moyennes.

D. Approche Hybride ou Multi-Pass

Il est possible de combiner plusieurs **stratégies** dans un même SCN multimodal. Par exemple, on peut :

- Prendre le **log** des mesures sur une dimension (ex. intensité audio) pour compacter l'amplitude,
- Appliquer un **min–max** ou une **normalisation** ℓ_2 aux données visuelles,
- Standardiser (moyenne 0, variance 1) les valeurs textuelles issues d'un comptage ou d'un vecteur TF–IDF.

Ensuite, on **concatène** ou on confronte ces vecteurs dans une synergie $S(i,j)$. Sur le plan **mathématique**, la normalisation assure que chaque flux ou dimension contribue de façon plus équilibrée.

E. Implication Mathématique dans le Calcul de Synergie

Lorsque plusieurs modalités coexistent, la normalisation unifie leur amplitude et permet un calcul de $S(i,j)$ plus fidèle, sans qu'un flux à grande échelle ne "domine". Dans le **DSL**, la mise à jour $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$ demeure plus cohérente si les valeurs de $S(i,j)$ ne sont pas saturées par des différences d'échelle.

Des signaux numériquement plus stables évitent à la dynamique de l'**auto-organisation** de s'emballer ou de converger trop lentement. Sur le plan **mathématique**, cela réduit les gradients ou incréments exagérément grands, ce qui peut accélérer la **convergence**.

8.5.1.3. Traitement de séquences ou de "frames" audio

Un **système** multimodal intégrant un **flux audio** se confronte à la question de la **nature** du signal acoustique. L'information auditive, contrairement à une image statique ou un bloc de texte, se déploie dans le temps. Il est alors fréquent de découper ce signal en **trames** (frames) successives, chacune relativement courte (ex. 10 ms, 20 ms), afin de représenter plus finement la **dynamique** temporelle. Le **Deep Synergy Learning** (DSL) appliqué au **Synergistic Connection Network** (SCN) prend alors en charge un ensemble d'entités $\mathcal{E}_m$ indexées par le temps (m) et veille à **auto-organiser** ces trames ou séquences selon leur cohérence spectrale, phonétique ou prosodique.

A. Segmentation Audio en Frames : Principes et Modèle Mathématique

Le **signal** audio continu $x(t)$ est classiquement découpé en trames temporelles de longueur L, possiblement chevauchantes (overlap). Chaque trame $\mathbf{a}^{(m)} \in \mathbb{R}^d$ correspond, par exemple, à un vecteur MFCC, un spectre ou un embedding audio (chap. 8.5.1.1). Les index $m = 1,2,\ldots$ parcourent ainsi les portions du signal à des intervalles réguliers Δ. D'un point de vue **mathématique**, on peut noter :

$$\mathbf{a}^{(m)} = \text{Transform}\left(\{x(t)\}_{t=m\cdot\Delta}^{(m+1)\cdot\Delta}\right),$$

où Transform décrit l'extraction de features (DFT, MFCC, etc.). Chaque $\mathbf{a}^{(m)}$ devient alors une entité $\mathcal{E}_m$ dans le **SCN**.

Une fois les frames $\mathbf{a}^{(m)}$ calculées, on les intègre dans le **Synergistic Connection Network**. Chaque frame $\mathcal{E}_m$ possédant un vecteur $\mathbf{a}^{(m)} \in \mathbb{R}^d$, la **synergie** $S(\mathcal{E}_m, \mathcal{E}_{m'})$ peut être définie, par exemple, via une distance ou un kernel :

$$S\left(\mathbf{a}^{(m)}, \mathbf{a}^{(m')}\right) = \exp(-\alpha \parallel \mathbf{a}^{(m)} - \mathbf{a}^{(m')} \parallel^2) \quad \text{(RBF kernel)},$$

ou encore une **similarité** cosinus, $\frac{\mathbf{a}^{(m)}\cdot\mathbf{a}^{(m')}}{\|\mathbf{a}^{(m)}\|\|\mathbf{a}^{(m')}\|}$.

Si la durée globale est T et la longueur d'une trame Δ, on obtient approximativement $\frac{T}{\Delta}$ frames. D'un point de vue **mathématique**, un SCN naïf aurait alors $O(n^2)$ liens potentiels, où $n = \frac{T}{\Delta}$. Pour des enregistrements longs ou un pas Δ très petit, il devient nécessaire de **sparsifier** (Chap.

7.2.3) en limitant les connexions aux trames voisines en temps ou présentant une forte ressemblance.

B. Cohérence Temporelle et Séquences de Frames

Le **DSL** tient compte de la **cohérence** entre frames adjacentes $\mathbf{a}^{(m)}, \mathbf{a}^{(m+1)}$. Dans la plupart des signaux audio, deux trames successives sont similaires, surtout si elles appartiennent au même phonème ou au même effet sonore. Par la **règle** DSL :

$$\omega_{m,m+1}(t+1) = \omega_{m,m+1}(t) + \eta\left[S\left(\mathbf{a}^{(m)}, \mathbf{a}^{(m+1)}\right) - \tau\,\omega_{m,m+1}(t)\right],$$

on s'attend à ce que $\omega_{m,m+1}$ devienne élevé lorsque la transition est fluide (même son, même tonalité).

Pour la **parole**, un phonème s'étend sur quelques frames. La similarité demeure élevée dans ce segment, puis chute à la frontière. Le **SCN** regroupe alors les frames cohérents en un **cluster** local, identifiant potentiellement un segment acoustique "unifié" (voyelle, consonne, etc.). Cette identification n'est pas imposée, mais **émerge** de la dynamique ω.

C. Mécanisme d'Auto-Organisation pour la Séquence Audio

Chaque paire (m, m') (frames indices m, m') voit sa liaison $\omega_{m,m'}(t)$ évoluer selon :

$$\omega_{m,m'}(t+1) = \omega_{m,m'}(t) + \eta\left[S\left(\mathbf{a}^{(m)}, \mathbf{a}^{(m')}\right) - \tau\,\omega_{m,m'}(t)\right].$$

Les liaisons inter-frames s'en trouvent **renforcées** si la ressemblance audio (MFCC, spectre, embedding) est réelle, et s'**amenuisent** dans le cas contraire.

Au fur et à mesure des itérations, il se forme des **clusters** de frames contigües ou semblables, traduisant la persistance d'un même phonème ou d'une même note musicale. La **dynamique** DSL fait ressortir la discontinuité là où le son change notablement (d'un phonème [a] à [i], ou d'un accord musical à un autre).

D. Avantages et Aspects Mathématiques

Le **DSL** évite de figer une segmentation en unités phonétiques ou syllabiques. La *force* de $\omega_{m,m'}$ reflète naturellement les proximités spectrales. D'un point de vue **analytique**, on se rapproche d'un **clustering** sans paramètre figé, où la continuité est "décidée" localement par le niveau de similarité.

Si on ajoute une **seconde modalité** (par ex. **vidéo** synchronisée), le SCN peut relier la frame audio $\mathbf{a}^{(m)}$ à la frame vidéo $\mathbf{v}^{(m)}$ si une synergie inter-modale est définie. On récupère alors la **cohérence audio-visuelle**, détectant par exemple un mouvement labial correspondant au son produit (voir 8.3.3.2).

La complexité $O(n^2)$ avec $n = \frac{T}{\Delta}$ demeure un défi. On peut, pour réduire les comparaisons inutiles, s'en tenir à un voisinage temporel $|m - m'| \leq w$ ou à un k-NN sur les embeddings audio, aboutissant à une structure plus éparse et un calcul de synergie plus sélectif.

8.5.2. Synergie Audio-Audio

Dans le contexte **multimodal**, l'analyse **audio-audio** consiste à évaluer la **synergie** entre deux **flux sonores** ou deux *segments* d'enregistrement. L'objectif est de déterminer leur **similarité**

ou leur **complémentarité** (cf. chap. 8.1) pour, par exemple, fusionner des signaux cohérents (mêmes sources, mêmes événements sonores), ou au contraire inhiber ceux qui apportent du bruit ou des informations redondantes.

8.5.2.1. Similarité de signatures sonores (distance euclidienne, cosinus)

Dans un **SCN** (Synergistic Connection Network) dédié au traitement ou à la fusion de flux **audio**, la notion de "**signature sonore**" d'un segment (ou d'un flux complet) s'avère déterminante. Chaque segment $\mathcal{A}_i$ est associé à un **vecteur** $\mathbf{x}_i \in \mathbb{R}^d$, résultant par exemple d'une extraction MFCC ou d'un **embedding** audio. Le **Deep Synergy Learning** (DSL) doit ensuite définir une **synergie** $S(\mathcal{A}_i, \mathcal{A}_j)$ reflétant la proximité ou la similarité entre deux de ces représentations. Les mesures les plus courantes pour comparer deux vecteurs $\mathbf{x}_i$ et $\mathbf{x}_j$ incluent la **distance euclidienne** et la **similarité cosinus**.

A. Représentation Vecteur du Segment Audio

Chaque segment $\mathcal{A}_i$ correspond à un bloc temporel ou un ensemble de frames audio (voir chap. 8.5.1.3). On peut construire un **vecteur** $\mathbf{x}_i \in \mathbb{R}^d$ en :

- Effectuant la **moyenne** ou la **concaténation** de MFCC sur plusieurs frames,
- Extrayant un **patch** du spectrogramme qu'on linéarise ou qu'on résume par un CNN,
- Employant un **embedding** profond (type Audio2Vec, autoencodeur, etc.).

D'un point de vue **mathématique**, le vecteur $\mathbf{x}_i$ est donc l'unité essentielle pour caractériser la "signature" d'un segment audio $\mathcal{A}_i$.
Une fois ces vecteurs audio $\{\mathbf{x}_i\}$ définis, la **synergie** $S(\mathcal{A}_i, \mathcal{A}_j)$ se fonde sur une **distance** ou une **similarité**. Le DSL utilisera cette valeur pour décider, via la mise à jour $\omega_{i,j} \leftarrow \omega_{i,j} + \eta[S(i,j) - \tau\,\omega_{i,j}]$, si deux segments audios se connectent fortement ou non dans le **SCN**.

B. Définition d'une Distance ou d'une Similarité

Une approche directe consiste à employer la **distance** L^2 entre les deux vecteurs $\mathbf{x}_i$ et $\mathbf{x}_j$. On définit :

$$d_{ij}^{\text{eucl}} = \| \mathbf{x}_i - \mathbf{x}_j \|_2 = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2}.$$

Puis on convertit éventuellement cette distance en une **similarité** $S(i,j)$:

$$S(i,j) = \exp\left(-\alpha \| \mathbf{x}_i - \mathbf{x}_j \|^2\right) \quad \text{ou} \quad \frac{1}{1+\| \mathbf{x}_i - \mathbf{x}_j \|^2}.$$

Une distance $\| \mathbf{x}_i - \mathbf{x}_j \|$ faible implique $S(i,j)$ grand, signifiant une **synergie** élevée entre les segments audio correspondants.

D'autres travaux préfèrent la **similarité cosinus** :

$$\cos(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\| \mathbf{x}_i \| \, \| \mathbf{x}_j \|}.$$

Cette mesure met en avant l'**angle** entre $\mathbf{x}_i$ et $\mathbf{x}_j$ plutôt que leur écart en norme. Pour des signaux audio qui peuvent différer fortement en amplitude, la cosinus-sim fournit un indice de "forme" spectrale ou "orientation" dans l'espace des features, indépendamment du volume.

Dans le **Deep Synergy Learning**, si on emploie la cosinus-sim, on peut poser :

$$S(\mathcal{A}_i, \mathcal{A}_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\| \mathbf{x}_i \| \, \| \mathbf{x}_j \|}.$$

Les segments ayant des signatures sonores analogues obtiennent un $S(\cdot,\cdot)$ proche de 1. La pondération $\omega_{i,j}$ grandit donc si la synergie $S(i,j)$ demeure > 0, fidélisant la liaison entre ces deux entités audio et aboutissant à leur regroupement dans le **SCN**.

C. Considérations Mathématiques

Si on choisit la **distance euclidienne**, il peut s'avérer crucial de **normaliser** ou de **standardiser** les vecteurs $\mathbf{x}_i$ (voir chap. 8.5.1.2). Une grande amplitude due à un enregistrement plus "fort" ou plus "long" ne doit pas se traduire en une distance faussée. Par contraste, la cosinus-sim s'accommode mieux des variations d'amplitude.

Le vecteur $\mathbf{x}_i \in \mathbb{R}^d$ peut être de taille réduite (ex. 13–39 MFCC) ou plus volumineux (embedding de 128 ou 512). Le **coût** de calcul pour comparer tous les couples (i,j) est alors $O(n^2\, d)$. Dans un SCN volumineux, on recourt souvent à des techniques de **sparsification** (k-NN, etc.) pour limiter la croissance quadratique.

Une fois la **similarité** cos ou la distance RBF choisie, la **règle** DSL :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big]$$

suit son cours. Si la ressemblance $S(i,j)$ dépasse un seuil, $\omega_{i,j}$ est renforcé, traduisant un **cluster** audio–audio se consolidant.

D. Applications

Lorsque plusieurs canaux enregistrent la même source sonore, ou lorsque des segments audio récurrents apparaissent, la **similarité** (distance faible, cosinus élevé) conduit le DSL à renforcer ω. On agrège ces segments, aboutissant à des super-nœuds représentant, par exemple, la même **voix**, le même **instrument** ou la même **séquence** musicale.

Dans le cadre d'une base de segments audio, la mise à jour DSL engendre des **clusters** d'entités audio partageant un timbre, un locuteur, ou une signature spectrale caractéristique. L'utilisateur peut ensuite repérer ces clusters (ex. "Segment audio 12 est relié aux segments 17, 45, 89… relevant du même événement sonore").

8.5.2.2. Détection d'Événements Communs (2 flux audio captant la même source ?)

Il est fréquent de disposer de **plusieurs flux audios** enregistrés en parallèle (plusieurs microphones, ou plusieurs canaux stéréo/5.1) qui couvrent la même scène ou le même espace sonore. L'un des objectifs est alors de **détecter** si un **événement particulier** (une voix, un objet qui chute, un claquement) se produit et est enregistré **simultanément** par ces canaux. Le **Deep**

Synergy Learning (DSL), au sein d'un **Synergistic Connection Network** (SCN), permet de modéliser un tel cas en définissant une **synergie** entre les trames (ou segments) de chaque flux, puis en laissant la dynamique ω révéler les correspondances.

A. Contexte : Fusion Audio dans le DSL

On se place dans la situation où l'on dispose de deux enregistrements :

$$\mathcal{A}_1 = \{\mathcal{E}_i^{(1)}\}_{i\in I_1}, \quad \mathcal{A}_2 = \{\mathcal{E}_j^{(2)}\}_{j\in I_2},$$

chaque $\mathcal{E}_i^{(1)}$ et $\mathcal{E}_j^{(2)}$ étant un segment ou une trame dans le temps (par exemple 20 ms de signal). Les flux peuvent être synchronisés temporellement (positions de micro différentes) ou légèrement décalés. L'idée est de comparer ces trames pour y repérer des "pics" de similarité au même instant.

Pour associer la trame $\mathcal{E}_i^{(1)}$ au flux 1 et la trame $\mathcal{E}_j^{(2)}$ au flux 2, on calcule une **similarité** ou un **score** $S(i,j)$ reflétant la correspondance acoustique :

$$S\left(\mathcal{E}_i^{(1)}, \mathcal{E}_j^{(2)}\right) = \rho\left(\mathbf{x}_i^{(1)}, \mathbf{x}_j^{(2)}\right),$$

où $\mathbf{x}_i^{(1)}$ et $\mathbf{x}_j^{(2)}$ sont, par exemple, des vecteurs MFCC, ou des vecteurs de spectrogramme, ou encore un embedding. La fonction ρ peut être une **similarité cosinus**, une **distance** inversée, ou même un **coefficient** de corrélation (Pearson) pour capturer la ressemblance temporelle.

Si la **dynamique** DSL **renforce** une liaison $\omega_{(i,j)}$ entre la trame $\mathcal{E}_i^{(1)}$ et $\mathcal{E}_j^{(2)}$, cela indique qu'ils "entendent" le **même** événement, un même son capté par les deux micros. Cette liaison devient un **témoin** d'un événement commun, éventuellement mis en évidence par un **seuil** θ ou par l'analyse de clusters dans le SCN.

B. Mise en Forme Mathématique

Dans le **SCN**, chaque **trame** $\mathcal{E}_i^{(1)}$ du flux 1 peut se lier à chaque trame $\mathcal{E}_j^{(2)}$ du flux 2. On définit donc une liaison $\omega_{(i,j)}(t)$. La **mise à jour** DSL s'écrit :

$$\omega_{(i,j)}(t+1) = \omega_{(i,j)}(t) + \eta\left[S\left(\mathcal{E}_i^{(1)}, \mathcal{E}_j^{(2)}\right) - \tau\,\omega_{(i,j)}(t)\right].$$

Une synergie $S(\cdot,\cdot)$ suffisamment élevée fait **croître** la pondération $\omega_{(i,j)}$. Sur le plan **analytique**, on aboutit (en régime stationnaire) à $\omega_{(i,j)} \approx S(i,j)/\tau$ si rien ne perturbe la dynamique.

Pour "comparer" utilement les segments $\mathcal{E}_i^{(1)}$ et $\mathcal{E}_j^{(2)}$, on peut exiger qu'ils se situent à peu près au **même** instant ($i \approx j$ si les flux sont synchronisés) ou introduire un paramètre de décalage δ. Cela évite de connecter $\mathcal{E}_i^{(1)}$ (temps t) à $\mathcal{E}_j^{(2)}$ (temps t') s'ils ne se chevauchent pas dans la dimension temporelle.

Une fois la **mise à jour** itérative de ω stabilisée, on peut poser un **seuil** afin que si $\omega_{(i,j)}(t)$ dépasse θ, on déclare qu'un **événement** est simultanément perçu dans les deux flux sur les trames $\mathcal{E}_i^{(1)}$ et $\mathcal{E}_j^{(2)}$. D'un point de vue **mathématique**, la formation d'un **cluster** regroupant $\{(i_1,j_1),(i_2,j_2),\dots\}$ rend compte d'un événement plus large ou plus prolongé.

C. Critères de Similarité Audio

Le plus simple est de comparer des vecteurs $\mathbf{x}_i^{(1)}$ et $\mathbf{x}_j^{(2)}$ (MFCC, spectral, embedding) via un **produit scalaire** (cosinus) :

$$S\left(\mathcal{E}_i^{(1)}, \mathcal{E}_j^{(2)}\right) = \frac{\mathbf{x}_i^{(1)} \cdot \mathbf{x}_j^{(2)}}{\| \mathbf{x}_i^{(1)} \| \, \| \mathbf{x}_j^{(2)} \|}.$$

Si ce score est élevé, on conclut que ces trames ont un contenu sonore proche. On peut aussi employer la **distance euclidienne**, ou un **coefficient** de corrélation de Pearson, notamment si la phase du signal (ou l'enveloppe) importe.

Dans le cas de bruits impulsifs ou de signaux brefs, on peut réaliser une **cross-corrélation** pour repérer le meilleur décalage temporel Δt. Une forte valeur de cette cross-corrélation indique que les deux segments captent la même source (ex. un claquement de mains arrivé au micro 1 puis au micro 2 avec un léger retard). On peut alors convertir la valeur de pic en un score $S(i,j)$.

D. Synchronisation et Structure du SCN

Dans le **DSL**, si un même son se produit, la synergie $S(i,j)$ devient forte pour les segments $\mathcal{E}_i^{(1)}$ et $\mathcal{E}_j^{(2)}$ englobant cet instant. La mise à jour $\omega_{(i,j)}(t+1) = \omega_{(i,j)}(t) + \eta\left[S(i,j) - \tau\, \omega_{(i,j)}(t)\right]$ amplifie la liaison $\omega_{(i,j)}$. On peut alors identifier le "cluster" de liens forts reliant flux 1 et flux 2 aux mêmes instants. Cela met en évidence l'existence et la durée de l'événement (une voix, une percussive…).

Cette logique s'étend si on possède plus de deux flux (ex. 3 ou 4 micros). Le **SCN** englobe alors de multiples couples $\omega_{(i,j,k,\ldots)}$ ou s'appuie sur un graphe plus riche. Le concept demeure, la synergy S s'appuie sur la similarité ou la corrélation acoustique, et la **dynamique** d'auto-organisation fait émerger un "super-nœud" audio commun.

8.5.2.3. Inhibition si bruit ou signaux incohérents

Dans un contexte **multimodal** où plusieurs flux (audio, vision, texte) se combinent au sein d'un **SCN** (Synergistic Connection Network), il arrive qu'un ou plusieurs de ces flux soient **bruités** ou **incohérents**. L'un des enjeux est de **préserver** l'auto-organisation du **DSL** (Deep Synergy Learning) malgré ces perturbations, afin que les liens $\omega_{i,j}$ se focalisent sur les signaux fiables. L'**inhibition** joue alors un rôle de **contrôle** en limitant ou diminuant la pondération attachée aux canaux jugés douteux, empêchant ainsi la formation de clusters artificiels induits par le bruit.

A. Logique d'Inhibition vis-à-vis du Bruit

Il arrive qu'une **modalité** donnée (par exemple un micro défectueux, ou un flux vidéo recouvert de neige numérique) produise des **données** fortement dispersées ou contradictoires par rapport aux autres sources. La **mise à jour** du DSL, si elle reposait uniquement sur une synergie "brute" $S(i,j)$, risquerait de connecter abusivement ces entités au reste du réseau. Pour y remédier, on introduit un **terme** d'**inhibition** qui pénalise la pondération $\omega_{i,j}$ lorsqu'une entité $\mathcal{E}_i$ est trop "bruyante".

On peut formaliser cela en définissant un **score** de bruit $B_i \geq 0$ pour l'entité $\mathcal{E}_i$. Si B_i est élevé, cela signale que la modalité de $\mathcal{E}_i$ est peu fiable ou contaminée par du bruit. Dans la règle de mise à jour, on ajoute un terme proportionnel à B_i pour **diminuer** $\omega_{i,j}$. Par exemple :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big] - \gamma\,\big(\delta\,B_i + \delta\,B_j\big)\,\omega_{i,j}(t).$$

Ici, $\gamma > 0$ et $\delta > 0$ sont des coefficients de réglage. L'idée est que plus B_i ou B_j est grand, plus le lien $\omega_{i,j}$ se voit "poussé vers le bas" (inhibé).

Si un micro donne un **signal** très fluctuant ou saturé, $B_{\text{micro}} \gg 0$ et la dynamique DSL diminue $\omega_{\text{micro},\dots}$. Ce canal ne crée plus de liaisons parasites dans le **SCN**, et les clusters formés par les flux restants demeurent cohérents. De même, si un flux vidéo est en "panne" (images noires), on peut affecter B_{video} élevé pour neutraliser son influence.

B. Incohérence entre Modalités

Au-delà du bruit "aléatoire", il peut survenir des **contradictions** entre deux modalités. Par exemple, un flux textual indique un certain mot ("chat"), tandis qu'une image montre en réalité un "chien". Si la **similarité** brute $S(i,j)$ n'arrive pas à distinguer cette incohérence (peut-être parce que l'extraction de features visuelles est ambiguë), on souhaite **pénaliser** leur liaison $\omega_{i,j}$.

On peut introduire un "facteur d'incohérence" $C_{i,j} \geq 0$ dès qu'un superviseur (ou un module externe) repère une contradiction sémantique ou temporelle.

La mise à jour DSL standard :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big]$$

se voit complétée d'un **terme** d'inhibition :

$$-\gamma\,C_{i,j}\,\omega_{i,j}(t),$$

ce qui donne :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big] - \gamma\,C_{i,j}\,\omega_{i,j}(t).$$

Quand $C_{i,j}$ est élevé, la pondération $\omega_{i,j}$ se trouve **freinée**, évitant un renforcement artificiel entre deux entités mal assorties.

C. Perspective Mathématique et Algorithmique

Dans la vision "énergie" (cf. chapitres analytiques), ajouter un terme de la forme $\gamma\,C_{i,j}\,\omega_{i,j}^2$ ou $\gamma\,C_{i,j}\,\omega_{i,j}$ fait croître l'énergie $\mathcal{J}\big(\{\omega_{i,j}\}\big)$ lorsque des liens s'établissent entre entités incohérentes. Cela provoque la **répulsion** ou la **dissuasion** de tels liens dans le schéma global.

Au niveau **itératif**, la formule "inhibition" agit à chaque pas et dépend de la pondération elle-même. Autrement dit, plus $\omega_{i,j}$ est élevé tout en ayant un $C_{i,j}$ non négligeable, plus on "soustrait" d'incrément dans la mise à jour. Cela évite qu'un bruit passager ou un désaccord modal ne provoque un cluster incorrect.

8.5.3. Fusion Audio–Vision ou Audio–Texte

Dans de nombreux domaines multimédias (reconnaissance de scènes, sous-titrage automatique, visioconférence intelligente, etc.), la **fusion** de sources **audio** et **visuelles** (ou audio et textuelles) améliore considérablement la robustesse et la qualité d'interprétation. Le **DSL** (Deep Synergy Learning), par sa logique de $\omega_{i,j}$ adaptatifs et ses stratégies d'auto-organisation, offre un cadre pour relier les entités provenant de flux hétérogènes (ex. frames vidéo, segments audio, tokens textuels) en un **SCN** synergique. Cette section **8.5.3** détaille les deux grandes approches de fusion, l'**audio–vision** (8.5.3.1) et l'**audio–texte** (8.5.3.2).

8.5.3.1. Audio–Vision : reconnaissance vidéo-audio conjointe (lip sync, ambiance sonore)

Les applications conjuguant un **flux vidéo** et un **flux audio** sont nombreuses et recouvrent des scénarios variés, comme la **lip sync** (analyse de la cohérence entre lèvres et voix), la **détection** d'événements sonores (sirènes, applaudissements), ou la **compréhension** d'une scène globale (apparition d'un véhicule visible accompagné d'un bruit de moteur). Le cadre du **DSL** (Deep Synergy Learning) et, en particulier, l'organisation du **SCN** (Synergistic Connection Network), facilite la **fusion** de ces deux modalités, car il autorise une représentation unifiée des **entités** (frames vidéo et segments audio) et un **calcul** itératif des **liens** $\omega_{i,j}$ reflétant leur **synergie**. L'objectif est de montrer comment cette synergie peut se définir et se mettre à jour pour aboutir à une **reconnaissance conjointe** plus robuste et plus fine.

A. Représentation et Synergie Audio–Vision

Une **séquence vidéo** peut être découpée en **frames** $\{\mathbf{v}_t\}_{t=1,\dots}$ s'échelonnant dans le temps selon un pas de capture (par exemple 25 images par seconde). Chaque frame $\mathbf{v}_t$ est considérée comme une **entité** $\mathcal{E}_{\text{vis},t}$. De son côté, le **flux audio** associé, qui s'étend sur la même période temporelle, est segmenté en petits blocs ou trames $\{\mathbf{a}_u\}_{u=1,\dots}$, chacune représentant un fragment d'onde (voir 8.5.1.3 sur le traitement de séquences audio). Chaque segment audio $\mathbf{a}_u$ devient une **entité** $\mathcal{E}_{\text{aud},u}$. Le **SCN** regroupe ainsi un ensemble mixte d'entités, comprenant des frames visuelles et des segments audio, prêtes à être reliées si elles présentent une **cohérence** temporelle ou sémantique.

Pour quantifier la **compatibilité** entre un frame vidéo $\mathbf{v}_t$ et un segment audio $\mathbf{a}_u$, on définit une fonction de **synergie** $S(\mathbf{v}_t, \mathbf{a}_u)$. Sur le plan mathématique, on peut recourir à une transformation

$$\mathbf{z}_{\text{vis}}(t) \;=\; \Phi_{\text{vision}}(\mathbf{v}_t), \quad \mathbf{z}_{\text{aud}}(u) \;=\; \Phi_{\text{audio}}(\mathbf{a}_u),$$

où Φ_{vision} et Φ_{audio} sont des réseaux ou des modules extrayant des **embeddings**. La synergie s'exprime ensuite par

$$S(\mathbf{v}_t, \mathbf{a}_u) \;=\; \rho\big(\mathbf{z}_{\text{vis}}(t),\, \mathbf{z}_{\text{aud}}(u)\big),$$

où $\rho(\cdot,\cdot)$ peut être une **similarité cosinus**, un **kernel** Gaussien, ou encore un **coefficient** de corrélation temporel (si l'on cherche un alignement plus précis). Des considérations de **fenêtrage** (κ) peuvent s'ajouter si l'on estime devoir modéliser un léger décalage Δt entre l'événement visuel et le son correspondant.

B. Mécanisme d'Auto-Organisation dans le SCN

Le **DSL** actualise les liaisons $\omega_{(\text{vis},t),(\text{aud},u)}$ selon la **règle** habituelle :

$$\omega_{(t,u)}(k+1) = \omega_{(t,u)}(k) \; + \; \eta\big[S(\mathbf{v}_t, \mathbf{a}_u) - \tau\, \omega_{(t,u)}(k)\big],$$

où $\omega_{(t,u)} \equiv \omega_{(\mathrm{vis},t),(\mathrm{aud},u)}$. Si $S(\mathbf{v}_t, \mathbf{a}_u)$ se maintient à un niveau élevé (par exemple, forte corrélation lip sync, ou ambiance sonore parfaitement concordante avec ce que montre la vidéo), la pondération $\omega_{(t,u)}$ croît. Dans le cas contraire, elle retombe vers zéro. Les entités (frames, segments audio) se **regroupent** donc au sein d'un **cluster** si elles perçoivent le même phénomène (voix synchronisée, bruit d'objets visibles, etc.).

Le **cas** de la lip sync se formalise en associant par exemple un vecteur $\mathbf{l}_t$ décrivant la **forme labiale** détectée dans $\mathbf{v}_t$, et un vecteur $\mathbf{p}_u$ décrivant la **phonétique** principale de $\mathbf{a}_u$. La synergie s'écrit :

$$S(\mathbf{v}_t, \mathbf{a}_u) = \rho(\mathbf{l}_t,\, \mathbf{p}_u) \cdot \kappa(|t-u|),$$

indiquant qu'on pèse la similarité labiale–phonétique par la proximité temporelle $|t-u|$. Une fois insérée dans la dynamique $\omega_{(t,u)}(k+1) = \cdots$, les paires (t,u) véritablement alignées voient leurs liaisons ω se consolider.

Un autre usage est la **reconnaissance d'événements** comme un "orchestre en train de jouer", où le flux vidéo repère des instruments, des musiciens ; le flux audio capte le timbre musical correspondant. La synergie $S(\mathbf{v}_t, \mathbf{a}_u)$ prend alors une forme plus large, par exemple un **embedding** cross-modal dérivé d'un entraînement supervisé (Chap. 8.3.3.1 sur texte–image, transposable à audio–vision). Les frames associées à l'instrument musical se lient alors fortement avec les segments audio contenant des ondes musicales similaires.

C. Résultats et Bénéfices

En fin de convergence, le **SCN** engendre des **clusters** reliant plusieurs frames vidéo $(t_1, t_2, \ldots)$ aux segments audio $(u_1, u_2, \ldots)$ jugés "similaires" ou "synchrones". Cette structure en **macro-nœuds** reflète un même *événement* ou *objet* perçu simultanément dans les deux modalités. L'analyse de la scène s'en trouve enrichie, un vecteur "voix–visage" est plus robuste qu'une simple piste audio isolée ou qu'un visage muet.

Le fait de disposer de deux sources (audio et vision) permet de **compenser** les failles de l'une via l'autre. Si la qualité vidéo est dégradée, le son confirme ou infirme la scène ; si l'audio est parasité, la vidéo maintient un indice visuel. Le **DSL** gère ces divergences en modulant $\omega_{(t,u)}$ selon la valeur $S(\mathbf{v}_t, \mathbf{a}_u)$. Les saturations ou bruits (Chap. 8.5.2.3) peuvent être traités par un **terme d'inhibition** freinant les liaisons issues d'un flux peu fiable.

Le système peut s'étendre à un flux audio supplémentaire ou à une modalité textuelle. Un **SCN** plus large comprendra des entités $\mathcal{E}_{\mathrm{vis},t}, \mathcal{E}_{\mathrm{aud1},u}, \mathcal{E}_{\mathrm{aud2},v}$ voire $\mathcal{E}_{\mathrm{txt},k}$ si l'on annexe des sous-titres ou légendes. La logique DSL reste la même en calculant $S(i,j)$ et en mettant à jour $\omega_{i,j}$ pour toute paire susceptible de présenter une synergie. Le graphe final illustre la **fusion** de multiples flux dans un réseau auto-organisé et localement cohérent.

8.5.3.2. Audio–Texte : Alignement transcription-voix, synergie si corrélation forte

Un **DSL** (Deep Synergy Learning) multimodal peut se doter d'un **SCN** (Synergistic Connection Network) dédié à relier des **segments audio** (par ex. trames vocales) à des **éléments textuels** (tokens, morceaux de phrase) pour former un **alignement** plus ou moins explicite entre le **signal acoustique** et la **transcription** proposée. La notion de **corrélation** ou de **cohérence** audio–

texte s'avère déterminante. Dès lors que la **synergie** entre un segment acoustique $\mathcal{E}_{\text{aud},m}$ et un segment textuel $\mathcal{E}_{\text{txt},n}$ dépasse un certain seuil (ou s'avère nettement plus grande que pour d'autres paires), la **pondération** $\omega_{(m,n)}$ se renforce, rendant l'alignement plus stable. Les paragraphes suivants détaillent les fondements mathématiques de cet alignement et son intégration dans la dynamique DSL.

A. Représentation Audio–Texte et Segments

Sur le plan **mathématique**, on découpe un **flux audio** continu en petites **trames** (ex. 20 ou 30 ms), chacune représentée par un **vecteur** $\mathbf{a}_m \in \mathbb{R}^d$ (ex. MFCC, spectrogramme résumé, ou embedding neuronal). On obtient alors un ensemble $\mathcal{A} = \{\mathbf{a}_1, \dots, \mathbf{a}_M\}$. On dispose également d'une séquence textuelle $\mathcal{T} = \{t_1, \dots, t_N\}$, où chaque t_n est un **token** (mot, sous-mot, phonème, caractère), et l'on peut associer à chaque token t_n un vecteur $\mathbf{x}_n \in \mathbb{R}^{d'}$ (embedding lexical, projection phonémique, etc.).

Dans un **SCN** multimodal, chaque trame audio $\mathcal{E}_{\text{aud},m}$ et chaque token textuel $\mathcal{E}_{\text{txt},n}$ deviennent des **nœuds**. On cherche à quantifier une **synergie**

$$S\big(\mathcal{E}_{\text{aud},m},\, \mathcal{E}_{\text{txt},n}\big) \;=\; S(\mathbf{a}_m,\; \mathbf{x}_n)$$

indicative de la **cohérence** entre $\mathbf{a}_m$ (descripteur audio) et $\mathbf{x}_n$ (descripteur textuel).

B. Définition de la Synergie Audio–Texte

Si l'on évalue la correspondance "voix" vs. "transcription", on peut modéliser chaque trame audio $\mathbf{a}_m$ comme un **embedding phonétique**, tandis que chaque token t_n porte un embedding sémantique ou phonétique. On définit alors

$$\rho(\mathbf{a}_m, \mathbf{x}_n) \;=\; \exp(-\alpha \;\|\; \Phi(\mathbf{a}_m) - \Psi(\mathbf{x}_n) \;\|^2),$$

où $\Phi(\cdot)$ et $\Psi(\cdot)$ sont deux **projections** (réseaux ou mappings) transformant la trame audio et le token dans un espace *common* pour la comparaison. Plus $\rho(\mathbf{a}_m, \mathbf{x}_n)$ se rapproche de 1, plus la trame $\mathbf{a}_m$ correspond à la prononciation du token t_n.

Le **DSL** met à jour la pondération $\omega_{(m,n)}$ reliant $\mathcal{E}_{\text{aud},m}$ et $\mathcal{E}_{\text{txt},n}$. La formulation standard est :

$$\omega_{(m,n)}(k+1) = \omega_{(m,n)}(k) + \eta\big[S(\mathbf{a}_m, \mathbf{x}_n) - \tau\, \omega_{(m,n)}(k)\big].$$

Ainsi, si $S(\mathbf{a}_m, \mathbf{x}_n) = \rho(\mathbf{a}_m, \mathbf{x}_n)$ se maintient à une valeur haute, la pondération $\omega_{(m,n)}$ croit au fil des itérations, marquant un **alignement** fort entre la trame audio m et le token textuel n. En revanche, si $\rho(\mathbf{a}_m, \mathbf{x}_n) \approx 0$, la liaison retombe vers zéro et la trame audio demeure non alignée à ce token textuel.

C. Alignement Temporel et Contrainte de Séquence

Dans un usage **classique** de speech-to-text, la prononciation des tokens suit l'ordre du texte. La trame audio $\mathbf{a}_m$ ne saurait correspondre à un token t_n précédé par un autre token t_{n+1} aligné avec $\mathbf{a}_{m'}$ où $m' < m$. Cette **monotonie** peut être intégrée dans le **SCN** en ajoutant un **terme** de pénalisation ou en restreignant les liaisons candidates $\omega_{(m,n)}$. Mathématiquement, on veille à ce que

$$m < m' \quad \Rightarrow \quad n \;\leq\; n',$$

limitant la configuration des liaisons $\{\omega_{m,n}\}$ dans le graphe.

Le **DSL** s'exécute toutefois localement et peut tolérer quelques violations ou retards si la corrélation le justifie (accents, sur-articulation). On obtient un alignement final plus **souple** qu'un forçage déterministe. Les liaisons fortes $\omega_{(m,n)}$ révèlent la séquence audio–texte la plus probable dans l'espace des correspondances.

D. Bénéfices et Applications

Dans un **pipeline** classique de speech-to-text, la liaison $\mathbf{a}_m \to t_n$ est déterminée par un modèle HMM ou un transformeur. Le **DSL** peut **améliorer** ou **réviser** cet alignement. Si un mot t_n semble mal placé (faible corrélation avec les trames audio en regard), la mise à jour $\omega_{(m,n)} \approx 0$ le met en évidence ; un mot plus approprié $t_{n'}$ pourra recevoir plus de pondération.

Si le **texte** n'est pas fidèle à la bande audio, les **valeurs** $\rho(\mathbf{a}_m, \mathbf{x}_n)$ restent basses, entraînant l'échec du renforcement $\omega_{(m,n)}$. On repère ainsi des divergences, ou on crée un "texte orphelin" (aucune trame audio ne se relie vraiment à lui).

Une fois l'**audio** aligné au **texte**, on peut **fusionner** un flux vidéo de la même scène. La dynamique DSL prend en charge **trois** modalités (audio, texte, vision), unifiant l'ensemble en un graphe plus riche. Chaque trame audio, token textuel et frame vidéo tente d'établir un lien de synergie, révélant des clusters tri-modaux (ex. "Cette personne vue à la caméra prononce ce mot, au moment T").

8.5.3.3. Exemples : SCN reliant segments audio et frames vidéo pour une scène de film

Dans de nombreux scénarios multimédias, tels que l'analyse de scènes cinématographiques, la compréhension de flux vidéo-audio, ou encore la recherche d'événements sonores dans un film, l'association automatique entre des **segments audio** (voix, bruitages, musique) et des **plans vidéo** (images clés, frames) se révèle primordiale. Le **SCN** (*Synergistic Connection Network*), dans le cadre du **DSL** (*Deep Synergy Learning*), fournit un cadre mathématique pour **fusionner** ces deux modalités de manière adaptative et **auto-organisée**.

A. Structuration des Entités : Segments Audio et Frames Vidéo

On considère la piste sonore d'une **scène** de film, que l'on découpe en segments $\{A_1, A_2, \dots\}$. Chaque segment A_i est choisi selon des critères temporels (découpage en intervalles de quelques secondes) ou des changements détectés (début/fin d'un dialogue, rupture dans le niveau sonore, changement de musique, etc.). Sur le plan **mathématique**, chaque segment audio $\mathcal{E}_i^{(\text{audio})}$ peut être muni d'un **vecteur** $\mathbf{a}_i \in \mathbb{R}^{d_a}$, lequel encapsule des descripteurs (MFCC, spectrogrammes abrégés, ou embeddings neuronaux d'Audio2Vec). Selon la complexité, on peut également recourir à des indicateurs plus symboliques (par ex. "voix personnage A", "bruit moteur", etc.).

Côté vidéo, on prélève des **frames clés** ou on segmente la scène en **plans** plus longs $\{V_1, V_2, \dots\}$. Chaque entité $\mathcal{E}_j^{(\text{video})}$ correspond alors à une image (frame) ou un bloc d'images (plan), associé(e) à un **embedding** visuel $\mathbf{v}_j \in \mathbb{R}^{d_v}$ (issu, par exemple, d'un réseau type ResNet, VGG, ou d'un transformeur visuel).

Dans une approche plus sémantique, il est possible d'enrichir ce vecteur par des métadonnées. Des informations telles que "gros plan sur personnage X", "scène de paysage" ou "mouvement de caméra rapide", etc.

Le **SCN** vise à capturer la **synergie** $S(A_i, V_j)$ entre chaque segment audio A_i et chaque frame/plan vidéo V_j. Sur le plan **mathématique**, on peut l'exprimer comme

$$S(A_i, V_j) \;=\; \kappa(\mathbf{a}_i, \mathbf{v}_j),$$

où κ est une fonction de similarité adaptée :

- **Similarité temporelle**. Si A_i se situe sur la bande-son entre t_1 et t_2, et V_j couvre le plan vidéo entre $t_1{}'$ et $t_2{}'$, on peut définir un score fonction du recouvrement Δt.
- **Similarité sémantique**. On peut calculer un embedding "commun" pour l'audio et le visuel (type CLIP audio–vidéo) et mesurer la distance cosinus.
- **Alignement d'événement**. Un bruitage particulier (coup de feu) coïncide avec la vision d'une arme. κ alors prend en compte l'objet détecté dans l'image et le spectre caractéristique du son.

B. Mise en Place d'un SCN Audio–Vidéo

Dans le **SCN**, chaque entité $\mathcal{E}_i^{(\text{audio})}$ et $\mathcal{E}_j^{(\text{video})}$ est un **nœud** du réseau. La **pondération** $\omega_{(A_i,V_j)}$ reflète la "force" de lien, c'est-à-dire le **degré** de corrélation ou de compatibilité entre le segment audio A_i et le plan vidéo V_j.

La règle de **mise à jour** DSL repose sur :

$$\omega_{(A_i,V_j)}(t+1) \;=\; \omega_{(A_i,V_j)}(t) \;+\; \eta\;\Big[S(A_i,V_j) \;-\; \tau\,\omega_{(A_i,V_j)}(t)\Big].$$

Le **terme** $S(A_i, V_j)$ correspond à la **synergie** audio–vidéo (temporelle, sémantique, etc.), τ contrôle la décroissance, et η fixe le pas d'apprentissage.

Pour éviter qu'un segment audio A_i ne se "répande" en se liant moyennement à plusieurs plans vidéo $\{V_j\}$ à la fois, on peut inclure une **inhibition** latérale. *Mathématiquement*, cela se traduit par un **terme** supplémentaire (voir chap. 7.4) pénalisant la somme des $\omega_{(A_i,\cdot)}$ ou la création de trop nombreux liens modérés, forçant chaque segment audio à **sélectionner** les frames vidéo les plus corrélées.

À mesure que la **dynamique** s'opère, certains liens $\omega_{(A_i,V_j)}$ deviennent **forts** (forte corrélation), tandis que d'autres tombent vers 0 (peu ou pas de correspondance). Ainsi se **forment** des **clusters** multimodaux, regroupant un **ensemble** de segments audio $\{A_{i_1}, \ldots, A_{i_k}\}$ et un **ensemble** de frames vidéo $\{V_{j_1}, \ldots, V_{j_m}\}$. Sur le plan **conceptuel**, cela correspond à détecter qu'une "séquence" audio (voix, musique, bruit répétitif) s'aligne avec une "séquence" visuelle (plan fixe sur un personnage, travelling, etc.).

C. Avantages Mathématiques et Opérationnels

La **mise à jour**

$$\omega_{(A_i,V_j)}(t+1) \;=\; \omega_{(A_i,V_j)}(t) \;+\; \eta\;\left[S(A_i,V_j) \;-\; \tau\,\omega_{(A_i,V_j)}(t)\right]$$

permet, de manière **auto-organisée**, d'**assortir** chaque segment audio à son (ou ses) plan(s) vidéo correspondant(s). Cela évite un pipeline rigide de type "forçage temporel". Les *clusters* émergent de la dynamique même du SCN, d'autant plus robustement que la **synergie** $S(A_i,V_j)$ est signifiante.

Lorsque la dynamique atteint un régime stable, on peut repérer des **macro-nœuds** (grands clusters) associant plusieurs segments audios liés à plusieurs frames vidéo. Cela permet :

- **Indexation** : retrouver instantanément les parties audio associées à une portion vidéo ;
- **Recherche de scènes** : par exemple, identifier tous les endroits où un *bruit* particulier (sirène) survient en même temps qu'un *élément visuel* (véhicule de police).
- **Structure** de la scène : on comprend qu'un certain "dialogue audio" correspond à un plan rapproché sur le personnage parlant, etc.

Les segments audio *parasites* (souffle, parasites) qui ne coïncident pas visuellement avec un événement et ne montrent aucune corrélation ı$S(A_i,V_j) \approx 0$ı resteront faiblement connectés ($\omega \approx 0$) et ne perturberont pas la structure. De même, si un plan vidéo est vide (ou peu informatif), il n'entre pas en forte synergie avec un segment sonore précis. Ainsi, le SCN "filtre" les liaisons non contributives, augmentant la **lisibilité** des clusters.

D. Exemple Illustratif : Séquence de Dialogue + Action

Considérons un **exemple** où une scène de western où deux personnages discutent longuement, puis survient un coup de feu. La **bande-son** est scindée en segments :

- *Segment A* : voix du personnage 1 (2 secondes),
- *Segment B* : voix du personnage 2 (2 secondes),
- *Segment C* : coup de feu (1 seconde).

La **vidéo**, quant à elle, est découpée en *plans* :

- *Plan X* : gros plan sur le visage du personnage 1,
- *Plan Y* : plan sur le personnage 2,
- *Plan Z* : plan large dévoilant l'action (impact du coup de feu).

On crée des **nœuds** $\{\mathcal{E}_1^{(\text{audio})}, \mathcal{E}_2^{(\text{audio})}, \mathcal{E}_3^{(\text{audio})}\}$ pour les segments audio A, B, C, et $\{\mathcal{E}_1^{(\text{video})}, \mathcal{E}_2^{(\text{video})}, \mathcal{E}_3^{(\text{video})}\}$ pour les plans X, Y, Z. Les pondérations $\omega_{(A,X)}$, $\omega_{(B,X)}$, ... évoluent via la règle DSL.

- **Voix 1** (A) se synchronise temporellement avec le *plan X* (gros plan), donc $S(A,X) \approx$ fort, conduisant à $\omega_{(A,X)}$ élevé.
- **Voix 2** (B) correspond en temps au plan Y, d'où renforcement de $\omega_{(B,Y)}$.

- **Coup de feu** (C) se cale sur le plan Z (où l'on voit l'action), entraînant un $\omega_{(C,Z)}$ dominant.

À l'issue, on perçoit **trois** micro-clusters :

$\{A, X\}$ (personnage 1),

$\{B, Y\}$ (personnage 2),

$\{C, Z\}$ (coup de feu + plan large).

Cette **organisation** permet de manipuler la scène de façon **sémantique**. On peut demander de "retrouver le segment audio de la voix de personnage 1 et les plans vidéo lui correspondant", ou encore "analyser la partie action (coup de feu) pour extraire la musique ou les bruitages environnants". Le DSL a thus "auto-appris" ces liens par la dynamique de ω, sans pipeline figé.

8.6. Construction d'un SCN Multimodal Unique

Après avoir examiné comment le **DSL** (Deep Synergy Learning) s'articule dans différentes modalités (vision, langage, audio, etc.), l'étape cruciale consiste à **fusionner** l'ensemble de ces sources d'information dans un **SCN unique**. Il s'agit de placer dans le même réseau des **nœuds** représentant des objets visuels ($\{\text{image}_i\}$), des éléments textuels ($\{\text{texte}_j\}$), ou encore des extraits sonores ($\{\text{audio}_k\}$). Les **liens** entre ces nœuds, mesurés par la synergie S, peuvent alors dépasser le strict domaine intra-modal (image-image, texte-texte, etc.) pour tisser des **connexions** inter-modales (vision–texte, audio–vision, texte–audio).

8.6.1. Entités Hétérogènes dans un Même Réseau

Ce sous-chapitre se penche sur la façon de **coexister** dans un unique SCN des entités issues de différentes modalités. Sur le plan mathématique, la fonction $S(\mathcal{E}_i, \mathcal{E}_j)$ doit être **spécifiée** ou **adaptée** en fonction des types (vision, texte, audio) que revêtent $\mathcal{E}_i$ et $\mathcal{E}_j$. Cela requiert une organisation modulaire dans le calcul de la synergie (chap. 5.4) ou dans la définition de S (chap. 3.3, 3.4) afin de gérer efficacement les **paires hétérogènes**.

8.6.1.1. $\{\text{image}_i\}, \{\text{texte}_j\}, \{\text{audio}_k\}$ coexistent comme nœuds dans le SCN

Lorsqu'un **SCN** (Synergistic Connection Network) prend en charge plusieurs **modalités** telles que la vision (images), le texte et l'audio, il est possible — et souvent très fructueux — de faire **coexister** ces entités dans le **même** graphe. Dans cette perspective, chaque nœud du SCN n'est plus nécessairement une entité unimodale ; il peut tout aussi bien représenter une image, un segment textuel ou un segment audio, le tout dans une architecture commune. Cette démarche constitue un pas décisif vers la **fusion** multimodale intégrale, car elle autorise des **liens** ω entre objets visuels, extraits sonores et fragments textuels, tout en conservant la logique d'**auto-organisation** inhérente au DSL (Deep Synergy Learning).

A. Définition d'un Ensemble de Nœuds Multimodaux

Considérons un ensemble d'**images** $\{\text{image}_1, \dots, \text{image}_{N_I}\}$, un **corpus** de **textes** $\{\text{texte}_1, \dots, \text{texte}_{N_T}\}$, et un **ensemble** de segments **audio** $\{\text{audio}_1, \dots, \text{audio}_{N_A}\}$. Il est possible de réunir toutes ces entités dans une **même** structure de graphe :

$$\mathcal{U} = \{\text{image}_i : i = 1, \dots, N_I\} \cup \{\text{texte}_j : j = 1, \dots, N_T\} \cup \{\text{audio}_k : k = 1, \dots, N_A\}.$$

Toutes les entités $\mathcal{E} \in \mathcal{U}$ sont alors traitées comme des **nœuds** du **SCN**, et pourront donc être reliées entre elles par des **pondérations** $\omega_{i,j}$. La différence tient au fait qu'on manipule désormais trois modalités distinctes (vision, texte, audio) au sein de la même dynamique DSL.

B. Intégration dans un Espace Fédérateur ou via Modules Distincts

Sur le plan **mathématique**, on peut chercher à **projeter** chaque modalité dans un **espace** vectoriel **unique** $\mathbb{R}^d$. Concrètement, il s'agit de définir des transformateurs

$$\Phi_{\text{image}}(\text{image}_i) \in \mathbb{R}^d, \quad \Phi_{\text{texte}}(\text{texte}_j) \in \mathbb{R}^d, \quad \Phi_{\text{audio}}(\text{audio}_k) \in \mathbb{R}^d,$$

puis de **comparer** deux entités (même ou différente modalité) via une **similarité** ou une **distance** dans cet espace commun :

$$S(\text{image}_i,\ \text{texte}_j)\ =\ \text{sim}\left(\Phi_{\text{image}}(\text{image}_i),\ \Phi_{\text{texte}}(\text{texte}_j)\right).$$

De même pour audio–vision, audio–texte, etc. Une **similarité cosinus** ou un kernel Gaussien peut servir de base, tant que l'on garantit que chaque modalité est correctement projetée. L'avantage est que le **calcul** de $S(\cdot,\cdot)$ devient homogène, sans nécessiter de règles complexes selon la pair de modalités.

À l'inverse, on peut conserver des **représentations** différenciées (images dans $\mathbb{R}^{d_{\text{img}}}$, textes dans $\mathbb{R}^{d_{\text{txt}}}$, audio dans $\mathbb{R}^{d_{\text{aud}}}$) et confier à des "**modules**" spécifiques (voir Chap. 5.4 sur le *Module Synergie*) le soin de calculer $S(\text{image}_i, \text{texte}_j)$, $S(\text{image}_i, \text{audio}_k)$, $S(\text{texte}_j, \text{audio}_k)$, etc. Le **SCN** fait alors appel, pour chaque pair (i, j), au **module** de synergie correspondant au couple de modalités, qui renvoie un **score** de compatibilité.

Cette seconde approche se prête bien à la **diversité** des modalités (images, sons, texte), évitant de forcer un unique espace latent. Elle exige néanmoins un certain nombre de **fonctions** $S_{\text{img–txt}}$, $S_{\text{img–aud}}$, $S_{\text{txt–aud}}$, rendant potentiellement le système plus complexe à calibrer.

C. Cohabitation des Nœuds dans le Même SCN

Dans l'architecture **SCN**, tous les **nœuds** (images, textes, audios) cohabitent dans un **même** graphe. Les **liaisons** $\omega_{(i,j)}$ relient potentiellement :

- **Image–Image** : $\omega_{(\text{image}_i, \text{image}_j)}$
- **Texte–Texte** : $\omega_{(\text{texte}_i, \text{texte}_j)}$
- **Audio–Audio** : $\omega_{(\text{audio}_i, \text{audio}_j)}$
- **Cross** (Image–Texte, Texte–Audio, Audio–Image, etc.).

Chacun de ces liens se **met** à jour selon la règle DSL :

$$\omega_{(i,j)}(t+1)\ =\ \omega_{(i,j)}(t)\ +\ \eta\ \left[S(i,j)\ -\ \tau\,\omega_{(i,j)}(t)\right].$$

Où $S(i,j)$ se spécialise selon la **modalité** ou l'**embedding** commun. Par exemple, s'il s'agit de deux images, S peut être la similarité cosinus de leurs vecteurs CNN ; s'il s'agit d'un texte et d'un audio, S peut être un score de correspondance phonético-sémantique ; etc.

À mesure que les liens $\omega_{(i,j)}$ évoluent, de **clusters** émergent rassemblant un **ensemble** d'entités hétérogènes (images + textes + audios). Par exemple, on peut avoir un cluster qui regroupe :

- Une **série** d'images montrant un chat,
- Un **texte** mentionnant "chat" ou décrivant la scène,
- Un **enregistrement** audio d'un miaulement.

Le **SCN** consolide ces associations si la synergie S est élevée, aboutissant à un **macro-nœud** suggérant un concept commun ("chat qui miaule, décrit textuellement"). C'est précisément cette **auto-organisation** tri-modale qui fait la force du DSL dans un environnement riche.

D. Avantages et Points d'Attention

En autorisant toutes les **paires** (image–texte, texte–audio, audio–image) à **exister** dans le **même** SCN, on laisse la dynamique DSL **découvrir** (et non imposer) quelles correspondances méritent d'être renforcées. Il peut émerger des clusters inattendus, comme un certain son associé à une description textuelle spécifique, ou un même thème (ex. "sport") reliant plusieurs images, articles de presse (texte) et extraits sonores (clameurs de foule).

Si le nombre d'entités $N_I + N_T + N_A$ est important, le graphe peut atteindre $\mathrm{O}(N^2)$ liens potentiels. Dans ce cas, il devient **nécessaire** de recourir à des techniques de **réduction** ou de **sparsification** (k-NN, rayon ϵ, filtrage adaptatif) pour éviter un coût trop élevé en calcul. Chap. 7.2.3 évoque ces stratégies.

Il est impératif que les diverses fonctions de **synergie** (image–image, image–texte, audio–texte, etc.) soient **cohérentes** en termes d'échelle. Si l'une renvoie des scores typiquement dans [0.7,1] tandis qu'une autre fluctue dans [0,0.3], la dynamique DSL peut se trouver déséquilibrée. Une **normalisation** (chap. 8.5.1.2 sur l'amplitude/log-scale) peut s'imposer pour aligner les distributions de S.

8.6.1.2. Chaque nœud a son type, le module synergie identifie la fonction $S_{\text{vision,texte}}, S_{\text{audio,vision}}, \ldots$

Dans un **DSL** (Deep Synergy Learning) réellement multimodal, les **entités** qui composent le réseau peuvent appartenir à différentes modalités, qu'il s'agisse de vision, de texte, d'audio ou encore de données issues de capteurs. L'un des piliers de cette approche consiste à **typer** chaque nœud pour renseigner sa modalité (image, texte, segment audio…) et à sélectionner la **fonction** de synergie S appropriée lors de la mise à jour des pondérations $\omega_{i,j}$. Le **module Synergie** (chap. 5.4, 5.5) incarne précisément cette logique. Il détecte le couple de **types** (m_1, m_2) auquel appartiennent les nœuds $\mathcal{E}_i$ et $\mathcal{E}_j$, puis calcule $S_{m_1,m_2}(\mathbf{x}_i, \mathbf{y}_j)$ selon la nature des deux modalités.

A. Notion de "Type" de Nœud et Sélection de la Fonction de Synergie

Chaque **nœud** $\mathcal{E}_k$ du **SCN** se voit attribuer un **type** (ou label modal) :

$$\mathrm{type}(\mathcal{E}_k) \in \{\mathrm{Vision},\ \mathrm{Texte},\ \mathrm{Audio},\ \ldots\}.$$

Sur le plan **mathématique**, on peut considérer différents espaces $\mathcal{X}_{\text{vision}}$, $\mathcal{X}_{\text{texte}}$, $\mathcal{X}_{\text{audio}}$, etc. Chaque entité $\mathcal{E}_k$ appartient à l'un de ces espaces. Si $\mathrm{type}(\mathcal{E}_k) = \mathrm{Vision}$, alors $\mathcal{E}_k$ est un **embedding** visuel $\mathbf{x}_k \in \mathcal{X}_{\text{vision}}$, etc.

Pour deux nœuds $\mathcal{E}_i$ et $\mathcal{E}_j$, il convient de distinguer **plusieurs** fonctions de synergie :

$$S_{m_1,m_2}: \quad \mathcal{X}_{m_1} \times \mathcal{X}_{m_2} \ \rightarrow\ \mathbb{R},$$

où $m_1 = \mathrm{type}(\mathcal{E}_i)$ et $m_2 = \mathrm{type}(\mathcal{E}_j)$. Autrement dit, si $\mathcal{E}_i$ et $\mathcal{E}_j$ sont tous deux des images (vision–vision), on utilise $S_{\text{vision,vision}}$. Si c'est un segment audio $\mathcal{E}_i$ et une entité textuelle $\mathcal{E}_j$, on appelle $S_{\text{audio,texte}}$, etc.

De cette manière, on ne confond pas la **mesure** de ressemblance audio–audio (spectre, MFCC) avec celle de ressemblance texte–texte (similarité cosinus sur embeddings linguistiques). Le **module** Synergie, lors de la mise à jour $\omega_{i,j}$, identifie automatiquement $m_1 = \mathrm{type}(i)$ et $m_2 = \mathrm{type}(j)$, puis appelle S_{m_1,m_2} avec les représentations $\mathbf{x}_i, \mathbf{y}_j$ appropriées.

B. Illustration : Exemples de Fonctions S

Si $\mathcal{E}_i$ et $\mathcal{E}_j$ sont deux **images** (même modalité "Vision"), on peut définir

$$S_{\text{vision,vision}}(\mathbf{v}_i, \mathbf{v}_j) \;=\; \cos(\mathbf{v}_i, \mathbf{v}_j) \;=\; \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\| \mathbf{v}_i \| \, \| \mathbf{v}_j \|},$$

où $\mathbf{v}_i \in \mathbb{R}^{d_{\text{img}}}$ et $\mathbf{v}_j \in \mathbb{R}^{d_{\text{img}}}$ sont des embeddings CNN. On peut aussi recourir à un RBF kernel, ou à une distance euclidienne inversée.

Lorsque $\text{type}(\mathcal{E}_i) = \text{Vision}$ et $\text{type}(\mathcal{E}_j) = \text{Texte}$, il arrive qu'on dispose d'un **espace latent commun** (ex. CLIP) où $\mathbf{v}_i \in \mathbb{R}^d$ (projection de l'image) et $\mathbf{t}_j \in \mathbb{R}^d$ (projection du texte). On définit alors :

$$S_{\text{vision,texte}}(\mathbf{v}_i, \mathbf{t}_j) \;=\; \cos(\mathbf{v}_i, \mathbf{t}_j).$$

Cette fonction renvoie un **score** de correspondance cross-modal image–texte.

De la même manière, on peut configurer :

- $S_{\text{audio,vision}}$: modèle la **corrélation** ou la **coïncidence** spatio-temporelle entre un signal audio et un flux vidéo (cf. lip-sync, ambiance sonore).
- $S_{\text{audio,texte}}$: compare un extrait audio (caractéristiques phonétiques) à une transcription textuelle (caractéristiques lexicales ou phonémiques).

C. Mise en Œuvre dans la Règle DSL

Au cours de la **mise à jour** DSL, la pondération $\omega_{i,j}(t+1)$ est calculée par :

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta \, \left[S(\mathcal{E}_i, \mathcal{E}_j) - \tau \, \omega_{i,j}(t) \right].$$

Le **module** Synergie se charge de renvoyer la valeur de $S(\cdot,\cdot)$ en examinant :

$$\text{type}(\mathcal{E}_i) = m_1, \quad \text{type}(\mathcal{E}_j) = m_2,$$

puis appelle :

$$S(\mathcal{E}_i, \mathcal{E}_j) \;=\; S_{m_1,m_2}(\mathbf{x}_i, \; \mathbf{y}_j).$$

Où $\mathbf{x}_i$ et $\mathbf{y}_j$ sont les **représentations** (vecteurs, frames, embeddings) de $\mathcal{E}_i$ et $\mathcal{E}_j$. Le reste du **DSL** ne se soucie pas de la modalité. Il applique la règle d'apprentissage de ω comme si $S(\cdot,\cdot)$ était unique.

La **clarté** du système permet de différencier efficacement vision–vision, qui regroupe des images similaires, de vision–texte, qui associe une image à sa légende. L'**évolutivité** est assurée, car l'ajout d'une **nouvelle** modalité, telle qu'un capteur ou un EEG, ne nécessite que la définition des fonctions de similarité $S_{\text{nouvelle, vision}}$ et $S_{\text{nouvelle, texte}}$, sans qu'il soit nécessaire de modifier le cœur du **DSL**.

D. Exemples d'Alignement Multi-Type

Image–Image : Permet un **clustering** d'images proches (ex. toutes les photos de paysages).

Texte–Texte : Regroupe des documents ou tokens sémantiquement similaires, formant des thèmes.

Audio–Audio : Classe des segments sonores (voix semblables, bruitages similaires).

Vision–Texte : Associe un objet visuel à sa description, facilitant l'annotation ou la recherche inversée (requête textuelle → images).

Audio–Texte : Aligne un flux vocal avec sa transcription ou des mots clés (reconnaissance).

Vision–Audio : Détecte la correspondance (lip sync, ambiance sonore corrélée aux actions vidéo).

8.6.1.3. Exemples de définitions multiples de S selon les paires de modalités

Dans un **DSL** (Deep Synergy Learning) véritablement multimodal, chaque entité $\mathcal{E}_i$ peut appartenir à l'une de plusieurs **modalités**. Images, textes, sons, données de capteurs, etc. Pour que le **SCN** (Synergistic Connection Network) sache relier ces entités de manière significative, il est indispensable de **définir** des **fonctions** de synergie $S(\mathcal{E}_i, \mathcal{E}_j)$ adaptées à chaque **couple** de modalités. Ainsi, la comparaison image ↔ image ne se fera pas de la même façon que image ↔ texte ou audio ↔ texte. La présente section illustre comment on peut **multiplier** les définitions de S en fonction de ces paires de modalités, tout en conservant la **même** dynamique DSL sur la matrice de liaisons $\omega_{i,j}$.

*A. **Synergie Image–Image** : mesures de similarité visuelle*

Lorsqu'on compare deux **images** (ou frames vidéo), on peut représenter chaque image $\mathcal{I}_i$ par un vecteur $\mathbf{v}_i \in \mathbb{R}^d$. Ce vecteur peut provenir :

- d'un **embedding** d'un réseau de neurones (CNN, ViT),
- de **descripteurs** locaux (SIFT, SURF) regroupés ou agrégés,
- d'un histogramme global (couleurs, textures).

Ensuite, la **synergie** $S_{\text{img,img}}$ se formule souvent via une **similarité cosinus**,

$$S_{\text{img,img}}(\mathcal{I}_i, \mathcal{I}_j) \;=\; \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\| \mathbf{v}_i \| \, \| \mathbf{v}_j \|},$$

ou bien via une **distance euclidienne** inversée ou un **RBF** (radial basis function),

$$S_{\text{img,img}}(\mathcal{I}_i, \mathcal{I}_j) \;=\; \exp\!\left(-\alpha \, \| \mathbf{v}_i - \mathbf{v}_j \|^2\right).$$

Un score élevé indique des images jugées **proches** (même objet, même catégorie).

Dans des cas plus fins, on peut calculer des correspondances **locales** (patchs, keypoints) et tirer un score global de matching, comme :

$$S_{\text{img,img}}(\mathcal{I}_i, \mathcal{I}_j) \;=\; \frac{\left|\mathcal{M}(\mathbf{k}_i, \mathbf{k}_j)\right|}{\max\left(|\mathbf{k}_i|, |\mathbf{k}_j|\right)},$$

où $\mathbf{k}_i$ et $\mathbf{k}_j$ désignent des ensembles de points-clés (SIFT) et $\mathcal{M}$ représente l'ensemble des appariements trouvés. Cela donne une mesure plus robuste aux changements d'échelle, rotations, etc.

B. ***Synergie Texte–Texte*** *: similarité sémantique ou distributionnelle*

Chaque **texte** (ou segment, paragraphe, document) $\mathcal{T}_i$ peut être converti en un vecteur $\mathbf{w}_i \in \mathbb{R}^m$ via un modèle de langage (Word2Vec, GloVe, BERT…). La **synergie** s'écrit alors :

$$S_{\text{text,text}}(\mathcal{T}_i, \mathcal{T}_j) \;=\; \cos(\mathbf{w}_i, \mathbf{w}_j),$$

pour refléter la **proximité sémantique**. On peut également recourir à un **kernel** exponentiel sur la distance $\| \mathbf{w}_i - \mathbf{w}_j \|$, etc.

De façon plus "classique", on peut considérer un **vector** TF–IDF $\mathbf{tfidf}_i$ pour le texte $\mathcal{T}_i$. Alors, la similarité cosinus sur ces vecteurs

$$S_{\text{text,text}}(\mathcal{T}_i, \mathcal{T}_j) \;=\; \frac{\mathbf{tfidf}_i \cdot \mathbf{tfidf}_j}{\| \mathbf{tfidf}_i \| \, \| \mathbf{tfidf}_j \|}$$

indique si les textes partagent les mêmes mots, avec pondération de leur fréquence et de leur rareté. On peut aussi définir un coefficient de Jaccard (chevauchement de tokens).

C. ***Synergie Audio–Audio*** *: correspondances de signatures acoustiques*

Pour deux segments audio $\mathcal{A}_i$, $\mathcal{A}_j$, on extrait des **features** (ex. MFCC, spectrogrammes résumés) sous forme de vecteurs $\mathbf{z}_i, \mathbf{z}_j$. On applique une **distance** ou une **similarité** :

$$S_{\text{aud,aud}}(\mathcal{A}_i, \mathcal{A}_j) \;=\; \cos(\mathbf{z}_i, \mathbf{z}_j) \quad \text{ou} \quad \exp(-\| \mathbf{z}_i - \mathbf{z}_j \|^2).$$

Des segments audio comparables (même locuteur, même bruit, même instrument) obtiennent un score $S_{\text{aud,aud}}$ élevé. Cela permet de **regrouper** (clustering) ou de **détecter** des répétitions sonores.

D. ***Synergie Inter-Modale*** *: image–texte, audio–texte, image–audio, etc.*

Lorsque $\mathcal{E}_i$ est une image, $\mathcal{E}_j$ un segment textuel, on définit souvent un **espace latent** commun (modèles de type CLIP) ou un simple **réseau** de correspondance :

$$S_{\text{img,txt}}(\mathcal{I}, \mathcal{T}) \;=\; \cos\left(\Phi_{\text{img}}(\mathbf{v}_i),\; \Phi_{\text{txt}}(\mathbf{w}_j)\right).$$

Cela reflète la **compatibilité sémantique** entre l'**embedding** visuel et l'**embedding** textuel.

Pour aligner un **segment audio** avec un **texte** (transcription, mot-clef), on peut projeter $\mathbf{a}_i$ (vecteur phonétique) et $\mathbf{t}_j$ (embedding lexical) dans un espace cross-modal, ou recourir à la **transcription** auto-supervisée (ASR). On obtient un score

$$S_{\text{aud,txt}}(\mathbf{a}_i, \mathbf{t}_j) \;=\; \rho\left(\Phi_{\text{audio}}(\mathbf{a}_i),\, \Phi_{\text{txt}}(\mathbf{t}_j)\right),$$

qui indique la **corrélation** entre la voix enregistrée et le contenu textuel présumé.

Plus rare, cette comparaison peut survenir dans un contexte vidéo (lip sync, bruits associés à des actions visuelles). On peut soit définir directement un **embedding** image–audio (modèle cross-modal), soit passer par un "hub" textuel (ex. l'audio transcrit en mots, l'image décrite par

tags), puis comparer les mots. Le DSL autorise cependant un lien direct $\omega_{(\text{img}),(\text{audio})}$ dès lors qu'on dispose d'une **fonction** $S_{\text{img,aud}}$.

*E. **Synthèse : une Famille de S pour chaque paire de Modalités***

Sur un plan purement **mathématique**, la synergie $S(\mathcal{E}_i, \mathcal{E}_j)$ n'est pas une unique fonction universelle, mais une **famille** de fonctions :

$$S_{(\text{modality}_i, \text{modality}_j)},$$

adaptées à la modalité (ou au type) de $\mathcal{E}_i$ et $\mathcal{E}_j$. Dans un **DSL multimodal**, le **module Synergie** (chap. 5) joue un rôle central dans l'évaluation des relations entre entités. Il **identifie** les types $\text{type}(\mathcal{E}_i)$ et $\text{type}(\mathcal{E}_j)$ afin de sélectionner la fonction de similarité appropriée. Ensuite, il **appelle** la fonction $S_{(\text{mod}(i), \text{mod}(j))}(\mathbf{x}_i, \mathbf{y}_j)$ correspondant aux modalités des deux entités concernées. Enfin, il **retourne** un score réel, intégrant si nécessaire un offset ou une fenêtre temporelle pour capturer la synchronisation entre signaux multimodaux.

Ce score est alors inséré dans la **mise à jour** :

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\,\big[S(\mathcal{E}_i, \mathcal{E}_j) - \tau\,\omega_{i,j}(t)\big].$$

De cette façon, un **SCN** unique gère la **même** matrice ω, tout en recourant à différentes **formules** de similarité/distances selon la nature des entités.

8.6.2. Densité et Parcimonie

Dans un contexte **multimodal**, où l'on fusionne potentiellement plusieurs ensembles d'entités (images, textes, sons…), le **SCN** (Synergistic Connection Network) peut se retrouver envahi par un **trop grand** nombre de liaisons. La synergie $S(\mathcal{E}_i, \mathcal{E}_j)$ calculée entre deux entités multimodales (ex. image vs. texte) n'est pas toujours nulle, et l'on risque de multiplier les liens $\omega_{i,j}$ dès lors qu'il existe un soupçon de corrélation. D'où l'importance de **contrôler** la densité du réseau et de préserver une forme de **parcimonie** qui maintient la cohérence et la scalabilité.

8.6.2.1. Risque de prolifération de liens si toutes les images se lient à tous les textes

Dans un **SCN** (Synergistic Connection Network) dédié à la fusion multimodale, il est naturel de vouloir relier des **images** à des **textes**, par exemple pour établir des correspondances entre le contenu visuel et des descriptions ou des mots-clés. Cependant, si l'on se contente de calculer une **synergie** $S(i,j)$ pour **toutes** les paires $(i \in \{\text{images}\}, j \in \{\text{textes}\})$ sans appliquer de mécanisme de filtrage ou de sélection, on peut rapidement faire face à une **explosion** du nombre de liens $\omega_{i,j}$. Cette situation nuit autant à la **lisibilité** du réseau qu'à son **efficacité** computationnelle et à sa capacité à faire émerger des **clusters** clairement distincts.

A. Explosion du nombre de liaisons et coût $O(N_{img}\, N_{txt})$

Considérons un ensemble de N_{img} images et un ensemble de N_{txt} textes. Dans un **SCN** où chaque image img_i est reliée à chaque texte txt_j via une pondération $\omega_{i,j}$, on obtient au total $O(N_{\text{img}} \times N_{\text{txt}})$ liaisons potentielles.

Sur le plan **mathématique**, la mise à jour du SCN (voir chapitre 4) impose, à chaque itération, de calculer la synergie

$$S\big(\mathrm{img}_i, \mathrm{txt}_j\big)$$

puis de mettre à jour la pondération

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\big[S\big(\mathrm{img}_i, \mathrm{txt}_j\big) \;-\; \tau\,\omega_{i,j}(t)\big].$$

Lorsque N_{img} et N_{txt} deviennent significatifs (des milliers, voire plus), cette itération globale requiert un nombre de calculs de l'ordre de $N_{\mathrm{img}} \times N_{\mathrm{txt}}$. Le **coût** peut alors s'avérer prohibitif et ralentir fortement la convergence du réseau.

B. Risque de densification et perte de discrimination

Lorsqu'une **image** se relie à un trop grand nombre de **textes**, ou qu'un **texte** se retrouve faiblement corrélé avec de nombreuses images, le SCN tend à devenir **très dense**. Cela engendre deux problèmes majeurs :

Manque de sélectivité des liaisons. Un lien $\omega_{i,j}$ qui n'est que moyennement élevé (et qui ne redescend jamais à un niveau quasi nul) peut persister artificiellement. On se retrouve alors avec un grand nombre de liaisons "moyennes" ou "intermédiaires", sans que le réseau n'affiche un réel contraste entre liens "très pertinents" et liens "faiblement pertinents".

Difficulté à faire émerger des clusters nets. Sur un plan **conceptuel**, on aimerait voir des **clusters** "Images de chats + mots-clés félins", ou "Images de paysages + vocables sur la nature". Si l'immense majorité des textes entretient un degré de similarité (même léger) avec la plupart des images, la structure du réseau devient confuse, car rien ne s'oppose à ce que liaisons "moyennes" se multiplient. Les macro-nœuds deviennent moins identifiables, le SCN ressemblant plus à un vaste nuage de liaisons qu'à une organisation claire.

C. Perte de lisibilité et charge mémoire

D'un **point de vue pratique**, stocker $\omega_{i,j}$ pour $\mathrm{O}\big(N_{\mathrm{img}} \times N_{\mathrm{txt}}\big)$ liens peut occuper une **grande quantité** de mémoire si les matrices de pondérations sont denses, en particulier quand N_{img} et N_{txt} se chiffrent en dizaines ou centaines de milliers.

Par ailleurs, la **visualisation** ou l'**inspection** du réseau devient ardue. Un SCN trop dense rend difficile la lecture de la structure. Même si, théoriquement, la mise à jour DSL pourrait faire baisser certains $\omega_{i,j}$, la dynamique risque de prendre beaucoup d'itérations pour "nettoyer" efficacement les liaisons moyennes ou faibles.

D. Perte d'efficacité d'auto-organisation

Le **DSL** (chap. 4) et ses mécanismes annexes (recuit, inhibition, etc.) ont précisément pour vocation de faire **émerger** des **clusters** (sous-groupes d'entités fortement connectées), tout en abaissant la plupart des pondérations hors cluster. Si, dès le départ, le réseau se retrouve avec trop de liaisons, le **temps** nécessaire pour que ces liaisons s'affaiblissent peut s'allonger considérablement.

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\big[S(i,j) \;-\; \tau\,\omega_{i,j}(t)\big].$$

Il faut de nombreuses itérations pour faire **redescendre** $\omega_{i,j}$ à un niveau quasi nul quand le réseau est $\mathrm{O}\big(N_{\mathrm{img}}\,N_{\mathrm{txt}}\big)$.

8.6.2.2. Heuristiques : k plus proches voisins multimodaux, ϵ-rayon, ou recuit (Chap. 7.3)

Lorsqu'un **DSL** (Deep Synergy Learning) traite simultanément plusieurs **modalités** (vision, audio, texte, capteurs, etc.), le risque de prolifération de liaisons $\omega_{i,j}$ devient particulièrement aigu. En effet, si le SCN tente d'évaluer la **synergie** $S(i,j)$ pour **toutes** les paires (i,j) issues d'un large ensemble d'entités multimodales, la complexité peut s'élever en $O(n^2)$. Au-delà du coût de calcul, une telle densité compromet la *lisibilité* des clusters et la *qualité* de l'auto-organisation. Diverses **heuristiques** se révèlent alors essentielles pour limiter le **nombre** de liens effectivement considérés ou pour maintenir la dynamique DSL hors de minima locaux mal configurés. Parmi ces heuristiques, on trouve notamment :

- Le **k plus proches voisins** (k-NN) multimodal,
- Le ϵ-**rayon** (ou ϵ-*ball*) limitant les connexions à un certain périmètre de distance,
- Le **recuit simulé** (réexpliqué au *Chap. 7.3* sur les méthodes d'optimisation stochastiques).

A. k plus proches voisins multimodaux

Le **k-NN** (k plus proches voisins) propose de **restreindre** la création (ou la mise à jour) de liaisons $\omega_{i,j}$ aux seuls "voisins les plus proches" selon une certaine **métrique** multimodale. Concrètement, pour chaque entité $\mathcal{E}_i$ (qu'elle soit image, texte ou segment audio), on ne retient que les k entités $\mathcal{E}_j$ minimisant une distance $d(\mathcal{E}_i, \mathcal{E}_j)$. Les liaisons hors de ce petit voisinage sont fixées à 0 ou négligées.

Le **gain** est double. D'une part, la complexité de calcul est réduite de $O(n^2)$ à $O(n\,k)$, chaque entité ne maintenant un lien qu'avec k voisins, ce qui optimise les ressources de traitement. D'autre part, le réseau devient **plus sparse**, avec une densité de liaisons sensiblement diminuée, ce qui facilite la **mise à jour** DSL et améliore la **lisibilité** des clusters, permettant une structuration plus claire des connexions.

L'approche **multimodale** suppose qu'une entité $\mathcal{E}_i$ puisse porter des composantes $\mathbf{x}_i^{(\text{vision})}$, $\mathbf{x}_i^{(\text{texte})}$, $\mathbf{x}_i^{(\text{audio})}$, etc. On définit alors une **distance** globale

$$d(\mathcal{E}_i, \mathcal{E}_j) \;=\; \alpha_1\, d_{\text{vision}}\left(\mathbf{x}_i^{(\text{vis})}, \mathbf{x}_j^{(\text{vis})}\right) \;+\; \alpha_2\, d_{\text{texte}}\left(\mathbf{x}_i^{(\text{txt})}, \mathbf{x}_j^{(\text{txt})}\right) \;+ \cdots$$

où $\alpha_1, \alpha_2, \ldots$ pondèrent l'importance relative de chaque modalité. Une fois cette distance définie, pour chaque $\mathcal{E}_i$ on recherche les k entités $\mathcal{E}_j$ les "moins distantes", et on autorise uniquement $\omega_{i,j}$ pour ces paires.

La **sparséfication** induite par le k-NN **soulage** fortement le SCN. La mise à jour $\omega_{i,j}$ ne s'effectue plus sur l'entièreté des couples (i,j), mais seulement sur $O(n\,k)$. De plus, le réseau évite de se noyer dans des liaisons moyennes.

Toutefois, la **pertinence** de k-NN dépend de la **cohérence** de la distance d. Si celle-ci n'est pas adaptée (mauvaises projections, déséquilibres entre modalités), on peut *oublier* des liens synergiques rares mais importants. D'où la nécessité de paramétrer soigneusement $\alpha_1, \alpha_2, \ldots$ et d'ajuster la dimension des embeddings.

B. ϵ-rayon

Une alternative au k-NN consiste à définir, pour chaque entité $\mathcal{E}_i$, un **voisinage** $\mathrm{N}_\epsilon(i)$ composé des entités $\mathcal{E}_j$ satisfaisant

$$d(\mathcal{E}_i, \mathcal{E}_j) \;\leq\; \epsilon,$$

où $\epsilon > 0$ est un **paramètre** fixant le "rayon" de connectivité. Seules les paires (i, j) pour lesquelles $d(\mathcal{E}_i, \mathcal{E}_j) \leq \epsilon$ sont susceptibles de voir leur liaison $\omega_{i,j}$ mise à jour et s'élever. Au contraire, si la distance est trop grande, on met $\omega_{i,j} = 0$.

Les **avantages** de cette approche sont multiples. La **sélectivité** de l'entourage permet de limiter les connexions aux entités réellement **proches** dans l'espace multimodal, ce qui évite de calculer la synergie pour des paires trop éloignées, à condition que le paramètre ϵ soit bien calibré. L'**interprétation géométrique** offre une visualisation intuitive du réseau. Chaque entité, représentée dans un espace vectoriel après projection, définit une **boule** de rayon ϵ, et seules les entités situées à l'intérieur de cette hypersphère sont considérées comme des voisines potentielles. Enfin, le **pilotage de la densité** du graphe est directement influencé par la valeur de ϵ. Un rayon plus grand favorise un réseau plus dense, tandis qu'un ϵ plus restreint renforce la **sparséfication**, optimisant ainsi l'**auto-organisation** du DSL en régulant le nombre de connexions actives.

Avec un rayon ϵ, le **nombre** de voisins peut varier d'une entité à l'autre en fonction de la distribution des points. k-NN, au contraire, fixe un **nombre** k identique pour chaque entité, ce qui peut entraîner un déséquilibre.

Une entité située dans une zone de forte densité risque d'accumuler plusieurs voisins très proches, tandis qu'une autre en zone de faible densité pourrait en avoir très peu. Pour concilier ces méthodes, on peut combiner les approches en utilisant un rayon ϵ afin d'éviter les distances excessives, tout en imposant une limite sur le nombre de voisins afin de prévenir une explosion de connexions dans les régions les plus denses.

C. Recuit (Chap. 7.3) pour surmonter les minima locaux

Le **recuit simulé** consiste à **injecter** un **bruit** stochastique dans la mise à jour $\omega_{i,j}$. À chaque itération t, on ajoute par exemple un terme $\xi_{i,j}(t)$ de variance contrôlée par une "température" $T(t)$. La mise à jour prend la forme

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\big[S(i,j) \;-\; \tau\,\omega_{i,j}(t)\big] \;+\; \sigma(t)\,\xi_{i,j}(t).$$

où $\sigma(t) = T(t)$ décroît au fil du temps (refroidissement).

Même en utilisant un voisinage restreint (k-NN, ϵ-rayon), le **DSL** peut se retrouver bloqué dans un **minimum local** où certains liens $\omega_{i,j}$ se stabilisent alors qu'il existerait une configuration plus globale, plus cohérente, si l'on acceptait de "briser" quelques associations établies au profit d'autres. L'injection de bruit incite le système à **explorer** des configurations "moins immédiates" et à franchir des barrières locales.

Il est tout à fait possible de limiter la dynamique aux paires (i, j) tombant dans le rayon ϵ (ou dans les k plus proches voisins) *et* d'appliquer un recuit sur ces paires sélectionnées. Cela réduit considérablement le **nombre** de liaisons actives, tout en permettant à la structure d'évoluer par perturbations aléatoires quand la synergie reste dans une zone ambivalente.

8.6.2.3. Surveiller la scalabilité

Dans un **DSL** (Deep Synergy Learning) orienté multimodal, la question de la **scalabilité** se révèle déterminante dès lors que le nombre n d'entités — qu'il s'agisse d'images, de segments audio, de textes ou d'autres flux — devient important. Chaque entité $\mathcal{E}_i$ est susceptible de se lier à de nombreuses autres entités $\mathcal{E}_j$, et la **logique** du SCN (Synergistic Connection Network) peut entraîner un nombre de liaisons $\omega_{i,j}$ potentiellement en $O(n^2)$. Sans mesures de régulation, les ressources de calcul et de stockage sont rapidement saturées, rendant l'auto-organisation du DSL non viable à grande échelle. La présente section **8.6.2.3** met en relief l'importance de **surveiller** et de **maîtriser** la complexité dès lors que plusieurs modalités coexistent, et expose quelques indicateurs et approches pour y parvenir.

A. Complexité Algorithmique et Nombre de Liaisons

En théorie, si l'on cherche à évaluer la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ pour *toutes* les paires (i, j) d'entités, on effectue $O(n^2)$ comparaisons. Dans un système multimodal, un même objet (par ex. une image) peut se comparer non seulement à d'autres images, mais aussi à tous les segments audio et textuels, multipliant les synergies à calculer. On se retrouve ainsi avec :

$$\text{nombre total de paires} \approx N_{\text{img}}^2 + N_{\text{txt}}^2 + N_{\text{aud}}^2 + 2\,N_{\text{img}}N_{\text{txt}} + 2\,N_{\text{img}}N_{\text{aud}} + 2\,N_{\text{txt}}N_{\text{aud}}\,,$$

ce qui peut grandement dépasser la capacité d'un simple algorithme en $O(n^2)$ si les flux sont massifs.

En l'absence de mécanismes de **sparsification**, la dynamique DSL peut maintenir un volume **conséquent** de liaisons $\omega_{i,j}$ d'intensité moyenne, ne décroissant pas suffisamment vite. Cela nuit à la **lisibilité**, car un réseau ultra-dense ne met plus en évidence de *clusters* nets, une même entité se connectant à un très grand nombre d'autres. De plus, la **charge de mise à jour** augmente, puisque maintenir et actualiser chaque $\omega_{i,j}$ requiert un temps de calcul important à chaque itération, ce qui ralentit la convergence du système.

Un **DSL** multimodal est souvent **dynamique**, ce qui signifie que de nouveaux objets tels que des images, des textes ou des extraits audio peuvent arriver en flux continu. Si, à chaque insertion, on compare le nouvel objet à *toutes* les entités existantes, la charge croît de façon quasi-cubique en régime continu $O(n)$ vérifications à chaque arrivée, répétées plusieurs fois). La complexité peut rapidement **s'emballer**, rendant nécessaire une surveillance attentive du **temps** moyen d'une itération, du **nombre** de liaisons actives $\omega_{i,j}$, ainsi que de la **densité** du graphe et de la **taille** totale de la matrice ω.

B. Stratégies pour Gérer la Scalabilité

Comme développé en 8.6.2.2, il est possible de restreindre les connexions à un **voisinage** calculé a priori. Le **k-NN multimodal** consiste à ne conserver, pour chaque entité $\mathcal{E}_i$, que ses k plus proches voisins selon une distance multimodale. Une autre approche repose sur le ϵ**-rayon**, où la connectivité est limitée aux entités $\mathcal{E}_j$ satisfaisant la condition

$$d(\mathcal{E}_i, \mathcal{E}_j) \leq \epsilon.$$

Ces approches ont l'avantage de convertir un possible $O(n^2)$ en $O(n\,k)$ ou $O(n \log n)$ si l'on emploie des structures d'indexation (k-d trees, etc.). Sur le **plan mathématique**, cette

sparséfication entraîne un graphe plus léger, tout en préservant la dynamique DSL dans le voisinage local.

Lorsqu'il devient impossible de gérer un grand $\{\omega_{i,j}\}$ sur une seule machine, il est possible de **distribuer** la matrice ou la collection d'entités sur plusieurs nœuds HPC (High Performance Computing). Une première approche consiste à diviser l'ensemble $\{1, \dots, n\}$ en **sous-SCN**, chaque bloc étant traité indépendamment par un sous-réseau (voir Chapitre 5.7 sur la distribution). Ces sous-SCN effectuent ensuite des **synchronisations périodiques**, où les pondérations inter-blocs sont échangées à intervalles réguliers pour assurer la cohérence globale. Enfin, un **équilibrage** peut être nécessaire. Si un bloc devient trop lourd, soit en raison d'un trop grand nombre d'entités, soit par accumulation de connexions, il est scindé ou redistribué pour optimiser la charge de calcul.

Cette approche est très efficace si l'on structure les entités en clusters "relativement indépendants", faisant que peu de liaisons inter-blocs doivent être maintenues.

Pour un **flux continu** d'entités, une **stratégie** incrémentale peut être adoptée. À l'arrivée d'une nouvelle entité $\mathcal{E}_{\text{new}}$, on ne compare celle-ci qu'à un **sous-ensemble** (k plus proches voisins, un échantillon aléatoire, un bloc HPC spécifique). Puis la dynamique DSL localise $\mathcal{E}_{\text{new}}$ dans le graphe déjà existant sans relancer une phase exhaustive. Cette **sélectivité** évite des réinitialisations coûteuses et étend la **scalabilité** à des environnements en flux.

C. Indicateurs de Survie : Densité et Temps d'Itération

Pour "**surveiller**" la scalabilité, on peut définir quelques **indicateurs** clés au fil des itérations :

Densité Effective δ. On calcule

$$\delta(t) = \frac{1}{n(n-1)} \sum_{i \neq j} \mathbf{1}\{\omega_{i,j}(t) \neq 0\},$$

c'est-à-dire la proportion de liaisons non nulles. Si $\delta(t)$ reste, disons, en dessous de 1 % (ou 5 %, etc.), le réseau est très clairsemé, ce qui est souhaitable pour un traitement large-échelle.

Temps par Époque / Itération. À chaque "tour" de mise à jour, on note le **temps** d'exécution. Si ce temps grandit trop vite avec n, on sait que l'approche n'est pas scalable.

Qualité du Clustering ou de la Représentation. On peut aussi mesurer la **modularité** ou l'**indice d'homogénéité** des clusters. Si une sparséfication trop sévère ruine la qualité, on fait un compromis en augmentant k ou ϵ.

D. Exemples Concrets de Contrôle

Dans un **SCN** multimodal, plusieurs mécanismes permettent de contrôler la structure du graphe et d'optimiser son évolution.

Un premier exemple repose sur un **ajustement dynamique** de k ou ϵ . Au début du processus, on fixe des valeurs relativement larges afin de permettre la formation de connexions pertinentes. Puis, à mesure que le réseau atteint un état plus stable, on réduit progressivement k ou ϵ pour favoriser la sparséfication et ne conserver que les liens les plus significatifs.

Une seconde approche combine **recuit simulé et inhibition**. La température du recuit facilite l'exploration de nouvelles configurations, tandis que l'inhibition régule la compétition entre

liaisons. On surveille la densité δ et, si celle-ci devient trop élevée, on intensifie l'inhibition ou on accélère la baisse de température pour éviter un surchargement du réseau.

Une autre solution consiste à organiser le système en **sous-SCN spécialisés** par modalité ou par blocs thématiques. Par exemple, un SCN peut être dédié aux entités visuelles, un autre aux entités textuelles, un troisième à l'audio, et un méta-SCN se charge de relier ces différents sous-ensembles via des super-nœuds allégés. Cette structuration réduit la charge de calcul tout en préservant la connectivité globale du réseau.

8.6.3. Synergie Tri-Modal (Vision–Langage–Audio)

Dans les systèmes **multimodaux**, il arrive souvent que l'on souhaite exploiter **trois flux** simultanément. Une **image** (ou une séquence vidéo), un **segment audio** (enregistrement sonore, musique, paroles) et un **texte** (sous-titres, transcription, métadonnées). Le **DSL** (Deep Synergy Learning) peut alors dépasser la simple synergie *binaire* (image–texte, texte–audio, ou audio–image) pour calculer un **score "triple"** lorsqu'il existe une convergence entre les **trois modalités**. Cette synergie tri-modale permet d'identifier des **coïncidences** plus riches (ex. un passage vidéo, une phrase correspondante, et un signal audio concordant) et de renforcer la **cohérence** du réseau face à des phénomènes multimédias complexes (ex. un film sous-titré, un concert enregistré avec commentaire textuel, etc.).

8.6.3.1. Possibilité de calculer un "score triple" si un segment audio, un embedding image et un texte concordent

Lorsqu'un **DSL** (Deep Synergy Learning) vise à gérer **plus de deux** modalités simultanément, par exemple **audio**, **image** et **texte**, il peut être pertinent de dépasser la seule logique binaire $S(i,j)$ pour définir, dans certains cas, un **score de synergie à trois**. On introduit alors $S^{(3)}(\mathcal{E}_a, \mathcal{E}_v, \mathcal{E}_t)$ qui quantifie la **concordance** entre un segment audio, un extrait visuel et un passage textuel.

Cette possibilité ouvre la voie à une **coopération** tri-modale permettant de repérer — dans le même laps de temps ou la même scène — l'**harmonie** entre ces trois sources (par exemple un objet filmé + son bruit spécifique + la légende textuelle décrivant l'objet).

A. Formalisation Mathématique d'un Score Triple

Dans le **DSL** traditionnel, on traite la synergie **par paires** $S(i,j)$. Pour un système tri-modal, on pourrait commencer par combiner les **scores binaires** déjà définis entre paires :

$$S^{(2)}(\text{audio}, \text{vision}), \quad S^{(2)}(\text{audio}, \text{texte}), \quad S^{(2)}(\text{vision}, \text{texte}).$$

Une **fonction** F prend alors ces trois valeurs en entrée et en déduit un **score** global :

$$S^{(3)}(a,v,t) \;=\; F\left(S^{(2)}(a,v),\; S^{(2)}(a,t),\; S^{(2)}(v,t)\right).$$

On peut imaginer plusieurs formes de F. Par exemple :

Additive :

$$S^{(3)}(a,v,t) = \alpha_1 S^{(2)}(a,v) \alpha_2 S^{(2)}(a,t) \alpha_3 S^{(2)}(v,t)$$

où $\alpha_1, \alpha_2, \alpha_3$ sont des poids.

Multiplicative :

$$S^{(3)}(a,v,t) = \left[S^{(2)}(a,v)\right] \times \left[S^{(2)}(a,t)\right] \times \left[S^{(2)}(v,t)\right],$$

un score élevé n'apparaissant que si **toutes** les synergies binaires sont elles-mêmes fortes.

Min ou **Médiane** :

$$S^{(3)}(a,v,t) = \min\left\{S^{(2)}(a,v),\, S^{(2)}(a,t),\, S^{(2)}(v,t)\right\} \text{ ou } \text{median}\{\dots\},$$

traduisant la contrainte qu'il faut un "socle commun" de toutes les paires pour afficher une bonne synergie tri-modale.

Cette **combinaison** binaire—>tri-modale s'avère assez **simple** à mettre en œuvre, car on suppose déjà avoir calculé $S^{(2)}$ entre chaque paires de modalités. Toutefois, elle peut rater certaines nuances si la "tri-synergie" ne se résume pas au simple produit ou à la simple somme des synergies binaires (certains signaux n'apparaissent qu'en conjonction).

Une autre approche, plus proche de la **théorie de l'information**, considère que l'on dispose de variables aléatoires $X_{\text{audio}}, X_{\text{image}}, X_{\text{texte}}$. La **co-information** à trois variables, inspirée du formalisme de l'information mutuelle, se définit en combinant entropies et entropies conjointes :

$$\begin{aligned}&\text{CoInfo}\left(X_{\text{aud}},\, X_{\text{img}},\, X_{\text{txt}}\right)\\ &= H(X_{\text{aud}}) + H\left(X_{\text{img}}\right) + H(X_{\text{txt}})\\ &- \left[H\left(X_{\text{aud}}, X_{\text{img}}\right)H(X_{\text{aud}}, X_{\text{txt}})H\left(X_{\text{img}}, X_{\text{txt}}\right)\right] + H\left(X_{\text{aud}}, X_{\text{img}}, X_{\text{txt}}\right)\end{aligned}$$

où H désigne l'entropie (ou entropie conjointe) selon la définition de la théorie de l'information.

- Si $\text{CoInfo} > 0$, on considère qu'il existe une **redondance** partagée par les trois variables ; si elle est négative, on parle d'**interaction synergique** plus complexe.
- D'un point de vue **DSL**, on peut alors définir un "score triple" $S^{(3)}$ proportionnel à $\max\{0, \text{CoInfo}\}$ ou toute autre dérivation, afin d'implémenter un mécanisme où la synergie tri-modale est élevée lorsque l'audio, l'image et le texte s'avèrent très fortement corrélés.

Une fois un "**score triple**" $S^{(3)}(a,v,t)$ établi, on peut imaginer étendre la **dynamique** DSL à des **hyper-liens** (voir plus loin dans l'ouvrage, ou Chap. 12 si l'on traite la synergie n-aire). Par exemple, on ne manipule plus seulement des pondérations $\omega_{i,j}$ pour deux entités, mais aussi des coefficients $\omega_{a,v,t}$ pour un triplet (a,v,t). La mise à jour :

$$\omega_{(a,v,t)}(t+1) = \omega_{(a,v,t)}(t) + \eta\left[S^{(3)}(a,v,t) - \tau\, \omega_{(a,v,t)}(t)\right]$$

renforcerait la liaison "tri-angulaire" si la synergie tri-modale est forte, ouvrant la possibilité de **clusters** n-aires (ex. un hyper-nœud englobant la portion audio, l'extrait visuel et la phrase textuelle qui décrivent un même évènement).

B. Exemples et Bénéfices Concrets

Prenons le cas d'un **documentaire** où :

- L'**audio** correspond à la voix off expliquant un concept,
- Les **images** (vidéo) montrent un plan rapproché de ce concept,
- Les **sous-titres** (ou résumé textuel) explicitent le contenu.

Si l'on définit un score $S^{(3)}(\text{audio}, \text{visuel}, \text{texte})$ important lorsque les trois coïncident à la fois dans la dimension temporelle (même séquence) et dans la dimension sémantique (même concept), le DSL pourra créer un **hyper-lien** $\omega_{(\text{audio}, \text{visuel}, \text{texte})}$ renforcé. Cela facilite l'indexation — on sait que cet ensemble tri-modal décrit un *même moment clé* du documentaire.

Dans un **clip publicitaire** :

- La piste **audio** (chanson ou jingle),
- Le composant **vidéo** (images dynamiques de produits),
- Le **slogan textuel** (affiché à l'écran à un instant donné).

Un "score triple" détecterait la concordance de trois signaux. La portion musicale, le plan vidéo illustrant le produit et le texte du slogan ou la phrase clé. L'**auto-organisation** du SCN, en exploitant ce score, permet de repérer (ou faire émerger) le moment où tous ces éléments coïncident parfaitement, et ainsi de "tagger" cette scène comme un *macro-événement* tri-modal.

Si l'on veut **retrouver** un évènement spécifique dans un enregistrement massif (spectacle, concert, conférence), on peut rechercher un certain *mot* prononcé sous forme de texte, une *image* ou un plan vidéo avec un cadrage particulier, ou encore un *son* distinct correspondant à une ambiance ou un instrument.

Le score tri-modal peut servir de *filtre* fort ; on ne considère un évènement détecté que lorsque l'audio, la vision et la transcription textuelle s'avèrent mutuellement cohérents. Cela réduit les faux positifs si un canal est bruité ou ambigu.

C. Limites et Considérations

Le **score triple** $S^{(3)}(a, v, t)$ suppose une évaluation pour **chaque** triplet (a, v, t). Si l'on a N_{aud} segments audio, N_{vid} frames/segments vidéo, N_{txt} blocs de texte, le total peut atteindre $\mathrm{O}(N_{\text{aud}} \times N_{\text{vid}} \times N_{\text{txt}})$. Sans sparséfication ni heuristiques, le coût peut devenir prohibitif.

S'il existe déjà une **dynamique** DSL sur :

- $\omega_{(\text{audio,vision})}$,
- $\omega_{(\text{audio,texte})}$,
- $\omega_{(\text{vision,texte})}$,

on peut juger parfois superflu d'introduire un **score tri-modal** explicite, car l'**émergence** d'un cluster reliant (audio, vision, texte) "naturellement" provient du renforcement de ces trois liaisons binaires. Cependant, dans certains cas, la prise en compte d'une **cohérence globale** (parfois non réductible à la seule addition/produit binaires) apporte des capacités plus expressives (ex. on ne veut un cluster qu'en cas de triple validation stricte).

Pour **implémenter** la tri-synergie de manière organique au sein d'un SCN, on peut envisager un **hypergraphe** où un hyper-lien connecte simultanément (a, v, t). La **mise à jour** se fait selon :

$$\omega_{(a,v,t)}(t+1) = \omega_{(a,v,t)}(t) + \eta\left[S^{(3)}(a,v,t) - \tau\,\omega_{(a,v,t)}(t)\right].$$

Cette approche, certes plus complexe à coder, peut **révéler** des schémas tri-modaux plus riches (voir Chap. 12 si l'ouvrage discute de la synergie n-aire et des hyper-liens).

8.6.3.2. Approches additive, multiplicative ou $\min(\cdots)$ pour combiner les synergies binaires

Lorsque l'on souhaite mesurer la **synergie** d'un **groupe** de plusieurs entités, par exemple trois modalités audio–image–texte, à partir des **scores binaires** déjà définis $S(i,j)$, se pose la question de savoir comment **fusionner** ces valeurs binaires pour obtenir un **score global**. Trois grandes approches s'imposent souvent, l'**additive**, la **multiplicative** et la min (ou d'autres "règles de consensus"). Le choix de l'une ou l'autre dépend de la **sémantique** recherchée et du **comportement** souhaité lorsque l'on combine plusieurs synergies.

*A. Approche **Additive***

Dans l'approche **additive**, on définit la **synergie n-aire** (ou tri-modale, etc.) d'un sous-ensemble $\{i_1, i_2, \ldots, i_m\}$ comme la **somme** (ou la moyenne) des synergies **binaires** qui relient les paires (i_a, i_b) à l'intérieur du groupe :

$$S_{\text{add}}(\{i_1, \ldots, i_m\}) = \sum_{1 \le a < b \le m} S\,(i_a, i_b) \quad \text{(somme simple)}$$

ou

$$S_{\text{add}}(\{i_1, \ldots, i_m\}) = \frac{2}{m(m-1)} \sum_{1 \le a < b \le m} S\,(i_a, i_b) \quad \text{(moyenne, normalisée par le nombre de paires).}$$

Cette approche **pondère** toutes les paires (i_a, i_b). Si **toutes** les paires sont fortes, alors la somme (ou moyenne) est grande ; s'il y a du bon et du moyen, on obtient un score intermédiaire, car tout se **cumule**. On peut l'interpréter comme une mesure de la "**cohésion globale**" du groupe où quelques paires faibles peuvent être **compensées** par un grand nombre de paires fortes.

L'un des principaux **avantages** réside dans la mise en œuvre simple et la **lecture** intuitive, basée sur la somme ou la moyenne des synergies binaires. Cette approche reflète l'idée qu'un groupe peut rester **globalement** synergique si la majorité des paires présentent une forte cohésion, même si certaines connexions restent moyennes.

La **limite** de cette méthode est qu'elle est insensible à la faiblesse d'une **unique** paire. Une connexion particulièrement faible peut être noyée dans la somme globale, ce qui rend cette approche moins stricte lorsque l'on souhaite garantir une "cohérence parfaite" entre toutes les paires.

*B. Approche **Multiplicative***

Dans l'approche **multiplicative**, la synergie n-aire s'obtient par le **produit** (ou le **produit transformé**) des synergies binaires :

$$S_{\text{mult}}(\{i_1, \dots, i_m\}) = \prod_{1 \le a < b \le m} S\,(i_a, i_b).$$

Pour des raisons pratiques (éviter sous-flux ou sur-flux), on procède souvent en **logarithmes** :

$$\ln S_{\text{mult}}(\{i_1, \dots, i_m\}) = \sum_{1 \le a < b \le m} \ln\,[S(i_a, i_b)].$$

Le **produit** exprime un **principe** de "tout ou rien" où une seule paire $S(i_a, i_b)$ proche de zéro fait chuter rapidement le **produit total**, indiquant ainsi que chaque **paire** doit être suffisamment forte pour garantir un score élevé. C'est une stricte exigence où *toutes* les binaires doivent coopérer.

L'un des principaux **avantages** de cette approche est qu'elle satisfait un **principe fondamental** affirmant que si un groupe est réellement synergique, *toutes* les paires doivent l'être. Elle se révèle particulièrement adaptée aux scénarios nécessitant une **cohérence stricte**, où l'on exige que *chaque* lien binaire maintienne un niveau de synergie élevé.

Toutefois, cette méthode présente une **limite importante**, car elle se montre extrêmement **sensible** au maillon le plus faible. Si une seule paire ((a, b)) possède une synergie très faible, alors le produit global s'effondre. Cette rigidité peut conduire à ignorer des groupes qui, bien qu'étant bons en moyenne, comportent *une* connexion moins optimale.

C. Approche min (ou "min–like")

Avec la règle $\min(\cdots)$, la synergie n-aire vaut la plus faible des synergies binaires du sous-groupe :

$$S_{\min}(\{i_1, \dots, i_m\}) = \min_{1 \le a < b \le m} S(i_a, i_b).$$

Cela veut dire que, pour considérer un ensemble comme **synergique**, *toutes* les paires doivent être bonnes au même niveau, sinon la min tombe sur la paire la plus faible.

On généralise encore l'idée de **tout ou rien** en imposant que la synergie n-aire ne puisse dépasser la plus faible synergie binaire interne. Cette approche constitue la variante la plus stricte en matière de cohésion des liens. Une seule paire franchement faible *neutralise* la synergie globale.

L'un des principaux **avantages** de cette approche réside dans sa **lecture explicite**, puisqu'elle garantit que le score global ne dépasse jamais la **moins bonne** synergie interne. Elle se montre idéale dans les cas où il est impératif que *toutes* les connexions soient parfaitement alignées, sans possibilité de compromis ou de tolérance aux écarts.

Cependant, cette méthode présente une **limite majeure**, car dans un groupe comportant un grand nombre de paires, une seule dysharmonie suffit à faire chuter l'ensemble du score. Elle ne permet pas de prendre en compte une éventuelle **compensation** par les autres paires, ce qui peut être problématique dans des structures complexes où des ajustements locaux existent sans altérer la cohérence générale.

D. Choix Pratique et Contexte

La **nature** de la fonction de combinaison dépend du **contexte** :

- **Additif** ("moyenne") : on veut un ressenti **global**. Un groupe peut être jugé positif même si certaines paires sont juste correctes.
- **Multiplicatif** : une **exigence** forte d'homogénéité entre toutes les paires. Un seul maillon faible *pénalise* tout le groupe.
- min (plus drastique) : on interdit qu'une **seule** paire soit faiblement synergique.

En **fusion multimodale**, on peut préférer un **produit** (ou un min) si l'on veut exiger l'accord simultané de tous les canaux (par ex. texte–image–audio doivent tous confirmer le même concept). Si un canal est en désaccord, le groupe n-aire est jugé inconsistant.

Pour un sous-ensemble $\{i_1, \dots, i_m\}$, il faut **accéder** aux synergies binaires $S(i_a, i_b)$ pour $\binom{m}{2}$ paires. Ensuite :

- **Additif** : on **somme** ou **moyenne**.
- **Multiplicatif** : on **multiplie** ou on **somme** les logs.
- min : on **prend** la plus petite valeur.

Ces opérations sont assez peu coûteuses si $\binom{m}{2}$ reste modéré.

8.6.3.3. Étude d'exemples (ex. scène d'un concert, documentaire avec sous-titres)

Les principes du **DSL** (Deep Synergy Learning), dans un **SCN** (Synergistic Connection Network) où cohabitent plusieurs modalités (audio, vidéo, texte…), se concrétisent aisément lorsqu'on considère des situations **réelles**. Par exemple, la captation d'un **concert** (audio + vidéo) ou la production d'un **documentaire** (vidéo + audio + sous-titres textuels). L'idée est de montrer de manière **opérationnelle** comment les entités issues de multiples flux se **fusionnent** dans un même réseau, et comment la **dynamique** $\{\omega_{i,j}\}$ fait émerger des **clusters** multimodaux cohérents.

A. Exemple : Scène d'un Concert (Audio + Vidéo)

Considérons une **scène de concert** filmée dans laquelle deux flux principaux interviennent.

Le **flux vidéo** est composé d'une suite d'images, ou plus fréquemment de *frames clés* sélectionnées à partir de l'enregistrement original. Ces frames peuvent être segmentées en *patchs* ou *plans,* puis transformées en **descripteurs** sous forme de vecteurs obtenus par des réseaux neuronaux convolutionnels (CNN).

Le **flux audio** capture la musique, incluant les instruments, les voix et les applaudissements. Ce signal sonore est souvent découpé en *trames* ou *segments* de quelques secondes, ensuite convertis en **features** telles que les coefficients cepstraux en fréquence de Mel (MFCC), les spectrogrammes ou encore des embeddings issus de modèles d'apprentissage profond spécialisés dans l'analyse acoustique.

Dans un **SCN** multimodal, chaque segment $\mathcal{E}_v$ (vidéo) et $\mathcal{E}_a$ (audio) est **intégré** comme un **nœud**. Dès lors, la **synchronicité** ou la **pertinence** qu'ont certains segments à coïncider sur un même moment (ou un même événement musical) se traduit par une **synergie** $S(\mathcal{E}_v, \mathcal{E}_a)$.

Une **fonction** audio–vidéo peut prendre en compte :

$$S(\mathcal{E}_v, \mathcal{E}_a) = \alpha\,\text{correlation\_temporelle}(\mathcal{E}_v, \mathcal{E}_a) \;+\; \beta\,\text{similarite\_features}(\mathcal{E}_v, \mathcal{E}_a),$$

où la composante correlation_temporelle vérifie si le segment audio coïncide en temps avec le plan vidéo (ex. le solo de guitare a lieu de 1:50 à 2:05, la caméra montre le guitariste à la même période), tandis que similarite_features évalue la *complémentarité* plus *sémantique* (par exemple, détection d'un musicien + analyse de la présence d'une guitare dans l'audio).

Le **DSL** met à jour chaque liaison $\omega_{(v,a)}$ suivant la formule usuelle :

$$\omega_{(v,a)}(t+1) = \omega_{(v,a)}(t) \;+\; \eta\big[S(\mathcal{E}_v, \mathcal{E}_a) - \tau\,\omega_{(v,a)}(t)\big].$$

Si le segment $\mathcal{E}_v$ (ex. gros plan sur le guitariste) *coïncide* régulièrement avec l'extrait $\mathcal{E}_a$ (riff de guitare), la synergie demeure élevée et la pondération $\omega_{(v,a)}$ se stabilise à un niveau fort, **forçant** le DSL à **associer** ces deux entités.

Au **fil** des itérations, une organisation structurelle se met en place.

Les **plans vidéo** capturant le batteur ainsi que l'audio associé à un "batterie solo" convergeront vers un **cluster** commun. De même, les **plans** montrant le chanteur seront étroitement reliés à la **voix** correspondante, établissant ainsi des connexions fortes entre les segments visuels et sonores pertinents.

Cette **auto-organisation** permet au **SCN** de différencier divers segments d'un concert (intro, solos, transitions, applaudissements), clarifiant la **structure** de la vidéo musicale. Dans la pratique, on peut associer chaque cluster à un **"morceau"** ou un **"passage"** du concert, facilitant la *recherche* ("retrouve-moi la partie où le guitariste fait son solo") ou le *montage* (on isole d'emblée les segments vidéo–audio les plus synchrones).

B. Exemple : Documentaire avec Sous-Titres (Vidéo + Audio + Texte)

Dans un **documentaire** complet, on trouve :

- Un **flux vidéo** (plans d'images),
- Un **flux audio** (voix off, interventions, musique d'ambiance),
- Des **sous-titres** ou un script texte ajoutés (souvent un fichier $.srt$ contenant $\langle t_{\text{start}}, t_{\text{end}}, \text{texte}\rangle$).

Le **SCN** se dote alors de nœuds vidéo $\{\mathcal{E}_i^{(\text{vid})}\}$, nœuds audio $\{\mathcal{E}_j^{(\text{aud})}\}$ et nœuds texte $\{\mathcal{E}_k^{(\text{txt})}\}$.

On peut définir :

$$S_{\text{vid,aud}}\big(\mathcal{E}_i^{(\text{vid})}, \mathcal{E}_j^{(\text{aud})}\big), \quad S_{\text{aud,txt}}\big(\mathcal{E}_j^{(\text{aud})}, \mathcal{E}_k^{(\text{txt})}\big), \quad S_{\text{vid,txt}}\big(\mathcal{E}_i^{(\text{vid})}, \mathcal{E}_k^{(\text{txt})}\big).$$

Chacune reflète, par exemple :

- **Vidéo–Audio** : alignement temporel d'une scène + bruitage ou dialogue correspondant,

- **Audio–Texte** : transcription plus ou moins exacte de la parole,
- **Vidéo–Texte** : une légende/sous-titre décrivant l'image.

Si le **Chap. 8.6.3.1** évoque la possibilité d'un *score triple* $S^{(3)}(v, a, t)$, la pratique la plus simple demeure de **combiner** ces binaires (ex. additive, multiplicative, min) ou de se contenter de *paires*, puis de laisser la mise à jour DSL *interpénétrer* les trois flux. Ainsi, lorsque qu'un **texte sous-titre** $\mathcal{E}_k^{(\text{txt})}$ coïncide à la fois **temporellement** et **sémantiquement** avec un segment **audio** $\mathcal{E}_j^{(\text{aud})}$, la pondération $\omega_{k,j}$ se renforce naturellement. Simultanément, si la **vision** capture la personne en train de parler, la pondération $\omega_{i,j}$ augmente lorsque $\mathcal{E}_i^{(\text{vid})}$ correspond à la même plage temporelle et / ou sémantique, que ce soit par détection du locuteur ou par reconnaissance faciale. Finalement, ces ajustements conduisent à la formation de **clusters multimodaux**, associant des segments de **vidéo**, des extraits **audio** et des **sous-titres**, chacun tourné autour d'un *même* événement ou *même* contexte.

Dans un processus de **chapitrage**, le réseau de synergie **SCN** peut faire émerger, à travers les pondérations ω, des **macro-nœuds** correspondant à des segments narratifs distincts. Par exemple, dans un documentaire, des chapitres clairement délimités peuvent apparaître, tels que *"présentation de la géologie"* ou *"interview de tel scientifique"*. L'**homogénéité** des données audio, textuelles et visuelles dans chaque bloc renforce cette structuration, facilitant l'interprétation de la *structure narrative globale*.

Dans le cas d'une **correction d'alignement**, un sous-titre $\mathcal{E}_k^{(\text{txt})}$ peut être légèrement *décalé* par rapport à son segment audio associé $\mathcal{E}_j^{(\text{aud})}$. Si ce décalage altère la **synergie audio–texte**, la pondération $\omega_{k,j}$ s'en trouve diminuée. Un **recuit simulé** (Chapitre 7.3) ou un ajustement local peut alors permettre de **réaligner** les segments et de *retrouver* la meilleure *fenêtre de correspondance*, optimisant ainsi la cohérence du lien et augmentant $\omega_{k,j}$.

La **fusion fiable** des canaux multimodaux permet de pallier les défaillances individuelles. Lorsqu'un canal est dégradé, comme une caméra de mauvaise qualité ou un micro qui grésille, les autres modalités peuvent compenser cette perte d'information. Le **SCN** s'auto-organise en renforçant les **clusters** où la somme ou le produit des synergies intermodales demeure élevée, garantissant ainsi une robustesse accrue du réseau.

La **structure multimodale explicite** offerte par le **SCN** met en évidence des **macro-nœuds**, révélant des "moments" ou "thèmes" multimédias. Des entités telles que *Concert — Morceau 1*, *Documentaire — Séquence interview* ou encore *Chat vidéo + Son miaulement + Sous-titre* se distinguent naturellement grâce à la densité des connexions ω. Cette organisation permet une représentation plus intuitive et exploitable du contenu analysé.

Les **applications** sont nombreuses. Dans le cas d'un **concert**, le système peut segmenter les morceaux, détecter les solos et identifier les musiciens à partir des liens renforcés entre les flux audio et vidéo. Dans un **documentaire**, l'indexation automatique permet un chapitrage précis, un alignement optimisé des sous-titres et une recherche avancée. Par exemple, il devient possible d'extraire une séquence où la voix off prononce un terme spécifique tout en affichant un lieu correspondant à l'écran.

8.7. Dynamique d'Auto-Organisation en Contexte Multimodal

Lorsque les entités manipulées (images, textes, sons, etc.) diffèrent grandement par leur nature, la **dynamique** d'auto-organisation du **DSL** (Deep Synergy Learning) doit néanmoins conserver sa **base** théorique. L'actualisation des pondérations $\omega_{i,j}$ suit la **même** formule qu'en contexte monomodal, mais en veillant à employer la fonction de **synergie** appropriée pour chaque paire d'entités. Les sections suivantes (8.7.2, 8.7.3) montreront comment cette mise à jour s'applique à un **SCN** (Synergistic Connection Network) où coexistent plusieurs modalités, et quelles **difficultés** spécifiques surgissent (oscillations, confusion sémantique, etc.).

8.7.1. Mise à Jour des Pondérations

8.7.1.1. Toujours la formule DSL classique :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) \; + \; \eta\big[S(i,j) \; - \; \tau\,\omega_{i,j}(t)\big].$$

Dans la dynamique du **DSL** (Deep Synergy Learning), cette **règle de mise à jour** reste le \textbf{socle principal}, indépendamment du fait que l'on manipule un unique canal (texte, image…) ou des canaux multiples (multimodal). La présente section (8.7.1.1) rappelle :

- **La nature** du terme de synergie $S(i,j)$,
- **Le rôle** simultané du renforcement $\eta\,S(i,j)$ et de l'amortissement $\eta\,\tau\,\omega_{i,j}(t)$,
- **Comment** cette même équation s'applique, sans être modifiée en profondeur, même dans un contexte multimodal complexe.

A. Rappel mathématique de la règle

La mise à jour d'une liaison $\omega_{i,j}(t)$ entre deux entités $\mathcal{E}_i$ et $\mathcal{E}_j$ dans un Synergistic Connection Network (SCN) procède selon une relation discrète qui s'apparente à une équation différentielle en temps discret. On part d'une règle fondamentale

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big].$$

Cette équation peut se lire comme la somme d'un terme d'**incrément** positif lié à la synergie $S(i,j)$ et d'un terme de **décroissance** proportionnel à $\omega_{i,j}(t)$. Le facteur η représente un taux d'apprentissage, tandis que τ contrôle la part d'**amortissement** imposée à chaque liaison. On peut également réécrire la règle de mise à jour sous la forme

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\,S(i,j) - \eta\,\tau\,\omega_{i,j}(t).$$

La différence $\omega_{i,j}(t+1) - \omega_{i,j}(t)$ reflète la dérivée discrète de $\omega_{i,j}(t)$. Cette dérivée comporte un ajout, $\eta\,S(i,j)$, qui renforce la liaison si la synergie est positive, et un retrait, $\eta\,\tau\,\omega_{i,j}(t)$, qui la ramène vers zéro en l'absence de stimulation. Lorsqu'un couple (i,j) affiche un score de synergie élevé, le poids $\omega_{i,j}(t)$ croît jusqu'à l'équilibre $\omega^*_{i,j} \approx S(i,j)/\tau$. Cette valeur d'équilibre est approximative, car d'autres mécanismes, tels que l'inhibition ou le recuit, peuvent se surajouter. Quand la synergie devient faible ou nulle, le terme de décroissance prédomine et $\omega_{i,j}(t)$ chute jusqu'à s'annuler.

L'algorithme DSL consiste alors à itérer cette mise à jour pour toutes les liaisons $\omega_{i,j}$. L'ordre de mise à jour peut varier, mais la formule reste la même pour chaque couple (i,j). Les itérations se poursuivent jusqu'à une forme de convergence, où chaque liaison se stabilise à un niveau dicté par la balance entre la synergie $S(i,j)$ et la dissipation $\tau\,\omega_{i,j}(t)$. D'un point de vue algorithmique, il est possible de traiter toutes les paires (i,j) en parallèle à chaque itération ou par lot (batch) successifs. On peut aussi envisager un mode streaming, où les pondérations se mettent à jour au fur et à mesure que les données arrivent.

Sur le plan énergétique, on peut décrire la configuration globale du réseau par une fonction

$$\mathcal{J}(\mathbf{\Omega}) = -\sum_{i,j} \omega_{i,j}\; S(i,j) + \frac{\tau}{2}\sum_{i,j} (\,\omega_{i,j})^2,$$

où $\mathbf{\Omega}$ regroupe l'ensemble des liaisons $\omega_{i,j}$. Cette fonction comprend un terme négatif $-\omega_{i,j}\,S(i,j)$ qui pousse chaque liaison vers une valeur élevée si la synergie est positive, et un terme de régularisation $\tau/2\;\omega_{i,j}^2$ qui empêche les poids de croître indéfiniment. L'évolution de $\omega_{i,j}(t)$ selon la règle ci-dessus équivaut à opérer une descente sur $\mathcal{J}$, dans un style local et distribué. Chaque liaison s'efforce de minimiser sa contribution à $\mathcal{J}$, stabilisant $\omega_{i,j}$ là où la dérivée partielle par rapport à $\omega_{i,j}$ s'annule.

Cette dynamique aboutit, en pratique, à la formation de **clusters** où les entités ayant de fortes synergies se retrouvent interconnectées par des liaisons $\omega_{i,j}$ élevées, tandis que les entités moins compatibles voient leurs liens mutuels s'étioler. Dans un contexte multimodal, il devient possible de faire émerger des regroupements associant images, texte et audio autour d'un même concept. Le résultat final illustre la logique auto-organisée du DSL où chaque liaison croît ou décroît en fonction d'informations purement locales, jusqu'à produire un état global structurant un réseau de clusters pertinents.

B. Application inchangée pour le multimodal

La puissance du Deep Synergy Learning (DSL) réside dans le fait que la règle de mise à jour des liaisons $\omega_{i,j}$ reste strictement la même, même dans un cadre multimodal intégrant différents types d'entités. Concrètement, la mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big]$$

ne subit aucune modification de forme. Cette invariance garantit que le noyau du DSL (son mécanisme auto-organisé de renforcement et de décroissance) demeure identique, quelle que soit la nature des paires d'entités mises en correspondance. L'élément qui devient plus sophistiqué concerne la définition de la synergie $S(i,j)$, qui doit être capable de gérer un large éventail de modalités.

Lorsque deux entités $\mathcal{E}_i$ et $\mathcal{E}_j$ appartiennent à la même modalité, on applique une fonction de similarité cohérente pour ce flux (par exemple un cosinus ou une distance exponentielle si les deux embeddings proviennent d'un CNN d'images). Lorsque les entités proviennent de modalités différentes, on applique un modèle ou un mapping spécifique correspondant à chaque paire, qu'il s'agisse de texte-texte, image-texte ou audio-texte.

Dans chaque cas, la logique interne du DSL (renforcer la liaison si la synergie se révèle élevée et appliquer une dissipation proportionnelle à $\omega_{i,j}$) ne change pas. La partie qui exige un travail

supplémentaire réside dans l'établissement d'un module ou d'une fonction de similarité adéquate pour chacune des combinaisons de modalités.

Le SCN (Synergistic Connection Network) doit être informé du type de modalité en jeu lorsqu'il calcule la synergie $S(i,j)$. Différentes formules apparaissent selon la nature des paires :

- $S_{\text{img,img}}$. Lorsque les deux entités sont visuelles, on peut recourir à un cosinus entre leurs embeddings CNN ou à une distance RBF sur ces vecteurs.
- $S_{\text{txt,txt}}$. Quand il s'agit de texte vs. texte, il est fréquent d'utiliser un cosinus sur les embeddings textuels (Word2Vec, GloVe, BERT) ou bien une distance TF–IDF.
- $S_{\text{img,txt}}$. Pour relier l'image et le texte, on peut s'inspirer de mécanismes à la CLIP, en projetant l'embedding image et l'embedding texte dans un espace commun, puis en calculant un score de correspondance.
- $S_{\text{aud,txt}}$. Dans le cas audio vs. texte, on peut employer un alignement phonème–mot, des embeddings audio-texte préappris ou un modèle combinant Wav2Vec pour l'audio et BERT pour le texte.

Dès que la synergie associée à un couple (i,j) s'avère élevée à travers plusieurs itérations, la liaison $\omega_{i,j}(t)$ se renforce progressivement, traduisant une cohérence inter-modale repérée de façon répétée.

La même équation de base sert pour tous les flux, ce qui veut dire qu'on peut conserver deux constantes globales η et τ pour l'ensemble du réseau, ou ajuster le taux d'apprentissage η localement. Par exemple, si la similarité audio–audio tend à afficher des valeurs plus faibles que la similarité image–image, on peut accroître le coefficient $\eta_{\text{aud,aud}}$ afin de compenser et de donner un poids similaire aux informations audio. On peut aussi normaliser, à l'avance, la fonction de synergie S pour chaque couple de modalités de sorte à la ramener dans un intervalle uniforme (souvent $[0,1]$). Cette uniformisation évite qu'une seule modalité domine systématiquement la construction du SCN.

Au final, le DSL applique toujours la même règle de renforcement/décroissance pour chaque pondération $\omega_{i,j}$. Le fait de brancher de nouveaux canaux (comme l'audio, la vidéo, le texte) n'oblige pas à repenser le cœur algorithmique du DSL. On se contente de définir ou de calibrer un module de similarité $S_{\text{mod}_a,\text{mod}_b}$ pour chaque paire de modalités mod_a et mod_b. On obtient ainsi une architecture cohérente où tous les flux cohabitent, partageant la même boucle de mise à jour tout en s'enrichissant mutuellement par la formation de clusters réellement multimodaux.

C. Structuration émergente et clusters multimodaux

Le même mécanisme de mise à jour $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$ s'applique de manière uniforme, quel que soit le type des entités $\mathcal{E}_i$ ou $\mathcal{E}_j$. Les seules différences résident dans la façon de calculer la synergie $S(i,j)$ en fonction des modalités mises en jeu, comme les images, l'audio, le texte ou leur combinaison. Cette approche permet à un Synergistic Connection Network (SCN) de bâtir des regroupements internes tant au sein d'une seule modalité que dans un cadre inter-modal. Par exemple, des entités purement visuelles se rassemblent si leur similarité d'embeddings est jugée élevée, et des triplets comprenant des frames vidéo, du son et un sous-titre textuel émergent si les fonctions S inter-modalités soutiennent cette concordance.

Lorsque l'on compare deux images, la synergie s'appuie sur un score de ressemblance visuelle. Lorsque l'on compare une image et un mot, la synergie se fonde sur un module d'alignement comme CLIP. Lorsque l'on compare un segment audio et un label textuel, un mécanisme audio–texte dédié s'emploie pour produire la valeur de synergie. Le SCN procède ensuite à la mise à jour de toutes les liaisons ω. Les liaisons associées à une forte synergie se renforcent et celles qui manquent de cohérence décroissent. Au bout de plusieurs itérations, le réseau donne naissance à des clusters multimodaux, illustrant l'association spontanée d'éléments issus de flux différents.

La robustesse de la fusion tient au fait que le SCN applique simultanément un renforcement au sein de chaque modalité et un renforcement entre différentes modalités lorsque la similarité est confirmée. Un concept se trouve représenté par plusieurs canaux qui partagent une synergie soutenue entre eux. Un exemple simplifie la compréhension. Une image de chat peut se relier fortement à un mot "cat" et un segment audio correspondant à un miaulement, tandis qu'une image de chien reste faiblement connectée au mot "cat". Les pondérations évoluent pour consolider ces liens robustes, et l'on obtient un noyau d'entités qui convergent autour de l'idée d'un "chat", par exemple des images de félins, des sons de miaulement et des mots ou expressions décrivant ce champ sémantique.

Cependant, quand le SCN gère plusieurs types de flux, la taille du problème peut croître rapidement, puisque le nombre total de liaisons $\omega_{i,j}$ est en ordre de n^2. Il devient donc utile de recourir à des heuristiques comme k-NN pour ne calculer la synergie que sur un voisinage réduit, ou à un algorithme d'inhibition pour maîtriser la densité des liens. Le chapitre 7 discute de ces dispositifs qui contribuent à la scalabilité du processus.

La formule de base $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$ reste identique en multimodal. Les modifications se situent plutôt dans le calcul de $S(i,j)$ pour chaque couple de modalités et dans le recours à des méthodes de filtrage ou d'inhibition. Cette constance de la règle autorise une architecture où toutes les entités, quels que soient leurs types, se soumettent au même mode d'actualisation. Les clusters émergents deviennent alors réellement multimodaux. Les éléments image, texte et audio se retrouvent au sein d'une même région du réseau si leurs synergies croisées l'emportent sur l'effet dissipatif. Les niveaux d'organisation qui en résultent sont bien plus riches qu'en traitement monomodal et rendent possible la révélation de concepts ou d'événements qui impliquent plusieurs flux de données de façon simultanée.

8.7.1.2. Adaptation si $\mathrm{type}(i) \neq \mathrm{type}(j)$: application de la synergie correspondante (vision, texte, etc.)

Lorsque le **DSL** (Deep Synergy Learning) prend en charge plusieurs **modalités** — par exemple la **vision**, le **texte** et l'**audio** — la fonction $S(\mathcal{E}_i, \mathcal{E}_j)$ doit impérativement **s'adapter** au **type** respectif des entités $\mathcal{E}_i$ et $\mathcal{E}_j$. C'est-à-dire que, dans la mise à jour

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) \;+\; \eta\big[S(\mathcal{E}_i, \mathcal{E}_j) \;-\; \tau\,\omega_{i,j}(t)\big],$$

le $S(\cdot,\cdot)$ utilisé dépendra de la **modalité** de $\mathcal{E}_i$ et de celle de $\mathcal{E}_j$. Ainsi, le **même** schéma DSL s'applique, mais on **change** la **formule** de la synergie selon les cas $(\mathrm{vision}, \mathrm{vision})$, $(\mathrm{vision}, \mathrm{texte})$, $(\mathrm{audio}, \mathrm{texte})$, ...

A. Logique générale : un "switch" sur les types

Lorsqu'on étend le Deep Synergy Learning (DSL) à un ensemble multimodal, il devient essentiel de distinguer les différentes **modalités** en jeu. Le mécanisme de mise à jour des pondérations $\omega_{i,j}$ reste fondamentalement le même, mais la façon de calculer la synergie $S(i,j)$ doit s'adapter à la nature des entités $\mathcal{E}_i$ et $\mathcal{E}_j$. Pour ce faire, chaque entité $\mathcal{E}_i$ dispose d'un champ $\text{type}(\mathcal{E}_i)$ qui spécifie sa modalité (vision, texte, audio, etc.). Dès lors qu'il s'agit de comparer $\mathcal{E}_i$ et $\mathcal{E}_j$, on se réfère à $\text{type}(i)$ et $\text{type}(j)$ pour déterminer la fonction de synergie adéquate :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \begin{cases} S_{\text{vision,vision}}(\mathbf{v}_i, \mathbf{v}_j), & \text{si } (\text{type}(i) = \text{vision et type}(j) = \text{vision}), \\ S_{\text{vision,texte}}(\mathbf{v}_i, \mathbf{t}_j), & \text{si } (\text{type}(i) = \text{vision et type}(j) = \text{texte}), \\ S_{\text{audio,texte}}(\mathbf{a}_i, \mathbf{t}_j), & \text{si } (\text{type}(i) = \text{audio et type}(j) = \text{texte}), \\ \dots & \dots \end{cases}$$

Cette organisation s'apparente à un "switch" (ou un "dispatch") sur les types de modalité. Les entités visuelles, textuelles et audio peuvent ainsi cohabiter au sein du même Synergistic Connection Network (SCN), tout en assurant qu'à chaque comparaison on emploie la bonne mesure de similarité, qu'elle soit intra-modale ou inter-modale.

Un exemple récurrent est l'association entre une image et un texte. Si $\text{type}(\mathcal{E}_i) = \text{vision}$ et $\text{type}(\mathcal{E}_j) = \text{texte}$, alors on invoque $S_{\text{vision,texte}}$. Ce dernier peut être un cosinus entre deux vecteurs appris par un système du type CLIP, qui projette les embeddings de l'image et du texte dans un espace latent commun. On peut également imaginer un module plus sophistiqué, comme un mini-réseau de co-attention traitant le patch visuel face à une séquence de tokens.

De même, si $\text{type}(\mathcal{E}_i) = \text{texte}$ et $\text{type}(\mathcal{E}_j) = \text{texte}$, on applique $S_{\text{txt,txt}}$. Par exemple, si les entités textuelles sont des phrases, on peut en tirer des vecteurs BERT, puis calculer un cosinus. Si ce sont des mots, on peut recourir à des embeddings statiques (Word2Vec, GloVe) et employer une similarité TF–IDF ou une distance cosinus sur ces vecteurs.

Lorsqu'on évalue un segment audio face à un autre segment audio, on utilise $S_{\text{audio,audio}}$. Cela peut être une distance spectrale ou un produit scalaire entre embeddings acoustiques fournis par Wav2Vec ou un modèle d'autoencodeur sur spectrogrammes.

Cette différenciation entre synergies intra-modales (image–image, texte–texte, audio–audio) et synergies inter-modales (image–texte, audio–texte, etc.) constitue un pivot central dans un SCN multimodal. Sans cette adaptativité, on risquerait de comparer arbitrairement un vecteur d'image à un vecteur de texte sans passer par une fonction appropriée, ce qui invaliderait la convergence du DSL.

Après avoir déterminé la synergie $S(i,j)$ adaptée, la mise à jour de la pondération $\omega_{i,j}$ se déroule selon la règle standard :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big].$$

Cette uniformité du cœur DSL, associée à la diversité dans les modules S, permet une fusion multimodale cohérente et flexible. Les liaisons $\omega_{i,j}$ entre entités potentiellement issues de flux variés se renforcent ou se réduisent en fonction du score produit par la fonction correspondante.

B. Mise à jour $\omega_{i,j}(t+1)$ inchangée

Dans un contexte multimodal, la **même** règle de mise à jour du Deep Synergy Learning (DSL) s'applique sans qu'il soit nécessaire de modifier sa forme fondamentale. Chacune des pondérations $\omega_{i,j}(t)$ entre deux entités $\mathcal{E}_i$ et $\mathcal{E}_j$ suit toujours la relation

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big].$$

Cette uniformité est ce qui confère au Synergistic Connection Network (SCN) une grande souplesse et une grande cohérence. D'un point de vue algorithmique, nul besoin d'introduire des heuristiques particulières ou de scinder la dynamique en plusieurs branches spécialisées lorsque l'on souhaite inclure des données audio, visuelles ou textuelles. Le $\omega_{i,j}$ se met à jour selon le même schéma, qu'il s'agisse de comparer une image et une autre image, ou un segment audio et un texte.

La **seule** évolution, dans le cas multimodal, concerne la définition de la **synergie** $S(i,j)$. Lorsque les modalités de $\mathcal{E}_i$ et $\mathcal{E}_j$ concordent (par exemple image–image), on utilise un module de similarité ajusté à ces embeddings visuels. Lorsqu'il s'agit d'un couple audio–texte, on se réfère à un module de correspondance audio–texte qui sait interpréter un vecteur audio et un embedding lexical. La mise à jour de $\omega_{i,j}$ se fonde alors sur la valeur de $S(i,j)$ fournie par ce module. Autrement dit, le réseau *appelle* la fonction $S_{\text{type}(i),\,\text{type}(j)}$ adaptée, mais la structure de la règle de descente ne varie pas.

Lorsque la synergie demeure suffisante au fil des itérations, le poids $\omega_{i,j}$ s'élève progressivement. Cet effet se produit autant pour des entités de même modalité (deux patches d'image similaires ou deux extraits sonores proches) que pour des entités inter-modales (vision–texte, audio–texte, etc.). Les liens se consolident exactement de la même manière :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\,S(i,j) - \eta\,\tau\,\omega_{i,j}(t).$$

On constate que le terme $\eta\,S(i,j)$ est un incrément de renforcement lorsque la correspondance détectée dans S est positive, tandis que le terme $\eta\,\tau\,\omega_{i,j}(t)$ joue le rôle d'amortissement ou de dissipation, poussant à réduire les poids en l'absence de similarité réaffirmée.

L'**avantage** majeur d'une telle invariant dans la dynamique du DSL réside dans la possibilité d'agréger toutes les entités (vision, texte, audio, etc.) dans une **unique** matrice ω. Il n'y a pas besoin de structurer le réseau en modules de descente distincts ou de superviser différemment la pondération en fonction des flux. On définit simplement une fonction de similarité adaptée pour chaque paire de types, ce qui enclenche la **même** descente pour toute liaison $\omega_{i,j}$. La force du **SCN** se concrétise par un **unique graphe** regroupant tous les nœuds, qu'il s'agisse d'images, de textes ou de pistes audio. Le calcul de $S(i,j)$ varie en fonction de $\text{type}(i)$ et $\text{type}(j)$, mais la mise à jour de ω s'effectue de manière uniforme. Chaque liaison, indépendamment de la modalité d'entrée, augmente en amplitude si la synergie persiste et décroît si la cohérence diminue.

Cette structure unifiée est particulièrement intéressante pour l'**évolutivité**. On peut introduire à n'importe quel moment de nouvelles modalités (par exemple un canal de données sensorielles ou biométriques) en ajoutant un module de calcul de synergie approprié, mais **sans** devoir revoir le mécanisme de descente, qui reste

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big].$$

L'**auto-organisation** multimodale s'opère alors par ce **processus** local et homogène, permettant de détecter aussi bien des clusters unimodaux (un ensemble de documents textuels très proches) que des clusters inter-modaux (un segment audio, une image et un texte décrivant la même scène). Chaque paire d'entités s'actualise de la même façon, gage de cohérence globale du SCN et de facilité d'implémentation.

C. Résultat : structure unifiée avec clusters mixtes

En appliquant la même règle de mise à jour à tous les couples (i, j), qu'ils soient issus d'une modalité unique ou de deux flux différents, le Synergistic Connection Network (SCN) produit une **structure** finale où les **clusters** émergent spontanément. Certaines de ces formations correspondent à des regroupements internes dans la même modalité. Un ensemble d'images semblables peut constituer un cluster « purement visuel », tandis que des mots ou des segments de texte fortement corrélés se rassemblent en un cluster purement textuel. Par ailleurs, grâce à la synergie inter-modale, des **clusters** plus complexes unissant plusieurs canaux apparaissent lorsque leurs pondérations se renforcent suffisamment de manière conjointe.

Un exemple typique illustre cette intégration. Supposez qu'un cluster "chat" prenne forme en rassemblant plusieurs entités. Un certain nombre d'images de chat, diverses expressions textuelles telles que "cat" ou "petit félin domestique" et un enregistrement audio portant sur le miaulement d'un chat. Dans cette configuration, chaque liaison $\omega_{i,j}$ reliant une image et un mot aura été renforcée, car la fonction de similarité image–texte aura perçu une correspondance (par exemple via un embedding de type CLIP). De même, si l'on dispose d'un module audio–image capable de reconnaître la coïncidence entre un miaulement et l'apparition visuelle d'un chat, les pondérations ω reliant l'entité audio au patch d'image auront crû de façon similaire. C'est donc la **même** dynamique :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big],$$

qui a permis, localement et au fil des itérations, d'accumuler des liaisons élevées vers un **cluster** cohérent, à la fois visuel, textuel et audio. Ce "co-assemblage" reflète la puissance auto-organisée du DSL.

Dans un usage pratique, un utilisateur ou une application peut alors **interroger** le SCN de diverses manières. En partant d'un simple mot ("cat"), on retrouvera aisément les images et les sons associés dans le cluster, car les pondérations ω reliant "cat" à ces entités s'avèrent particulièrement élevées. Inversement, on peut partir d'une image et remonter vers le segment audio ou le label textuel qui lui correspond le mieux dans le réseau. À l'échelle plus large, ce même mécanisme enrichit les possibilités d'**indexation** et d'**annotation** multimédia, ainsi que de **clustering** sémantique intégrant plusieurs canaux. Autrement dit, un seul et même **SCN** exploite l'auto-organisation du DSL pour faire émerger des sous-groupes thématiques ou conceptuels très riches, couvrant vision, texte et audio au sein d'une structure unifiée.

8.7.1.3. Contrôle de la Compétition (Inhibition, Saturation) pour Éviter la Saturation en Liens

Dans un **SCN** (Synergistic Connection Network) conçu pour la multimodalité (cf. §8.7.1.1 et §8.7.1.2), l'abondance de flux et d'entités — images, audio, textes, etc. — peut conduire à une **profusion** de connexions $\omega_{i,j}$ si l'on ne restreint pas la formation ou la croissance des liens moyens. Cette **surabondance** nuit à la fois à la **lisibilité** du réseau et à la **qualité** des clusters formés, tout en alourdissant le **coût** algorithmique. Pour y remédier, on recourt à des

mécanismes de **compétition** où l'**inhibition** limite la somme des liaisons sortantes d'une entité et la **saturation** borne la valeur maximale de chaque liaison.

A. Inhibition Dynamique et Limitation de la Somme des Liens Sortants

La règle de mise à jour initiale du Deep Synergy Learning (DSL) définit la croissance d'une liaison $\omega_{i,j}(t)$ en fonction de la synergie $\eta\, S(i,j)$ et d'un amortissement $\eta\, \tau\, \omega_{i,j}(t)$. Cette équation peut être étendue pour inclure un mécanisme d'inhibition qui maintient la somme des liens sortants d'un nœud à un niveau modéré. On ajoute alors un terme supplémentaire au moment de l'incrément de $\omega_{i,j}(t)$. Cette forme modifiée de la règle d'actualisation est décrite par

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big] - \gamma \sum_{k \neq j} \omega_{i,k}\,(t).$$

Le coefficient positif γ introduit une compétition parmi les liaisons sortant d'une même entité $\mathcal{E}_i$. À chaque itération, la somme $\sum_{k\neq j} \omega_{i,k}\,(t)$ reflète la quantité de ressources que $\mathcal{E}_i$ mobilise déjà pour d'autres liens. La présence de ce terme dans la mise à jour de $\omega_{i,j}(t)$ freine la croissance simultanée d'un trop grand nombre de liaisons de force comparable. Une entité trop dispersée dans ses connexions se voit ainsi freinée chaque fois qu'elle tente d'augmenter encore un lien sortant.

Cette pénalisation se rapproche d'une contrainte de type L1 sur la somme des pondérations sortantes, car plus la somme des liens $\omega_{i,k}$ augmente, plus la dérivée instantanée de $\omega_{i,j}(t)$ se retrouve négative pour les nouveaux liens qui cherchent à se renforcer. Un nœud doit donc "choisir" les connexions les plus significatives afin de contrer la décroissance liée à l'inhibition. Il s'agit d'une forme de compétition latérale qui rend le réseau plus épars et plus lisible. Les liens qui ne compensent pas assez la pénalisation disparaissent ou se réduisent, tandis que les liaisons vraiment pertinentes survivent en maintenant un score de synergie suffisamment fort pour surpasser la dissipation et l'inhibition.

La conséquence de ce phénomène est l'apparition de clusters plus marqués dans le Synergistic Connection Network, car on évite la situation où un nœud consolide un grand nombre de liaisons moyennes. La dynamique reposant sur une équation localement inchangée, on préserve la nature auto-organisée du DSL. Le coefficient γ se règle selon le degré de parcimonie recherché. Une valeur plus grande force une séparation plus franche en poussant les nœuds à concentrer leurs pondérations sur un petit ensemble de voisins, alors qu'une valeur modérée autorise un certain niveau de dispersion dans les liaisons sortantes. Dans tous les cas, on obtient un contrôle plus fin de la topologie du réseau et de la clarté des communautés qui émergent, sans remettre en cause la base du mécanisme de renforcement et d'amortissement déjà en place.

B. Saturation (Clipping) : Borne Supérieure sur Chaque Liaison

L'inhibition dans un Synergistic Connection Network (SCN) restreint déjà la somme totale des liaisons sortantes d'un nœud, mais il demeure possible qu'une seule liaison $\omega_{i,j}$ prenne une valeur trop élevée si la synergie $S(i,j)$ reste très grande. Afin d'éviter qu'un poids ne devienne démesuré, il est judicieux d'introduire un mécanisme de **clipping** qui impose une borne maximale $\omega_{\max}$. Concrètement, après avoir mis à jour la pondération selon la règle du DSL,

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big],$$

on applique la saturation

$$\omega_{i,j}(t+1) \leftarrow \min\big(\omega_{i,j}(t+1),\, \omega_{\max}\big).$$

Cette borne supérieure limite la croissance d'un poids, même si le terme $\eta\, S(i,j)$ encourageait une valeur potentiellement bien plus grande. On empêche ainsi un lien isolé de dominer le réseau à cause d'une synergie exceptionnellement forte. La saturation présente aussi l'avantage de stabiliser plus rapidement la dynamique si une liaison tend à atteindre un équilibre très élevé. Une fois que $\omega_{i,j}$ sature, la mise à jour du DSL ne la fait plus grimper, ce qui allège la pression pour d'autres liaisons et contribue à un ajustement global plus rapide.

Il est en outre possible de rendre la borne $\omega_{\max}$ temporellement variable. On peut par exemple choisir une fonction $\omega_{\max}(t) = \omega_{\max}^{(0)} + \beta\, t$ afin de démarquer un seuil initial bas, imposant une plus grande parcimonie, puis d'autoriser une expansion progressive des valeurs de ω à mesure que le réseau se structure. Cette forme d'adaptation temporelle peut s'avérer utile si la phase initiale de l'apprentissage exige un tri rapide des liens alors que la phase avancée autorise un raffinement plus généreux.

En définitive, le clipping n'affecte pas la forme essentielle de la règle de mise à jour. Il agit après coup, comme un correctif local sur la valeur de $\omega_{i,j}(t+1)$. On maintient ainsi l'**équation** :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big],$$

puis on applique $\omega_{i,j}(t+1) = \min\big(\omega_{i,j}(t+1),\, \omega_{\max}\big)$. L'objectif est de restreindre chaque liaison au-delà d'une certaine valeur, évitant la surreprésentation d'un lien unique et contribuant à un SCN plus stable et plus lisible.

C. Combinaison des Deux Mécanismes

La version la plus complète de la mise à jour associe l'**inhibition** et la **saturation** en un seul schéma. La règle du DSL devient

$$\omega_{i,j}(t+1) = \{\omega_{i,j}(t) + \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big] - \gamma \sum_{k \neq j} \omega_{i,k}\,(t)\}_{\text{clip à } \omega_{\max}},$$

où l'on borne la valeur obtenue au maximum $\omega_{\max}$. On cumule ainsi deux principes où la compétition latérale imposée par $\gamma \sum_{k \neq j} \omega_{i,k}\,(t)$ et la restriction de tout poids $\omega_{i,j}$ à la valeur $\omega_{\max}$. L'inhibition contraint une entité $\mathcal{E}_i$ à distribuer ses ressources, tandis que la saturation empêche une liaison unique de grimper sans limite.

D. Éviter la Saturation en Liens pour un SCN Multimodal

Dans un SCN volumineux, où cohabitent de multiples modalités (images, textes, audio, etc.), il arrive qu'une entité se connecte à un très grand nombre d'autres entités avec un niveau moyen de pondération. Un graphe trop dense en résulte, ce qui complique la convergence et la lecture de la structure. L'inhibition force une entité à sélectionner plutôt que de maintenir un large éventail de liaisons non spécialisées. Simultanément, la saturation borne les liaisons dominantes, ce qui évite qu'une poignée de poids monopolise la dynamique.

Ce double contrôle se révèle précieux dans un contexte multimodal. Au lieu d'observer un nœud qui entretient "un peu" de synergie avec des dizaines ou des centaines d'éléments, le SCN fait émerger des grappes plus resserrées où le lien reste fortement justifié par la fonction S. Chacun des deux mécanismes agit à un stade différent où l'inhibition pénalise la somme des pondérations sortantes, alors que la saturation limite la valeur maximale d'une liaison

individuelle. Une entité ne peut plus étendre ses connexions de manière indifférenciée et ne peut pas non plus pousser un lien particulier au-delà de $\omega_{\max}$.

Cette approche encourage un réseau plus parcimonieux et plus stable. L'algorithme DSL, identique dans sa structure fondamentale, voit simplement chacun de ses incréments modifiés par l'inhibition et finalisé par un éventuel clipping. Le résultat, sur le plan expérimental, est un **SCN** plus lisible, dont les clusters multimodaux se démarquent clairement, sans que des liens moyens ou disproportionnés ne viennent obscurcir la structure globale. Sur le plan algorithmique, on définit les paramètres γ et $\omega_{\max}$ selon les caractéristiques du problème, en tenant compte de la densité visée et du niveau de compétition qu'on souhaite introduire.

8.7.2. Émergence de Clusters Multimodaux

Au sein d'un **SCN** (Synergistic Connection Network) gérant des **entités multimodales** (images, textes, sons, etc.), la dynamique DSL (Deep Synergy Learning) peut donner naissance à des **clusters** réunissant des contenus de **natures différentes** autour d'un **thème** ou d'un **concept** commun. L'auto-organisation s'applique alors non seulement à l'intérieur de chaque modalité (groupement de documents textuels, similarité entre images, etc.) mais aussi **entre** modalités, via des mesures de synergie "cross-modales" (texte–image, son–image, etc.). Ce phénomène, particulièrement visible lorsque l'on considère la **fusion** de plusieurs flux (8.7.1), se manifeste ici par la **cohésion** de données variées en un **cluster** multimodal.

8.7.2.1. Cas d'un cluster contenant **images**, **textes** et **sons** liés par un **thème** commun

Les données multimédias se prêtent particulièrement bien à la logique du **Deep Synergy Learning (DSL)**, dès lors que l'on peut construire une **synergie** multimodale entre des entités hétérogènes (images, textes, sons). Dans ce **cas d'usage**, on envisage la formation d'un **cluster** $\mathcal{C}$ qui réunisse des éléments de plusieurs modalités autour d'un **thème** commun où les **pondérations** entre des images, des extraits sonores et des textes **se renforcent** dès lors que ces documents véhiculent une même sémantique sous-jacente, comme un style musical, un genre cinématographique ou un concept scientifique.

A. Situation de Base

Considérons une **base** de données multimédia composée de trois **ensembles** :

6. $\mathcal{I} = \{\mathcal{E}_i^{(\text{img})}\}$: un **ensemble d'images**,
7. $\mathcal{T} = \{\mathcal{E}_j^{(\text{txt})}\}$: un **ensemble de textes** (articles, résumés, descriptions),
8. $\mathcal{A} = \{\mathcal{E}_k^{(\text{aud})}\}$: un **ensemble d'extraits audio** (fragments musicaux, enregistrements de parole, bruitages, etc.).

Chaque **entité** $\mathcal{E}_u$ appartient donc à l'une de ces **modalités**. Dans l'optique du DSL, on dispose d'une **fonction** de **synergie** $S(\mathcal{E}_u, \mathcal{E}_v)$, notée aussi $S(u, v)$, qui **mesure** la pertinence ou l'**affinité** entre deux entités, **quelle que** soit leur modalité d'origine. Ainsi, $\mathcal{E}_u$ peut être une image, $\mathcal{E}_v$ un texte, et la **synergie** $S(\mathcal{E}_u, \mathcal{E}_v)$ exprime le **score** de "cohérence" entre l'image et le texte (voir la section **2.2.1.2** pour les grandes familles de *similarités*). L'**objectif** est de laisser

le **DSL** faire émerger un **cluster** combinant des $\mathcal{E}_i^{(\text{img})}$, des $\mathcal{E}_j^{(\text{txt})}$ et des $\mathcal{E}_k^{(\text{aud})}$ relatifs à un même **thème**.

B. Fonction de Synergie Cross-Modale

Le Deep Synergy Learning (DSL) s'applique à un Synergistic Connection Network (SCN) dans lequel les entités peuvent appartenir à plusieurs modalités (image, texte, audio, etc.). La seule manière pour le réseau de **fusionner** ces sources hétérogènes est de définir une **mesure** de synergie $S(\mathcal{E}_i, \mathcal{E}_j)$ entre chaque paire, y compris lorsque leurs modalités diffèrent. Pour cela, on procède à la construction de **fonctions partielles** spécialisées selon la nature des entités, puis on les assemble de manière cohérente afin de couvrir l'ensemble des combinaisons (image–image, image–texte, texte–audio, etc.). Sur le plan mathématique, cela débouche sur une "table" de fonctions $S_{\text{mod1,mod2}}$ qui, pour chaque couple (i, j), fournit un score de similarité ou de correspondance.

Synergie image–texte

Lorsque l'on compare une entité $\mathcal{E}_i^{(\text{img})}$ (image) à une entité $\mathcal{E}_j^{(\text{txt})}$ (texte), on part généralement d'un embedding visuel $\mathbf{v}_i^{(\text{img})} \in \mathbb{R}^{d_{\text{img}}}$ et d'un embedding textuel $\mathbf{v}_j^{(\text{txt})} \in \mathbb{R}^{d_{\text{txt}}}$. On définit ensuite une **fonction** $S_{\text{img-txt}}(\cdot,\cdot)$ permettant d'estimer la compatibilité sémantique entre ces deux vecteurs. Par exemple, on peut recourir à un modèle de type CLIP, où l'on projette image et texte dans un espace latent partagé grâce à deux encodeurs Φ_{img} et Φ_{txt}, puis on compare le résultat via une similarité cosinus ou exponentielle :

$$S\left(\mathcal{E}_i^{(\text{img})}, \mathcal{E}_j^{(\text{txt})}\right) = \exp(-\alpha \, \| \Phi_{\text{img}}\left(\mathbf{v}_i^{(\text{img})}\right) - \Phi_{\text{txt}}\left(\mathbf{v}_j^{(\text{txt})}\right) \|).$$

Cette approche évalue à quel point le concept véhiculé par l'image s'aligne sur le concept véhiculé par le texte, condition nécessaire pour que la liaison $\omega_{i,j}$ se renforce dans le SCN.

Le lien entre un extrait audio $\mathcal{E}_k^{(\text{aud})}$ et un texte $\mathcal{E}_j^{(\text{txt})}$ exige lui aussi un "pont" multimodal. Chaque entité audio est décrite par un vecteur $\mathbf{v}_k^{(\text{aud})} \in \mathbb{R}^{d_{\text{aud}}}$ (par exemple un embedding généré par Wav2Vec), tandis que le texte $\mathbf{v}_j^{(\text{txt})} \in \mathbb{R}^{d_{\text{txt}}}$ peut provenir d'un encodeur BERT. On conçoit alors une fonction

$$S\left(\mathcal{E}_k^{(\text{aud})}, \mathcal{E}_j^{(\text{txt})}\right) = \text{sim}\left(\psi_{\text{aud}}\left(\mathbf{v}_k^{(\text{aud})}\right), \psi_{\text{txt}}\left(\mathbf{v}_j^{(\text{txt})}\right)\right),$$

où sim s'appuie sur un cosinus ou un score dérivé d'un alignement sémantique audio–texte. Dans ce cas, ψ_{aud} et ψ_{txt} sont des mappings ou des encodeurs entraînés à rapprocher le contenu sonore et le contenu textuel équivalent.

Bien qu'elle soit moins fréquente que l'appariement image–texte, la comparaison d'une image et d'un signal audio peut également se rencontrer. Des travaux émergent pour aligner un son (bruit d'océan, rugissement d'animal, etc.) et une image représentative (vue de plage, photo d'un lion). La fonction de synergie

$$S\left(\mathcal{E}_i^{(\text{img})}, \mathcal{E}_k^{(\text{aud})}\right)$$

peut faire usage d'un encodeur visuel et d'un encodeur audio projetant les deux vecteurs dans un espace latent, puis comparer le résultat par un cosinus ou une distance gaussienne. Dans un

SCN, cela donne la possibilité de lier, par exemple, un patch d'image de voiture et un segment audio de moteur.

Il ne faut pas négliger la synergie au sein de la même modalité. Deux images $\mathbf{v}_i^{(\text{img})}$ et $\mathbf{v}_j^{(\text{img})}$ se comparent par une similarité cosinus ou une distance RBF pour établir à quel point elles représentent des scènes similaires. Deux morceaux de texte $\mathbf{v}_u^{(\text{txt})}$ et $\mathbf{v}_v^{(\text{txt})}$ font l'objet d'un cosinus sur leurs embeddings BERT ou Word2Vec, tandis que deux segments audio $\mathbf{v}_p^{(\text{aud})}$ et $\mathbf{v}_q^{(\text{aud})}$ se rapprochent si leurs signatures spectrales se recoupent. Ce sont souvent ces synergies intra-modales qui forment la base des clusters unimodaux.

Le SCN devient un graphe dont les nœuds renvoient à des entités variées (images, extraits audio, textes, etc.), et dont les arêtes correspondent aux pondérations $\omega_{i,j}$. À chaque paire $(\mathcal{E}_i, \mathcal{E}_j)$, on associe un score $S(i,j)$ défini par la fonction de similarité ou de correspondance multimodale appropriée. Mathématiquement, on bâtit plusieurs **fonctions** de synergie partielles (image–image, texte–texte, image–texte, audio–texte, etc.) et on les "**colle**" pour en obtenir une version globale cohérente sur l'union des espaces. Les principes de "collage" discutés en section 2.2.1.1 (théorème de consistance) assurent la régularité de S lorsque plusieurs blocs d'entités se rencontrent.

Au moment d'activer la dynamique du DSL, la mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big]$$

se base sur cette fonction globale S. Les paires jugées cohérentes à travers leur modalité (ou entre différentes modalités) voient leurs liaisons renforcées ; celles qui ne révèlent pas de correspondance s'affaiblissent jusqu'à s'annuler, favorisant l'émergence naturelle de clusters mixtes ou purement unimodaux selon les scénarios.

C. Émergence d'un Cluster Multimodal

Une fois la fonction de **synergie** $S(u,v)$ définie pour chaque couple (u,v), y compris dans un cadre multimodal, le Deep Synergy Learning (DSL) met en branle la **règle** de mise à jour des pondérations $\omega_{u,v}(t)$. Cette règle, rappelée en **section 2.2.2**, se présente typiquement sous la forme

$$\omega_{u,v}(t+1) = \omega_{u,v}(t) + \eta\big[S(u,v) - \tau\,\omega_{u,v}(t)\big].$$

L'algorithme effectue cette mise à jour à chaque itération ou "tick" du réseau, de sorte que les liaisons $\omega_{u,v}$ se renforcent si la synergie $S(u,v)$ est jugée importante, tout en subissant une décroissance $\tau\,\omega_{u,v}(t)$ si cette synergie n'est pas régulièrement réaffirmée. Lorsque $S(u,v)$ reste élevé sur plusieurs itérations, $\omega_{u,v}$ se fixe approximativement vers $S(u,v)/\tau$. À l'inverse, deux entités n'ayant pas de correspondance prononcée voient leur liaison décroître vers zéro.

Le **processus** auto-organisé du DSL, via cette règle de mise à jour, conduit naturellement à la **constitution** de clusters lorsque différents canaux (image, texte, audio, etc.) décrivent un même **thème** sous des angles complémentaires. Dans un exemple, un sujet "Jazz" peut rassembler :

Des **entités visuelles** : Plusieurs images ou photos de musiciens, d'instruments (saxophone, trompette), ou de pochettes d'album associées au Jazz.

Des **entités textuelles** : Fragments de texte ou mots clés tels que "Miles Davis", "Duke Ellington", "improvisation", "swing", "blue note", etc.

Des **entités audios** : Extraits sonores présentant la signature harmonique jazz (rythmique swing, modes, instruments spécifiques) reconnus par un classifieur audio ou un embedding Wav2Vec.

Chacune de ces entités, prise individuellement, relève de sa propre modalité (vision, texte, audio). Mais le SCN peut faire correspondre une image spécifique (un musicien en concert) à un extrait textuel (mentionnant l'improvisation jazz) et à un segment audio (solo de saxophone). Le fait que la **synergie** conserve des scores élevés (image–texte, texte–audio, image–audio) sert de moteur à la consolidation de liaisons $\omega_{i,a}$, $\omega_{a,p}$, $\omega_{i,p}$ dans le DSL.

À mesure que les pondérations se stabilisent, on voit ainsi **émerger** un **cluster** cohérent, noté $\mathcal{C}_{\text{jazz}}$, qui regroupe ces images, textes et extraits audio autour de la même thématique. On parle alors de **cluster multimodal**, car il ne se limite pas à une catégorie de données, il embrasse des éléments différents unis par une idée commune.

Exemple concret

On suppose que certains nœuds $\{\mathcal{E}_1^{(\text{img})}, \dots, \mathcal{E}_m^{(\text{img})}\}$ correspondent à des photographies de concerts de Jazz (musiciens, instruments, ambiance de club). D'autres nœuds $\{\mathcal{E}_a^{(\text{txt})}, \dots\}$ sont des passages de texte citant "Duke Ellington", "Miles Davis", "improvisation", "swing", et ainsi de suite. Enfin, d'autres nœuds $\{\mathcal{E}_p^{(\text{aud})}, \dots\}$ renvoient à des enregistrements où un modèle audio détecte un tempo, une harmonie ou un style caractéristique du Jazz.

La fonction de synergie inter-modale identifie qu'une image exhibant un saxophone $\mathbf{v}_i^{(\text{img})}$ se rapproche d'un mot "saxophone" $\mathbf{v}_a^{(\text{txt})}$, ou qu'un extrait sonore $\mathbf{v}_p^{(\text{aud})}$ présente une corrélation forte avec ce concept musical. À chaque itération DSL, les liaisons $\omega_{i,a}, \omega_{a,p}, \omega_{i,p}$ se **renforcent** car la valeur $S(i,a)$, $S(a,p)$, $S(i,p)$ demeure élevée. Les entités finissent par "coaguler" en un **cluster** $\mathcal{C}_{\text{jazz}}$ unissant images, textes et audio référant à la thématique Jazz.

D. Vision Mathématique : Liens dans un Graphe Hétérogène

On peut formaliser l'ensemble des entités images–textes–audio dans un **graphe** $\mathcal{G}$ (ou hypergraphe), dont les **nœuds** u représentent les entités $\mathcal{E}_u$. À chaque couple (u, v) est associée une **pondération** $\omega_{u,v}(t)$, qui évolue selon la **dynamique** DSL. Avec un **terme** de *parsimonie* $\omega_{\min}$ (section 2.2.3), on peut élaguer les liens trop faibles. Au terme d'un certain nombre d'itérations, un **sous-graphe** cohésif apparaît là où la **synergie** est notable, formant ainsi un **cluster** transcendant la diversité des modalités.

Si l'on se limite à la **règle** additive, on a :

$$\omega_{u,v}(t+1) \;=\; (1-\eta\,\tau)\,\omega_{u,v}(t) \;+\; \eta\,S(u,v),$$

et la convergence se produit quand $\omega_{u,v}(t+1) \approx \omega_{u,v}(t) \approx \frac{S(u,v)}{\tau}$.

E. Rôle dans la Découverte de Thèmes

Le Deep Synergy Learning (DSL), lorsqu'il est appliqué à un réseau d'entités multimodales (images, texte, audio, voire plus), autorise l'**auto-organisation** de ces données en **clusters** sans supervision préalable. Les pondérations $\omega_{i,j}$ se mettent à jour selon la règle de base, et la **synergie** $S(i,j)$ détermine dans quelle mesure deux entités de différentes modalités se révèlent pertinentes l'une pour l'autre. Cette logique permet la **découverte** de thèmes transversaux qui

associent librement des informations diverses, sans qu'on impose de catégories nommées comme "Jazz", "Classique" ou "Pop".

Lorsque le SCN opère en mode non supervisé, aucune étiquette n'est imposée pour guider la constitution des clusters. Les images, le texte et l'audio cohabitent dans le même graphe, et chaque pondération $\omega_{i,j}(t)$ grandit ou décroît au fil des itérations en fonction du score de synergie $S(i,j)$. Peu à peu, le réseau forme des **groupements** stables, chacun reflétant un **thème** ou un **concept** partagé. Par exemple, un groupe peut réunir des images de saxophones, des extraits sonores de jazz et des références textuelles à "Miles Davis" ou "improvisation", même si le système n'a jamais reçu explicitement le label "Jazz". L'émergence de ce cluster s'opère simplement parce que la synergie image–texte–audio demeure élevée entre les entités concernées. Cette capacité à s'**auto-organiser** sans label explicite est ce qui distingue le DSL d'architectures strictement supervisées.

Le **principe** multimodal fournit une **complémentarité** entre les flux. Quand une entité visuelle (une photo de saxophone) affiche déjà une cohérence avec un extrait textuel (mention d'un instrument "saxophone"), et que l'audio lié présente un spectre caractéristique du jazz, l'association se trouve renforcée simultanément sur trois axes (image–texte, texte–audio, image–audio). Les indices se **soutiennent** mutuellement puisqu'une image isolée de saxophone pourrait prêter à confusion si elle n'est pas identifiée, mais le texte "saxophone" et l'extrait audio "solo de sax" confirment la cohérence. Le résultat est un **cluster** plus fiable et précis que s'il n'y avait qu'une seule modalité pour juger de la similarité.

Cette **philosophie** multimodale s'étend aisément à de nouvelles sources de données. Il suffit de définir une fonction de synergie S adéquate pour la paire (nouvelle modalité, modalité existante). Par exemple, pour de la **vidéo**, on conçoit un encodeur vidéo (CNN 3D, transformeur spatio-temporel) permettant d'extraire un embedding, puis on définit un score de correspondance vidéo–image ou vidéo–texte, etc. De même, l'ajout de **capteurs** IoT (température, pression, localisation GPS) se gère en construisant un module de similarité entre ces mesures et les éventuels flux sémantiques. Au niveau énergétique, la somme

$$\sum_{u,v} \omega_{u,v}\ S(u,v)$$

s'enrichit de contributions supplémentaires, reflétant la synergie que de nouveaux canaux peuvent apporter. Le SCN élargit alors la possibilité de faire **émerger** des **communautés** ou des thèmes encore plus vastes, associant différents types d'informations et conservant la logique habituelle de renforcement et de dissipation ($\tau\ \omega_{u,v}$).

La conséquence finale est un **réseau** dont la topologie s'ajuste de manière distribuée pour faire ressortir des **clusters** multimodaux. Ils peuvent être observés comme des "macro-nœuds" qui regroupent un ensemble d'éléments disparates (visuels, textuels, audio, etc.) tous liés par des pondérations ω suffisamment fortes. Cette dynamique exploite pleinement la **complémentarité** des flux pour découvrir des **thèmes** qui n'étaient pas explicitement nommés au départ, rendant le DSL particulièrement adapté aux tâches de fouille de données et d'exploration non supervisée dans des environnements riches et diversifiés.

8.7.2.2. Outils pour repérer ces groupes de nœuds hétérogènes

Dans un **SCN** (Synergistic Connection Network) où coexistent plusieurs **modalités** (images, textes, sons, etc.), il arrive fréquemment qu'un **cluster** rassemble des nœuds aux

caractéristiques très **dissemblables** tout en entretenant entre eux des **liaisons** $\omega_{i,j}$ significatives. Ces *groupes hétérogènes* peuvent révéler des **patterns** transversaux (thèmes ou processus communs) ou, au contraire, signaler des **anomalies** (brassage inhabituel de modalités). La présente section dresse un **panorama** des méthodes et **indicateurs** permettant d'**identifier** ces **regroupements** multimodaux dans le cadre d'un **DSL**.

A. Analyse des Entropies et Indicateurs de Diversité

Une **approche** classique pour quantifier la **multimodalité** d'un sous-groupe $\mathcal{C} \subseteq \{\text{nœuds}\}$ consiste à mesurer l'**entropie** de sa **composition**. En supposant que $\mathcal{C}$ renferme des entités issues de plusieurs **modalités** $\text{m} \in \{\text{img}, \text{txt}, \text{aud}, \dots\}$, on note

$$p_m(\mathcal{C}) \;=\; \frac{|\{\, i \in \mathcal{C} \mid \text{mod}(i) = m \,\}|}{|\mathcal{C}|} \quad \text{(pour chaque modalité } m\text{)}.$$

L'**entropie** de Shannon se définit alors par

$$H(\mathcal{C}) \;=\; -\sum_{m} p_m(\mathcal{C}) \,\ln\big(p_m(\mathcal{C})\big).$$

Plus $H(\mathcal{C})$ est **élevée**, plus la **composition** en modalités est **diversifiée**, signifiant que $\mathcal{C}$ "mélange" fortement des nœuds de natures variées. À l'inverse, un **cluster** quasi-monolithique (ex. presque uniquement des images) aura un $H(\mathcal{C})$ faible. On peut alors décider de **repérer** les clusters où $H(\mathcal{C}) > H_{\min}$ pour ne retenir que les sous-groupes vraiment **hétérogènes**.

Au-delà de l'entropie, divers **indices** de **diversité** (Simpson, Gini, Hill numbers, etc.) sont employés en écologie ou en classification pour rendre compte de la **répartition** des classes. Par exemple, l'**indice de Simpson** sur la distribution $\{p_m\}$ se définit par

$$D_{\text{Simpson}}(\mathcal{C}) \;=\; 1 \;-\; \sum_{m} \big(p_m(\mathcal{C})\big)^2.$$

Il atteint son maximum lorsque les p_m sont **uniformes**. Ainsi, pour la **détection** de clusters "vraiment multimodaux", on peut filtrer les valeurs supérieures à un seuil $D_{\min}$. D'un point de vue **mathématique**, ces indices donnent un **chiffre** synthétique de l'**hétérogénéité** interne, ce qui facilite la **comparaison** et le **rang** des clusters trouvés par un algorithme DSL ou de partition.

B. Cartographie du Degré ou des Forces de Liaison Inter-modales

Dans le cadre d'un **SCN multimodal**, on peut examiner les **liaisons** $\omega_{i,j}$ en **stratifiant** par **paires** de modalités $(\text{m1}, \text{m2})$. Soit $\mathbf{W}^{(\text{m1},\text{m2})}$ la **sous-matrice** contenant les poids $\omega_{i,j}$ pour $\text{mod}(i) = \text{m1}$ et $\text{mod}(j) = \text{m2}$. Les **densités** de ces blocs peuvent révéler l'**intensité** des connexions inter-modales :

$$d(\text{m1}, \text{m2}) \;=\; \frac{1}{|\text{m1}| \;\cdot\; |\text{m2}|} \sum_{\substack{i \in \text{m1} \\ j \in \text{m2}}} \omega_{i,j}.$$

Si $d(\text{m1}, \text{m2})$ est **beaucoup** plus élevé que d'autres combinaisons, on suspecte une **affinité** exceptionnelle entre ces deux modalités. Une étude plus détaillée de la structure (ex. un bloc dans la matrice $\mathbf{W}^{(\text{m1},\text{m2})}$ dense) peut alors faire émerger un **cluster** mixte associant m1 et m2.

Pour chaque nœud i, on peut décomposer son **degré** (ou sa **force**) en **catégories** selon la modalité de ses voisins :

$$\mathbf{deg}_i = \left(\sum_{j\,:\,\mathrm{mod}(j)=\mathrm{img}} \omega_{i,j},\ \sum_{j\,:\,\mathrm{mod}(j)=\mathrm{txt}} \omega_{i,j},\ \sum_{j\,:\,\mathrm{mod}(j)=\mathrm{aud}} \omega_{i,j},\ \ldots \right).$$

Ainsi, si un nœud i présente un vecteur $\mathbf{deg}_i$ dont **tous** les composants sont significatifs (forte connexion image, forte connexion texte, etc.), on soupçonne un "**hub** multimodal". Un regroupement de tels nœuds "hubs" (avec liens transverses) peut être un **sous-ensemble** hétérogène à forte interconnexion inter-modale.

C. Méthodes de Détection de Communautés Mixtes

Lorsque le Deep Synergy Learning (DSL) a produit un réseau dont les liaisons $\omega_{i,j}$ unissent des entités potentiellement issues de plusieurs modalités (images, textes, audio, etc.), il devient utile de recourir à des **méthodes** de partitionnement afin d'identifier des **communautés** ou sous-réseaux thématiques. De nombreux algorithmes de détection de communautés dans les graphes pondérés ont été développés, comme Girvan–Newman, Louvain, Infomap, ou le clustering spectral. La dimension multimodale amène quelques ajustements, car on peut souhaiter repérer des groupes hétérogènes englobant plusieurs flux de données.

Un premier choix consiste à appliquer la **modularité** pondérée telle que définie par Louvain ou Infomap, mais en l'adaptant aux spécificités des flux. On peut insérer un **facteur** qui valorise ou pénalise les liaisons inter-modales, par exemple en majorant les poids $\omega_{i,j}$ lorsque les entités $\mathcal{E}_i$ et $\mathcal{E}_j$ appartiennent à des modalités différentes. Cette astuce oriente l'algorithme vers une communauté plus variée. Inversement, si l'on cherche à privilégier des regroupements unimodaux, on diminue ou on neutralise les poids inter-modaux. De manière plus technique, il est possible de définir une fonction de **modularité** Q qui inclut un terme de "compatibilité modale" $\chi\big(\mathrm{mod}(i),\ \mathrm{mod}(j)\big)$. Ainsi, une liaison entre un nœud visuel et un nœud textuel reçoit un coefficient plus élevé lorsqu'on vise à constituer des groupes multimodaux.

Une deuxième approche consiste à employer un algorithme de détection de communautés **standard** sur la matrice **W** issue du SCN, puis à **filtrer** ex post les clusters identifiés pour n'en garder que ceux qui révèlent une entropie modale ou une diversité modale satisfaisante. Ce filtrage se base sur la distribution des types de nœuds dans chaque communauté. Un cluster composé uniquement d'images se voit écarté si l'on recherche absolument un mélange, alors qu'un cluster contenant plusieurs images, quelques extraits audios et des segments textuels est considéré comme "mixte" ou "hétérogène".

Dans certains scénarios, l'on souhaite extraire un **sous-graphe** restreint mais riche en hétérogénéité, plutôt qu'une partition globale. Cette recherche de motifs ou de hubs multimodaux fait appel à d'autres techniques. On peut définir une **fonction** $\mathcal{H}(\mathcal{C})$ mesurant la force interne du sous-ensemble $\mathcal{C}$ (somme des pondérations $\omega_{i,j}$ pour $i,j \in \mathcal{C}$) tout en tenant compte de la répartition modale. Ce critère $\mathcal{H}$ peut être maximisé à travers l'examen de sous-ensembles, ou encore exploré par un algorithme de recherche de motifs. Dans un cadre spectral, la factorisation de la matrice Laplacienne $\mathcal{L} = \mathbf{D} - \mathbf{W}$ révèle des ensembles de nœuds fortement connectés. On peut examiner la composition modale de ces ensembles et sélectionner ceux qui arborent une mixité marquée. D'autres approches se fondent sur le concept de "multi-view clustering", où chaque modalité est considérée comme une vue, et l'on cherche des groupes où la cohérence se retrouve sur plusieurs vues, ce qui correspond à la logique DSL quand les liaisons inter-modales se confortent mutuellement.

L'objectif commun à toutes ces techniques est de repérer des **communautés** qui reflètent un **thème** riche, associant différents canaux dans un même bloc d'entités. C'est grâce à la synergie inter-modale calculée dans le DSL que de tels groupes peuvent émerger, car la matrice **W** renferme déjà l'information des correspondances images, textes, audio. Les algorithmes standard de détection de communautés se retrouvent simplement confrontés à un graphe où les liens inter-modaux sont intégrés. Selon le choix de valorisation ou de filtrage, l'on peut forcer un regroupement plus varié ou au contraire extraire des motifs purement unimodaux. La souplesse de ce processus permet un large éventail d'analyses et d'usages, depuis la recherche de motifs spécifiques dans un gros réseau multimédia, jusqu'à la partition globale en clusters hétérogènes selon la modularité pondérée.

D. Stratégies Stochastiques ou DSL-based pour la Détection

Le **Deep Synergy Learning** (voir chap. 2.2.2) peut lui-même être **modifié** pour **encourager** la connexion inter-modale. Par exemple, dans le **score** de synergie $S(i,j)$, on ajoute un **bonus** δ lorsque $\mathrm{mod}(i) \neq \mathrm{mod}(j)$. Alors la **mise à jour** des poids

$$\omega_{i,j}(t+1) \;=\; \omega_{i,j}(t) + \eta\left[S(i,j) + \delta \cdot \mathbf{1}_{\mathrm{mod}(i)\neq\mathrm{mod}(j)} \;-\; \tau\,\omega_{i,j}(t)\right]$$

tend à **renforcer** systématiquement les **liens** multimodaux, ce qui fait **émerger** plus facilement des **clusters** hétérogènes. D'autres variantes consistent à appliquer des **inhibitions** à l'intérieur d'une même modalité pour forcer les entités à se tourner vers d'autres types de nœuds.

En combinant le DSL avec des **méthodes stochastiques** (recuit, random walk, bruit sur $\omega_{i,j}$), on peut **explorer** des partitions plus audacieuses. Le recuit simulé, par exemple, perturbe ponctuellement les pondérations pour *sortir* de minima locaux dominés par des regroupements monomodaux. Cela peut **ouvrir** la voie à des **structures** plus complexes où plusieurs modalités cohabitent au sein d'un même **cluster**.

8.7.2.3. Potentiel d'Utilisation : Indexation Automatique, Annotation, Groupement Conceptuel

Dans un environnement **multimodal**, où différents types d'objets (textes, images, audio, métadonnées, etc.) se trouvent représentés dans le **Synergistic Connection Network** (SCN) d'un **Deep Synergy Learning** (DSL), la possibilité de réaliser une **indexation** automatique, une **annotation** de contenu et un **groupement** conceptuel dynamique revêt une importance cruciale. Le **DSL**, en créant et en maintenant des **pondérations** $\omega_{i,j}$ dictées par la **synergie** $S(i,j)$, offre une **approche adaptative** qui s'étend bien au-delà des simples **classifications** ou **recherches** par mots-clés traditionnelles. L'objectif devient de **faire émerger** des liens et des regroupements entre entités, de **gérer** l'assignation d'étiquettes ou de mots-clés en continu, et de **former** des clusters conceptuels plus larges reflétant l'évolution des données.

A. Indexation Automatique

L'**indexation** vise à lier chaque entité (document texte, image, extrait audio, etc.) à un **ensemble** de mots-clés ou de **concepts** censés en **résumer** la teneur. Dans un **SCN**, l'entité "mot" (ou "concept") peut coexister au même titre qu'un **document**, le DSL **renforce** leurs liaisons ω dès que la **synergie** s'avère élevée, révélant ainsi une **affinité** ou une co-occurrence fréquente. L'idée s'incarne dans une **dynamique** locale où plus un *mot* m est pertinent pour un *document* d, plus le score $S(m,d)$ est grand, et plus $\omega_{m,d}$ s'accroît à chaque itération. Ce

mécanisme confère un cadre itératif et "vivant" à l'indexation, permettant de réviser les **associations** mots–documents dès lors que des **informations** nouvelles font évoluer S.

On considère un **ensemble** de mots $\{\mathcal{E}_m\}$ et un ensemble de documents $\{\mathcal{E}_d\}$. L'évaluation de $S(\mathcal{E}_m, \mathcal{E}_d)$ peut reposer sur :

$$S(m,d) \;=\; \mathrm{sim}(\mathbf{v}_m,\ \mathbf{v}_d),$$

où $\mathbf{v}_m$ et $\mathbf{v}_d$ sont des **représentations** (embeddings) vectorielles ; ou encore sur un **score** TF–IDF, ou une **co-occurrence** dans un grand corpus.

À chaque pas t, la **pondération** $\omega_{m,d}(t)$ suit :

$$\omega_{m,d}(t+1) = \omega_{m,d}(t) \;+\; \eta\left[S(m,d) \;-\; \tau\,\omega_{m,d}(t)\right].$$

Si la **similarité** $S(m,d)$ se maintient élevée, $\omega_{m,d}$ tend vers un **point** d'équilibre $\omega^*_{m,d} \approx S(m,d)/\tau$. Une fois la **pondération** supérieure à un **seuil** θ, on peut considérer que "le mot $\mathcal{E}_m$ indexe le document $\mathcal{E}_d$". Cette **auto-organisation** se substitue ou complète les approches statiques de "bag of words" en offrant un **système adaptatif** où l'arrivée de **nouveaux** mots ou documents réinjecte de la plasticité, permettant de **réajuster** $\omega_{m,d}$ de façon incrémentale.

L'un des principaux avantages d'un **SCN** appliqué à l'indexation réside dans sa capacité d'**adaptation** continue. Contrairement aux systèmes d'indexation traditionnels, où les relations entre mots et documents sont préétablies et figées, un **SCN** ajuste dynamiquement ses **liens** à mesure que de nouveaux documents et termes apparaissent. Cette plasticité permet au réseau de rester pertinent sans nécessiter de mise à jour manuelle constante.

L'**indexation** ne se limite pas à une simple correspondance **mot-document**, mais s'enrichit d'une **structuration dynamique** où un même mot peut référencer plusieurs documents et inversement. La force des liens entre un mot et un document est quantifiée par un **score** $\omega_{m,d}$, qui reflète leur pertinence mutuelle. Plutôt que d'imposer une relation binaire, le **SCN** capture des **niveaux de pertinence**, améliorant ainsi la précision des recherches et des associations.

Enfin, pour éviter une prolifération excessive des liaisons et assurer une **cohérence** structurelle, un **mécanisme de parsimonie** peut être intégré. Ce mécanisme filtre les connexions faibles en supprimant les liens où $\omega_{m,d} < \omega_{\min}$, ne conservant ainsi que les **associations les plus robustes**. Cette approche garantit une indexation plus stable et efficace, en ne retenant que les relations les plus significatives entre mots et documents.

B. Annotation et Étiquetage Dynamique

Au-delà de la simple association "document–mot-clé", on peut introduire dans le **SCN** des **nœuds** représentant des "catégories" ou "annotations" plus abstraites (thèmes, domaines, labels). Les documents, images ou autres entités qui entretiennent une **synergie** élevée avec un label $\mathcal{L}$ verront la pondération $\omega_{\mathcal{L},d}$ grimper, signant l'**appartenance** (ou la pertinence) de l'entité pour cette **catégorie**.

Dans un cadre strictement **non supervisé**, le DSL peut faire émerger des clusters hétérogènes où l'on peut décider ex post de "**nommer**" un cluster stable en lui attribuant un **label** explicite. Dans un cadre **semi-supervisé**, on intègre dès le départ des **nœuds** "label" dans le SCN, et on laisse la **dynamique** DSL (synergie + plasticité) **relier** ces labels aux entités appropriées.

Mathématiquement, il suffit d'introduire un **ensemble** d'entités $\{\mathcal{E}_\mathcal{L}\}$ où $\mathcal{L}$ désigne une catégorie ou un label potentiel, et de définir :

$$S(\mathcal{E}_{\mathcal{L}}, \mathcal{E}_d) = \text{sim}(\mathbf{v}_{\mathcal{L}}, \mathbf{v}_d),$$

où $\mathbf{v}_{\mathcal{L}}$ capture la **représentation** du label (ex. un vecteur sémantique). On applique ensuite la mise à jour usuelle :

$$\omega_{\mathcal{L},d}(t+1) = \omega_{\mathcal{L},d}(t) \; + \; \eta \left[S(\mathcal{L}, d) \; - \; \tau \, \omega_{\mathcal{L},d}(t) \right].$$

À la **convergence**, si $\omega_{\mathcal{L},d}$ dépasse θ, on en déduit que l'entité $\mathcal{E}_d$ est efficacement **annotée** par la catégorie $\mathcal{L}$. L'éventuel **changement** des vecteurs $\mathbf{v}_{\mathcal{L}}$ ou l'introduction d'une **nouvelle** catégorie dans le SCN fera rejaillir ces dynamiques d'adaptation, sans qu'il soit nécessaire de relancer une classification globale.

Contrairement à des systèmes d'**étiquetage** rigides reposant sur une hiérarchie fermée de labels ou un regroupement one-shot, un **SCN** sous **DSL** possède plusieurs avantages. Il peut **intégrer** de nouvelles catégories ou sous-catégories de manière dynamique, sans nécessiter de refonte complète. Il conserve les associations passées tout en les **réactualisant** en cas de conflit ou d'évolution des données. Enfin, il bénéficie d'une **unification** avec l'indexation, où un label ou un mot-clé est simplement une entité $\mathcal{E}$ supplémentaire, régie par les mêmes principes de synergie.

C. Groupement Conceptuel

Un **SCN** ne se limite pas à la relation "entité–mot" ou "entité–catégorie" puisque l'ensemble des nœuds peut inclure des *documents, des mots, des étiquettes, des métadonnées*, voire des "profils d'utilisateur" ou des "auteurs". À travers la **dynamique** DSL, de **macro-clusters** peuvent se constituer, rassemblant simultanément :

- Des mots-clés.
- Des articles/documents.
- Des catégories ou labels.
- Des entités annexes (des auteurs, des lieux, etc.).

Un tel **macro-cluster** se présente comme un **"groupe conceptuel"** où divers types de nœuds cohabitent autour d'une **cohérence** sémantique ou thématique. Au plan mathématique, on peut détecter ces clusters via la **densité** des liens ω, avec l'aide de techniques mentionnées en (8.7.2.2) pour **repérer** les sous-graphes multimodaux.

Exemple Illustratif

Imaginons un **réseau** où :

- Des *mots* comme "semi-conducteur", "processeur", "GPU" possèdent de fortes similarités textuelles ou co-occurrences,
- Des *documents* techniques mentionnent les mêmes notions,
- Un *label* "Informatique Matérielle" est lui-même corrélé à ces documents,
- Un *auteur* A revient régulièrement dans les références de ces articles.

Si la **synergie** se maintient élevée entre "semi-conducteur", "processeur", "informatique matérielle", "auteur A", "article X, Y, Z", on aboutit à un **cluster** stable, représentant un **groupe**

conceptuel autour du hardware computing. Les pondérations ω marquent chacun des liens (mot ↔ document, document ↔ auteur, label ↔ mot, etc.), donnant une **vision** globale de la **cohérence**. Cette vision s'extrait mathématiquement en repérant un sous-graphe très connecté, dont l'**entropie** modale (ou la **richesse** en entités distinctes) peut être élevée.

Le **DSL** autorise une **adaptation en continu**. Si des *nouveaux mots* (ex. "RISC-V") apparaissent et entretiennent une forte similarité sémantique avec "processeur", "semi-conducteur", les **liens** $\omega_{(\text{RISC-V,others})}$ se renforcent progressivement et peuvent faire **bifurquer** le cluster existant vers une définition plus large ou entraîner la **naissance** d'un sous-cluster (voir chap. 6.5 sur les bifurcations et agrégations).

8.7.3. Oscillations ou Confusions possibles

Dans l'architecture **multimodale** (texte, image, audio…) du DSL, il est courant qu'une même entité (par exemple, un **mot** en texte) se trouve liée à plusieurs **objets** (images, segments audio, etc.). Cette surabondance de connexions peut engendrer des **oscillations** ou des **confusions**, où la pondération $\omega_{i,j}$ varie de manière instable entre différents partenaires. Dans ce paragraphe (8.7.3), nous analysons comment de tels phénomènes de flou ou d'ambiguïté peuvent survenir et comment on peut y **remédier**.

8.7.3.1. Si un mot est surreprésenté (ex. "cat") relié à plusieurs images (chats, lions, etc.), peut générer confusion

Lorsqu'un **mot** textuel (ex. "cat") entretient une **synergie** élevée avec plusieurs **images** (chats domestiques, lions, tigres, etc.) au sein d'un **DSL** multimodal, la **dynamique** de mise à jour des ω peut exhiber des **oscillations** entre ces différentes options, traduisant une **confusion** (ou un "clignotement") dans l'attribution des liens. Cette section (8.7.3.1) illustre comment un mot **trop générique** ou **polysémique** — relié à plusieurs entités d'une autre modalité — peut déclencher une **instabilité** locale dans le **SCN**.

Considérons le **mot** "cat" apparaissant dans des documents textuels, et un ensemble d'**images** décrivant divers félins (chat domestique, lion, tigre, etc.). Si l'on dispose d'une **fonction** $S(\text{“cat”}, \text{image}^k)$ qui évalue la *pertinence* sémantique, on peut très bien observer des valeurs S *toutes* relativement élevées — par exemple :

$$S(\text{“cat”},\text{chat\_domestique.jpg}) \approx 0.85, \quad S(\text{“cat”},\text{lion.jpg}) \approx 0.82, \quad S(\text{“cat”},\text{tigre.jpg}) \approx 0.80.$$

Puisque "cat" recouvre un champ lexical large (chats domestiques, félins sauvages), la **mise à jour** des pondérations $\omega_{\text{“cat”, img}^k}$ tentera de **renforcer** parallèlement les liens vers "chat_domestique.jpg", "lion.jpg", "tigre.jpg", etc. À défaut d'un mécanisme discriminant plus **fort** (inhibition, recuit simulé, ou un ratio de similarité), on peut assister à un **phénomène** d'indécision où, au fil des itérations, la liaison $\omega_{\text{“cat”, img}^k}$ "balance" entre l'une et l'autre image.

La **règle** DSL (2.2) indique :

$$\omega_{\text{“cat”, img}^k}(t+1) = \omega_{\text{“cat”, img}^k}(t) \; + \; \eta \left[S(\text{“cat”}, \text{img}^k) \; - \; \tau \, \omega_{\text{“cat”, img}^k}(t) \right].$$

Si $S(\text{“cat”}, \text{img}^k) \approx$ *même ordre de grandeur* pour plusieurs k, la **dynamique** peut faire croître la pondération vers l'une ($\omega_{\text{cat, lion.jpg}} \uparrow$), puis, à l'itération suivante, détecter que l'autre (cat,chat_domestique.jpg) est également légitime, ajustant alors $\omega_{\text{cat, chat\_domestique.jpg}}$ à la hausse, etc. L'absence d'un **mécanisme** supplémentaire pour *sélectionner* la meilleure correspondance entraîne un cycle d'**indécision**.

Un **cas** simplifié :

- $S(\text{cat,chat\_domestique.jpg}) = 0.80$,
- $S(\text{cat,lion.jpg}) = 0.79$,

Ces valeurs très **proches** fournissent un **flux** de renforcement quasi équivalent vers deux liens $\omega_{(\text{cat, chat\_domestique})}$ et $\omega_{(\text{cat, lion})}$. Si la dynamique DSL est *sensible* (i.e. η pas trop petit, τ modéré), on peut voir la **pondération** pencher un temps pour “lion.jpg”, puis le “chat_domestique.jpg” reprendre l'avantage, provoquant des **oscillations**.

Cette **oscillation** correspond à une *confusion* du point de vue sémantique. Le mot “cat” rebondit entre *chat domestique* et *lion*, sans se “fixer” clairement. Dans un **SCN** plus large, ce phénomène peut **perturber** l'interprétation, rendant difficile de déterminer si “cat” étiquette davantage “lion.jpg” ou “chat_domestique.jpg”. Il peut également **entraîner** des oscillations dans d'autres liens connexes, notamment lorsque des mots associés à “cat” se retrouvent alternativement liés à l'image du lion ou à celle du chat domestique, sans stabilisation claire.

Le **score** $S(\text{cat}, \text{img}^k)$ est relativement **similaire** pour plusieurs images, ce qui empêche un écart net permettant de trancher définitivement. L'**absence d'inhibition** amplifie ce phénomène, car la pondération $\omega_{(\text{cat}, \cdot)}$ n'est pas contrainte, permettant une répartition sur plusieurs img^k qui peuvent alterner en dominance d'une itération à l'autre. La **polysémie** du mot “cat” en anglais accentue encore cette indécision, puisqu'il peut désigner un **chat domestique** mais aussi des **félins sauvages** comme les lions et les tigres. Sans une hiérarchie explicite (ex. distinction entre “domestic cat” et “wild cat”), la dynamique reste floue et oscille entre plusieurs interprétations possibles.

Dans un *cas-limite*, la dynamique DSL :

$$\omega_{\left(\text{cat, img}^k\right)}(t+1) = (1 - \eta\,\tau)\,\omega_{\left(\text{cat, img}^k\right)}(t) \;+\; \eta\, S\left(\text{cat}, \text{img}^k\right)$$

ne favorise pas nettement img^α sur img^β si les valeurs S respectives sont de même ordre. La **confusion** résulte du fait que ω **peut** converger vers plusieurs attracteurs locaux de quasi même énergie (ou bien *osciller* autour d'eux).

Le **score** $S(\cdot,\cdot)$ issu de la similarité *strictement* textuelle (ou un embedding sémantique global) ne discrimine pas toujours des nuances cruciales. D'un point de vue **opérationnel**, la confusion “cat” = “lion” ou “chat domestique” peut nuire à l'**annotation** ou la **recherche** d'images — on ne sait plus si le mot “cat” renvoie à un chat de maison ou à un félin sauvage.

On peut imaginer une hiérarchisation plus fine où “**domestic cat**” se distingue de “**wild cat** (lion, tigre)”. Le DSL pourrait alors dédoubler le nœud “cat” en deux entités $\mathcal{E}_{\text{cat\_domestic}}$ et $\mathcal{E}_{\text{cat\_wild}}$. Sinon, sans ce niveau de détail, la dynamique se heurte à une ambiguïté persistante.

8.7.3.2. Inhibition multimodale : on restreint la somme des liaisons “texte–images”

Dans un **DSL** (Deep Synergy Learning) à composante **multimodale**, il arrive parfois que les entités de deux modalités (par exemple, “texte” et “images”) aient tendance à **surconnecter** entre elles, formant trop de liaisons $\omega_{i,j}$ et générant ainsi une **densité** excessive ou des **confusions** dans le réseau. Pour **rééquilibrer** ces interactions, l’**inhibition multimodale** (ou *cross-modality inhibition*) instaure un **frein** à la prolifération de liens, en imposant qu’il existe une **limite** ou un **coût** si la somme de toutes les pondérations “texte–images” devient trop élevée. Cette approche peut s’avérer cruciale afin de **forcer** la sélectivité des liens et de **stabiliser** la dynamique globale du Synergistic Connection Network.

A. Motivation : Équilibrer les Modèles Multimodaux

Dans un **SCN** multimodal, les entités de type **texte** (paragraphes, mots-clés, documents) peuvent avoir de la **synergie** avec de nombreuses **images** (visuelles), aboutissant potentiellement à un **grand** nombre de liaisons $\omega_{t,v}$. Toutefois, un trop-plein de **liaisons** texte–images conduit à un **réseau** trop dense, rendant difficile la distinction des **associations** réellement pertinentes. De plus, une certaine **modalité**, comme le texte, peut **dominer** en établissant massivement des connexions, ce qui fausse l’auto-organisation en masquant les vraies correspondances fines.

L’**inhibition multimodale** propose alors d’**introduire** un mécanisme qui **pénalise** la somme globale $\sum_{t\in\mathcal{T},\, v\in\mathcal{V}} \omega_{t,v}$ (ou une variante), de sorte à **limiter** l’expansion de ces liens.

On ajoute, dans la **fonction** d’énergie ou dans la **règle** de mise à jour des ω, un **terme** qui “coûte” davantage dès lors que la **somme** $\sum_{t,v} \omega_{t,v}$ (texte–images) devient importante. Sur le plan **mathématique**, cela oriente la **dynamique** du DSL à **choisir** plus sélectivement quelques liens texte–image bien synergiques, au lieu d’élargir tous les couplages de façon indifférenciée. Le résultat **attendu** est un **SCN** plus économe en liaisons inter-modales, et donc plus lisible et plus stable.

B. Formulation Mathématique de l’Inhibition Cross-Modality

Considérons la forme générale d’une **énergie** $\mathcal{J}$ traitée par un DSL (voir chap. 2.2.2, 7.2) :

$$\mathcal{J}_0(\mathbf{\Omega}) \;=\; \underbrace{-\sum_{i,j} \omega_{i,j}\, S(i,j)}_{\text{terme “synergie”}} \;+\; \underbrace{\frac{\tau}{2}\sum_{i,j} \omega_{i,j}^2}_{\text{régularisation linéaire ou quadratique}} \quad ,$$

où $\mathbf{\Omega}$ désigne l’ensemble des $\omega_{i,j}$. Pour **inhiber** la modalité “texte–images”, on ajoute :

$$\mathcal{J}_{\text{multi}}(\mathbf{\Omega}) = \mathcal{J}_0(\mathbf{\Omega}) \;+\; \gamma_{\text{cross}}\, F\big(\{\omega_{t,v}\}\big),$$

avec $t \in \mathcal{T}$ (texte), $v \in \mathcal{V}$ (images), et $\gamma_{\text{cross}} > 0$ comme **coefficient** de pénalisation. La **fonction** F peut être :

- $F = (\sum_{t\in\mathcal{T},\, v\in\mathcal{V}} \omega_{t,v})^2$.
- $F = \sum_{t,v}\ \omega_{t,v}^2$.
- Un **mix** ou d’autres variantes (somme des valeurs absolues, etc.).

Plus la somme $\sum_{t,v} \omega_{t,v}$ s'élève, plus le **terme** $\gamma_{\text{cross}} F$ grandit, renforçant la "**dissipation**" de ces liens.

Avec la forme la plus simple $F = (\sum_{t,v} \omega_{t,v})^2$, on a :

$$\frac{\partial F}{\partial \omega_{t_0,v_0}} = 2 \sum_{t,v} \omega_{t,v},$$

de sorte que la descente de gradient injecte un **terme négatif** proportionnel à $\sum_{t,v} \omega_{t,v}$ dans la mise à jour de ω_{t_0,v_0}. Autrement dit, **plus** la somme "texte–image" est déjà grande, **plus** on va freiner l'augmentation de chaque ω_{t_0,v_0}. Ce mécanisme **réduit** la propension du réseau à créer trop de liens entre $\mathcal{T}$ et $\mathcal{V}$.

On introduit une **compétition** sur les liens "texte–images" où, si certains liens sont déjà forts, **augmenter** encore d'autres liens inter-modaux devient plus coûteux (à cause du terme $\gamma_{\text{cross}} F$). En pratique, cela **encourage** la spécialisation — seules les liaisons jugées vraiment **pertinentes** (score de synergie élevé) parviendront à maintenir un niveau $\omega_{t,v}$ important, tandis que les autres seront "**rabotées**".

C. Application dans un DSL Multimodal

Supposons qu'on gère un corpus **texte** comprenant des captions et des descriptions ainsi qu'un ensemble d'**images**. Sans **inhibition**, le DSL peut relier un texte $\mathcal{E}_t$ à un grand nombre d'images $\{\mathcal{E}_v\}$ avec des pondérations $\omega_{t,v}$ moyennement élevées, ce qui engendre une confusion pour l'**indexation**. En revanche, avec une **inhibition multimodale**, le réseau est contraint de **retenir** principalement les liens texte–image dont la **similarité** $S(t,v)$ est la plus marquée, décourageant ainsi les pondérations intermédiaires et limitant la sur-connexion.

Cette inhibition "cross-modality" n'affecte pas les **liaisons** texte–texte, les **liaisons** image–image ni les liens d'une autre modalité comme l'audio.

Seules les **connexions** entre **texte** et **images** se voient freinées lorsque leur somme collective dépasse un seuil implicite. Ainsi, on obtient un **SCN** mieux organisé où les entités texte forment un bloc structuré, les entités image un autre, et seules **certaines** paires (texte, image) "franchissent" la barrière entre modalités grâce à une synergie **réellement** forte.

D. Pistes Mathématiques et Extensions

Une variante consiste à imposer, pour **chaque** entité texte t, une **contrainte** $\sum_{v \in \mathcal{V}} \omega_{t,v} \leq \Lambda_t$. Ainsi, on limite le "nombre" total de liens (ou la somme des pondérations) partant de $\mathcal{E}_t$. Cela revient à un schéma "compétitif" local, où le texte t doit "choisir" l'image la plus alignée, plutôt que de se connecter à dix images moyennement liées. Au niveau **mathématique**, on peut réaliser cela via un terme de type $\sum_t (\sum_v \omega_{t,v} - \Lambda_t)^2$.

Il est possible de **cumuler** une **inhibition intra-modale**, qui limite $\sum_{i,j \in \text{texte}} \omega_{i,j}$ ou $\sum_{v_1,v_2 \in \text{images}} \omega_{v_1,v_2}$ afin d'éviter la formation de clusters excessivement denses dans une seule modalité, et une **inhibition cross-modality**, qui restreint $\sum_{t,v} \omega_{t,v}$ pour empêcher un déséquilibre dans les connexions entre le texte et l'image.

Cette approche mixte garantit un **équilibre** global des liaisons, limitant simultanément la surdensité texte–texte, image–image ou texte–image.

L'**ajout** d'un terme de pénalisation $\gamma_{\text{cross}} \left(\sum_{t,v} \omega_{t,v}\right)^2$ accentue la **non-convexité** de la fonction d'énergie, ce qui peut rendre la **convergence** plus délicate. On peut alors employer des heuristiques stochastiques (recuit simulé, chap. 7.3) ou un **façonnage** progressif (on augmente γ_{cross} au fil du temps) pour aider la dynamique à trouver un **arrangement** convenable.

8.7.3.3. Recuit simulé (Chap. 7.3) comme levier pour réorganiser les clusters si conflit

Dans un **SCN** (Synergistic Connection Network) piloté par un **DSL** (Deep Synergy Learning), il peut arriver que certains **conflits** de répartition se produisent lorsqu'une entité est "disputée" par plusieurs clusters, ou des sous-groupes entiers se coincent dans une configuration **sous-optimale** (mauvaise séparation, liens incorrects). Comme expliqué au **chapitre 7.3**, le **recuit simulé** apporte une **dimension stochastique** permettant de "**secouer**" le réseau et d'**échapper** à ces minima locaux. La présente section (8.7.3.3) décrit comment ce **mécanisme** s'applique en pratique pour **réorganiser** les pondérations $\omega_{i,j}$ lorsque des **conflits** surviennent ou que la configuration n'est pas satisfaisante.

A. Origine du Conflit et Besoin de Réorganisation

Un **conflit** naît souvent lorsqu'un **nœud** (ou un petit groupe de nœuds) bénéficie d'une **synergie** élevée avec plusieurs **clusters** rivaux. Au fil du temps, les liaisons $\omega_{i,j}$ entre l'entité i et le cluster C_1 augmentent, mais la synergie avec C_2 reste non négligeable. Cela entraîne l'apparition d'**oscillations**, où l'entité i bascule d'un sous-ensemble à l'autre.

Sur le plan **mathématique**, aucune configuration n'est assez "forte" pour emporter définitivement i, et le **SCN** se retrouve dans un état de tension. Dans certains cas, il converge vers un **compromis** insatisfaisant (pondérations moyennes) ou continue à **osciller** sans se fixer.

Même en l'absence de conflit direct, un **SCN** peut se stabiliser localement dans un **minimum** d'énergie $\mathcal{J}(\Omega^*)$ plus élevé que d'autres configurations possibles (voir 7.2.2.2 sur les minima locaux). Il n'existe plus d'**incrément** local suffisant pour franchir la barrière d'énergie et rejoindre une répartition **globalement** plus favorable. On parle de **verrouillage** local lorsque la mise à jour déterministe par descente de gradient ne parvient pas à "sauter" au-dessus d'un "col" d'énergie.

B. Recuit Simulé : Injection de Bruit "Tempéré"

Le **recuit simulé** (Chap. 7.3) consiste à ajouter un **terme** aléatoire $\sigma(t)\,\xi_{i,j}(t)$ à la mise à jour de chaque pondération $\omega_{i,j}$. Ainsi, l'équation

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) \;+\; \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right] \;+\; \sigma(t)\,\xi_{i,j}(t),$$

où :

1. $\sigma(t)$ est la "**température**" (amplitude du bruit) à l'itération t,
2. $\xi_{i,j}(t)$ est une **variable** de bruit (souvent gaussienne ou uniforme centrée),

autorise des *fluctuations* aléatoires autour de la logique stricte $\eta[S - \tau\,\omega]$. Quand $\sigma(t)$ est relativement **grand**, ce "bruit" peut "**briser**" les configurations trop rigides.

Le **recuit** se déroule en deux phases principales :

- **Chauffage** (ou début à température élevée) : $\sigma(t) \approx \sigma_0$ assez grand. On effectue un certain nombre d'**itérations** où les ω subissent des perturbations importantes, permettant des "sauts" entre configurations éloignées.

- **Refroidissement** : on diminue $\sigma(t)$ au fil du temps ($\sigma(t+1) < \sigma(t)$), rendant les perturbations plus modestes. On se **stabilise** alors dans un nouveau **minimum** d'énergie potentiellement plus global.

Ce procédé renvoie à l'**inspiration** de la *métallurgie,* où un métal chauffé et lentement refroidi (recuit) trouve une structure cristalline plus stable.

C. Action du Recuit en Situation de Conflit

En **phase chaude** (température élevée), les liaisons $\omega_{i,j}$ en conflit peuvent être **secouées** hors de l'équilibre local. Par exemple, si un nœud i oscille entre C_1 et C_2, le bruit peut "**casser**" momentanément la liaison ω_{i,C_1} pour encourager ω_{i,C_2} à croître, ou vice versa. L'avantage est que le réseau peut **explorer** divers rattachements, sans rester bloqué dans un compromis. Après plusieurs itérations, on amorce la **phase de refroidissement** où la dynamique redevient plus déterministe, et les **liens** les plus cohérents avec S finissent par se consolider.

Si la configuration $\mathbf{\Omega}^*$ atteint localement un "mauvais" minimum, le recuit donne la possibilité à certaines $\omega_{i,j}$ d'aller à l'encontre du gradient local. Ainsi, on franchit parfois la barrière d'énergie, aboutissant à une nouvelle configuration $\mathbf{\Omega}'$ de **meilleure** énergie $\mathcal{J}(\mathbf{\Omega}') < \mathcal{J}(\mathbf{\Omega}^*)$. En pratique, on obtient un **SCN** qui réorganise plus finement ses clusters ou résout la **confusion** de pondérations conflictuelles.

D. Mise en Œuvre Concrète

Le **chapitre 7.3** présente plusieurs **schémas** pour $\sigma(t)$:

- **Exponentiel** : $\sigma(t) = \sigma_0 \cdot \alpha^t$.
- **Logarithmique** : $\sigma(t) = \frac{\sigma_0}{\ln(t+t_0)}$.
- **Linéaire** : $\sigma(t) = \max(0, \sigma_0 - \lambda\, t)$.

Le choix dépend des préférences en termes de "vitesse" de refroidissement et de robustesse. En général, on souhaite un **démarrage** suffisamment **chaud** pour autoriser de grandes fluctuations, puis un **refroidissement** lent pour **stabiliser** solidement le réseau.

Trop de bruit (température excessive ou prolongée) peut faire "exploser" les ω aléatoirement ou empêcher toute convergence. Trop peu de bruit ne résout pas le conflit. On cherche donc un **calibrage** adéquat pour $\sigma(t)$. Des heuristiques comme le "recuit simulé adaptatif" adaptent σ selon les progrès de la convergence.

On peut cesser le recuit après un certain **nombre** d'itérations fixes (plan de recuit déterministe) ou lorsqu'un **critère** d'énergie / de variation $\| \mathbf{\Omega}(t+1) - \mathbf{\Omega}(t) \|$ tombe en dessous d'un seuil. L'**objectif** est de préserver suffisamment de temps pour **réorganiser** les clusters, tout en évitant de maintenir un bruit trop fort quand on est proche d'une bonne solution.

8.8. Apports en Applications Concrètes

Après avoir présenté, dans les sections précédentes, les principes du DSL (Deep Synergy Learning) appliqués à la **fusion multimodale** (chap. 8.1 à 8.7), il importe de souligner **comment** ces approches se déclinent dans des **applications concrètes**. Le chapitre 8.8 met l'accent sur la **mise en pratique**, on y verra comment un **SCN** (Synergistic Connection Network) gère simultanément des **données visuelles** (images, vidéos) et des **données textuelles / auditives** afin de produire des fonctions de **recherche**, d'**annotation**, ou de **reconnaissance** multimodale.

- Dans la première section (8.8.1), nous abordons l'**annotation d'images** par le **langage** (mots, phrases),
- Puis, en (8.8.2), la **reconnaissance** audio–visuelle,
- Et en (8.8.3), un exemple où l'on combine IA **symbolique** et IA **sub-symbolique** sur des flux multimodaux.

8.8.1. Annotation d'Images par le Langage

L'un des exemples d'application typiques du **DSL** multimodal concerne l'**association** entre des entités "images" et des entités "mots" ou "phrases". L'objectif est de **retrouver** quels mots ou phrases "légendent" telle image, et inversement, en s'appuyant sur la **synergie** entre ces deux modalités.

8.8.1.1. Entités "images" et entités "mots/phrases"

Dans une approche **multimodale**, il est essentiel de représenter simultanément des **entités** visuelles et des **entités** textuelles au sein d'un **SCN** (Synergistic Connection Network). Les travaux précédents (voir la référence au chapitre **2.2** pour la définition générale de la *synergie*) indiquent que chaque **entité** est caractérisée par un **embedding** propre, et qu'une fonction de **synergie** S permet de comparer deux entités, même si elles ne partagent pas la même modalité. La présente section (**8.8.1.1**) illustre spécifiquement comment introduire des **images** et des **mots/phrases** dans un **DSL** (Deep Synergy Learning), afin de construire un couplage automatique image–texte, s'apparentant à un processus de **légendage** ou d'**annotation** auto-organisée.

A. Représentation des Entités Images et Textes

La première étape consiste à définir un **ensemble** d'**images**, noté $\{\mathcal{I}_1, \dots, \mathcal{I}_N\}$, et un **ensemble** de **textes**, noté $\{\mathcal{T}_1, \dots, \mathcal{T}_M\}$. Chaque **image** $\mathcal{I}_i$ se voit associée à un **vecteur** $\mathbf{v}_i^{(\text{img})}$ (embedding visuel), par exemple généré par un **réseau CNN** ou par un **autoencodeur**. En parallèle, chaque **texte** $\mathcal{T}_m$ (qu'il s'agisse d'un **mot** unique ou d'une **phrase** entière) est transformé en un **embedding** $\mathbf{v}_m^{(\text{txt})}$, potentiellement dérivé de **GloVe**, **Word2Vec**, ou d'un **modèle** Transformer (BERT, etc.). Cette mise en correspondance permet de considérer $\mathcal{I}_i$ et $\mathcal{T}_m$ comme deux **entités** d'un même réseau, à savoir $\mathcal{E}_i^{(\text{img})}$ et $\mathcal{E}_m^{(\text{txt})}$.

B. Définition de la Synergie entre Image et Texte

La **fonction** de **synergie** $S\left(\mathcal{E}_i^{(\text{img})}, \mathcal{E}_m^{(\text{txt})}\right)$ capture la **compatibilité** sémantique entre un **embedding** visuel et un **embedding** textuel. Une manière typique de la formuler est d'utiliser la **similarité cosinus** :

$$S\left(\mathcal{E}_i^{(\text{img})}, \mathcal{E}_m^{(\text{txt})}\right) = \frac{\mathbf{v}_i^{(\text{img})} \cdot \mathbf{v}_m^{(\text{txt})}}{\| \mathbf{v}_i^{(\text{img})} \| \, \| \mathbf{v}_m^{(\text{txt})} \|} \quad .$$

Dans de nombreux cas, on peut recourir à des **réseaux** pré-entraînés (voir la référence à la section **2.2.1.2** sur la définition générale de la fonction *S*), qui projettent images et textes dans un **espace** latent commun. Les valeurs de S ainsi obtenues reflètent la proximité sémantique effective entre l'image et le fragment textuel.

C. Mise à Jour des Pondérations dans le DSL

Le **SCN** (voir la référence à la section **2.2.2.1** sur la mise à jour des **pondérations**) introduit une matrice ω reliant **entités images** et **entités textes**. Notons $\omega_{i,m}$ la **pondération** reliant l'image $\mathcal{E}_i^{(\text{img})}$ et le texte $\mathcal{E}_m^{(\text{txt})}$. Conformément au **DSL**, on dispose d'une **règle** d'évolution :

$$\omega_{i,m}(t+1) = \omega_{i,m}(t) + \eta \left[S\left(\mathcal{E}_i^{(\text{img})}, \mathcal{E}_m^{(\text{txt})}\right) - \tau \, \omega_{i,m}(t) \right] \quad .$$

Le **terme** η (taux d'apprentissage) régule la vitesse d'adaptation, tandis que τ évite que $\omega_{i,m}$ ne croisse indéfiniment. Au fil des itérations, les **paires** image–texte affichant une **synergie** S plus forte consolident $\omega_{i,m}$, tandis que les autres régressent. Il est possible, par ailleurs, de recourir à des **mécanismes** d'**inhibition** ou de **recuit simulé** (références en **8.7.3.2** et **8.7.3.3**) pour affiner la structure ou échapper à des configurations sous-optimales.

D. Interprétation en Annotation Automatique

Lorsque $\omega_{i,m}$ atteint une valeur notable, on considère que l'image $\mathcal{I}_i$ se trouve **associée** au mot (ou à la phrase) $\mathcal{T}_m$. D'un point de vue pratique, cela se traduit par un **légendage** ou une **annotation** de l'image $\mathcal{I}_i$ via le texte $\mathcal{T}_m$. À l'issue de la convergence, le **SCN** fait émerger de façon **auto-organisée** un ensemble de liaisons fortes $\omega_{i,m}$ signalant quels mots décrivent quelles images. Cette logique permet aussi d'identifier des **clusters** multimodaux où un groupe d'images semblables reliera simultanément un sous-ensemble de mots communs, révélant un **thème** transversal.

Formellement, si l'on opère une coupe $\omega_{i,m} > \theta$ pour un certain seuil θ, on isole toutes les **paires** (i, m) jugées pertinentes. Cela se conforme aux principes de **parcimonie** (voir la section **2.2.3** sur la limitation du nombre de liaisons) et aboutit à un graphe bipartite (images–textes) plus clairsemé, où chaque image se connecte uniquement aux segments textuels les plus appropriés.

E. Avantages Mathématiques et Pratiques

Le **DSL** ne requiert pas d'apprentissage supervisé strict ni de labels fixés, dans la mesure où la **synergie** se fonde sur une mesure préétablie (distance, similarité, projection multimodale). D'un point de vue théorique, cette approche garantit une grande **flexibilité** pour gérer des entités **hétérogènes**, à condition de posséder une fonction *S* capable de comparer un **embedding** visuel avec un **embedding** textuel. Les références à la section **8.7.3.2** (inhibition multimodale) ou

8.7.3.3 (recuit simulé) montrent comment stabiliser la dynamique ou résoudre les conflits si de multiples mots tentent de décrire la même image de façon indistincte, ou si plusieurs images s'attachent à un même mot trop générique.

8.8.1.2. DSL auto-organise $\omega_{\text{image,text}}$ pour identifier quels mots légendent quelles images

Dans un **scénario multimodal** mettant en jeu des **entités** de type **image** et **texte**, il est possible d'exploiter le **Deep Synergy Learning (DSL)** afin de faire émerger automatiquement la correspondance entre un contenu visuel et des tokens textuels pertinents. L'objectif est de laisser la dynamique interne du **SCN** (Synergistic Connection Network) construire une **matrice de pondérations** $\omega_{\text{image,text}}$, laquelle indique en fin de convergence les **couples** (image, mot) jugés les plus cohérents. Les considérations mathématiques et les propriétés d'**auto-organisation** présentées plus haut (voir la référence à la section **2.2.2**) s'appliquent alors pour renforcer les paires dont la **synergie** est élevée et pour affaiblir les associations moins pertinentes.

A. Représentation des Images et Mots

Les entités visuelles sont représentées par des **vecteurs** $\mathbf{x}_i^{(\text{img})} \in \mathbb{R}^{d_{\text{img}}}$, dont l'obtention repose souvent sur un **réseau** convolutif (CNN) ou sur un **encodeur** (voir la discussion sur les embeddings en section **8.8.1.1**). De même, les **entités** textuelles (mots ou tokens) sont décrites par des **embeddings** $\mathbf{x}_j^{(\text{txt})} \in \mathbb{R}^{d_{\text{txt}}}$, obtenus par Word2Vec, GloVe ou un **modèle** Transformer. Afin de quantifier la **synergie** entre une image $\mathcal{I}_i$ et un mot $\mathcal{T}_j$, on définit souvent

$$S(i,j) \;=\; \cos\text{similarity}\left(\mathbf{x}_i^{(\text{img})}, \mathbf{x}_j^{(\text{txt})}\right) \;=\; \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\| \mathbf{x}_i \| \, \| \mathbf{x}_j \|} \quad .$$

Cette formule mesure la **proximité** dans le **domaine** latent, et plus $S(i,j)$ est grand, plus l'image i et le mot j sont censés véhiculer un même contenu conceptuel.

B. Pondérations $\omega_{image,text}$ et Dynamique du DSL

La **matrice** ω reliant l'entité image $\mathcal{I}_i$ et l'entité texte $\mathcal{T}_j$ évolue selon la règle d'**auto-organisation** (voir la référence à la section **2.2.2.1**), de la forme

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right] \quad ,$$

où η est le **taux** d'apprentissage et τ un **coefficient** limitant. Si $S(i,j)$ est élevé, la pondération $\omega_{i,j}(t)$ croît progressivement, révélant l'**affinité** image–texte. Inversement, si la synergie est faible ou nulle, le terme $\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$ demeure négatif, entraînant la réduction de $\omega_{i,j}(t)$. À mesure que les itérations avancent, les paires (image, mot) considérées comme importantes s'établissent avec une **pondération** forte, tandis que les autres chutent proche de zéro.

Dans certains cas, on peut inclure un **terme** d'**inhibition** (voir la section **8.7.3.2**) pour empêcher qu'une image ne se connecte simultanément à trop de mots, ou qu'un mot ne s'étale sur trop de visuels. On peut aussi introduire un **bruit** simulant le **recuit** (voir la section **8.7.3.3**) pour échapper à des minima locaux.

C. Équations d'Énergie et Couplages Image–Texte

On peut formaliser cette mécanique en postulant une **fonction** d'énergie

$$\mathcal{J}(\omega) = -\sum_{i,j} \omega_{i,j}\, S(i,j) + \frac{\tau}{2} \sum_{i,j} \left[\omega_{i,j}\right]^2 \quad ,$$

dont la **descente** de gradient implicite recouvre la mise à jour ci-dessus. Les paires (image, mot) pour lesquelles $S(i,j)$ est élevé offrent un gain énergique, conduisant à l'**émergence** de liens $\omega_{i,j}$ importants. Si l'on rajoute un **terme** de pénalisation pour liaisons excessives, la fonction $\mathcal{J}$ devient **non convexe**, renvoyant alors à la nécessité éventuelle d'algorithmes stochastiques (comme le recuit).

D. Utilité Pratique : Légendes Automatiques et Filtrage

Après **convergence**, une **auto-organisation** se met en place où chaque image $\mathcal{I}_i$ se trouve reliée à un ensemble restreint de mots $\{\mathcal{T}_j\}$ présentant une pondération $\omega_{i,j}$ élevée. Cette relation s'interprète naturellement en termes de **légendes** ou d'**annotations**. Lorsque $\omega_{i,j}$ atteint un seuil significatif, cela indique que "l'image i est décrite par le mot j". De même, il est possible d'exploiter cette structure pour une recherche inversée, où un mot $\mathcal{T}_j$ permet d'accéder aux images $\mathcal{I}_i$ qui lui sont fortement associées, facilitant ainsi un mécanisme de **recherche** cross-modale.

En pratique, on peut fixer un **seuil** θ et ne considérer que les paires $\omega_{i,j} > \theta$. Cela restitue les **mots** pertinents pour chaque image. Les **clusters** qui se forment (voir la référence à la section **8.7.2**) montrent des sous-groupes d'images partageant un vocabulaire proche et inversement, des mots rattachés à des visuels similaires. Cette structuration constitue un **processus** de classification ou d'**indexation** non supervisé, piloté par la **synergie**.

8.8.1.3. Exemple : un dataset "Flickr8k" ou autre, observation des clusters (chats, chiens, personnes...)

Dans une configuration **multimodale** combinant des **images** et du **texte** (légendes, mots, etc.), il est souvent illustratif d'étudier des bases telles que **Flickr8k** (ou **Flickr30k**, **MS-COCO**, etc.) où chaque image comporte plusieurs **légendes** décrivant son contenu. L'objectif est de montrer comment un **SCN** (Synergistic Connection Network) auto-organise les **entités** (visuelles, textuelles) en **clusters** cohérents, par exemple "chats", "chiens", "personnes", etc. La dynamique du **DSL** (Deep Synergy Learning) assure cette classification spontanée via les **synergies** entre images, entre textes, et entre image et texte, évitant ainsi la supervision explicite.

A. Présentation du Dataset "Flickr8k"

Les ensembles d'images de **Flickr8k** comportent environ huit mille **photos** extraites de la plateforme Flickr, chacune associée à **cinq** légendes (phrases courtes décrivant la scène, le ou les objets présents). Sur le plan **multimodal**, cela crée une structure où il existe :

— Des **entités** $\{\mathcal{E}_i^{(\mathrm{img})}\}$, représentant chacune un embedding visuel (voir la section **8.8.1.1** pour la forme de l'embedding).

— Des **entités** $\{\mathcal{E}_j^{(\mathrm{txt})}\}$, correspondant aux tokens ou phrases provenant des légendes, chaque token disposant d'un embedding textuel (Word2Vec, GloVe, BERT, etc.).

B. Mise en Œuvre d'un SCN Multimodal

Pour traduire les images et les légendes en un **réseau** unique, il convient d'instituer :

- Un **embedding** visuel $\mathbf{v}_i^{(\text{img})}$ pour chaque image $\mathcal{I}_i$. Cet embedding peut résulter d'un **CNN** comme VGG ou ResNet.
- Un **embedding** textuel $\mathbf{w}_j^{(\text{txt})}$ pour chaque token (ou pour une phrase entière) issu des légendes associées.
- Une **synergie** $S\left(\mathcal{E}_i^{(\text{img})}, \mathcal{E}_j^{(\text{txt})}\right)$ capable de quantifier la **compatibilité** entre $\mathbf{v}_i^{(\text{img})}$ et $\mathbf{w}_j^{(\text{txt})}$. On peut se contenter d'une **distance** exponentielle ou d'une **similarité** cosinus entre les vecteurs d'embedding :

$$S(i,j) = \cos\text{similarity}\left(\mathbf{v}_i^{(\text{img})}, \mathbf{w}_j^{(\text{txt})}\right).$$

Parallèlement, on peut également établir :

— Des synergies **image–image** ($S(\text{img}, \text{img})$) si l'on souhaite grouper les images similaires.

— Des synergies **texte–texte** ($S(\text{txt}, \text{txt})$) si l'on veut refléter la ressemblance sémantique entre différents mots ou phrases.

C. Observation des Clusters (Chats, Chiens, Personnes, etc.)

La **dynamique** DSL (voir la section **2.2.2**) amène la matrice $\{\omega_{i,j}(t)\}$ de pondérations à évoluer au fil des itérations. L'équation de mise à jour,

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right],$$

assure le **renforcement** des paires (i, j) présentant une synergie importante (image et tokens proches), tandis que les paires moins pertinentes se voient affaiblies. À mesure que le système converge, chaque ensemble de **liens** ω se **stabilise**, les **images** affichant un **contenu** semblable, comme des chiens, se relient à des mots-clés identiques tels que "dog" ou "canine" et tendent à constituer un **cluster** commun, renforcé par des **tokens** similaires. De même, les **images** représentant des **chats**, des **personnes** ou des **scènes** sportives regroupent chacune un **noyau** de tokens récurrents, formant ainsi des **macro-clusters** homogènes.

D'un point de vue **mathématique**, on peut filtrer les liens $\omega_{i,j}$ dépassant un seuil θ. Les **sous-graphes** émergents sont alors très denses en leur sein (images très connectées aux mêmes légendes ou mots-clés), et peu connectés aux autres blocs. Cette structure traduit un **partage** sémantique. Un cluster "chiens" agrège toutes les images canines et les tokens évoquant un chien, un cluster "chats" rassemble les images félines et les mots "cat", "kitty", etc.

D. Validation sur Flickr8k

Dans une étude pratique, on charge toutes les images **Flickr8k** (environ 8000) dans le SCN, on en extrait des **embeddings** (dimension 512 ou 2048 selon le réseau), et pour chaque mot récurrent dans les légendes, on génère un embedding textuel (Word2Vec ou BERT). La fonction de synergie S combine la **similarité** cosinus (voir la section **8.8.1.2**). Après un certain nombre d'**itérations**, on note le rassemblement d'**images** appartenant au même thème, et le **DSL** fait apparaître des **macro-clusters** cohérents. Tous les liens $\omega_{i,j}$ conduisent les photos de

chiens et les tokens “dog” ou “puppy” à former un bloc, les photos de chats et le token “cat” se concentrent dans un autre bloc, etc. De même, les images montrant des **personnes** en extérieur se lient avec les mots “people”, “person”, “man”, “woman”, etc. Cette segmentation **auto-organisée** reflète la structure sémantique du corpus image–légende sans exiger de supervision externe.

8.8.2. Reconnaissance Audio–Visuelle

Au sein d’un **SCN** (Synergistic Connection Network) appliqué à la **reconnaissance audio–visuelle**, on traite simultanément des **flux vidéo** (image, mouvement) et des **flux audio** (sons, paroles, bruitages), en formant des **entités** distinctes pour chacun. L’objectif est de **repérer** les liens $\omega_{i,j}$ forts entre des segments vidéo et des segments audio qui correspondent, par exemple, à la même action ou au même événement (voix d’une personne qui parle, chien qui aboie, clap synchronisé, etc.).

8.8.2.1. Entités “segments vidéo” + entités “segments audio”

Un **SCN** (Synergistic Connection Network) appliqué à l’analyse **audio–visuelle** implique de représenter, d’une part, des **segments vidéo** et, d’autre part, des **segments audio**, chacun tenu pour une **entité** du réseau. La structure **multimodale** ainsi formée fournit un cadre permettant de lier chaque portion visuelle à la portion sonore qui lui correspond, selon une **synergie** S_{AV}. La présente section (8.8.2.1) détaille la manière de **segmenter** et de **caractériser** les flux vidéo et audio, ainsi que la définition d’une **synergie inter-modale** destinée à guider l’**auto-organisation** des pondérations $\omega_{\mathcal{V},\mathcal{A}}$.

A. Segments vidéo

La **vidéo** se découpe couramment en **sous-séquences** $\{\mathcal{V}_k\}$, par exemple des intervalles de une à deux secondes, ou des plans détectés par des méthodes de détection de coupure. Chaque **segment** $\mathcal{V}_k$ peut alors être traité comme une **entité** à part entière dans le **SCN**. Sur le plan **mathématique**, on construit un ensemble $\{\mathcal{V}_1, \dots, \mathcal{V}_{n_v}\}$. Pour décrire $\mathcal{V}_k$, on extrait un **vecteur** de **features** $\mathbf{v}_k$, qui peut provenir d’un **CNN** (mesurant la structure spatiale de l’image clé), de **motion vectors** (synthétisant la dynamique), d’un **histogramme** de couleurs ou encore d’un **embedding** sémantique (détection d’objets et de leur position).

D’un point de vue purement **interne** à la modalité vidéo, on peut définir une **synergie** $S_{\mathrm{vid}}(\mathcal{V}_p, \mathcal{V}_q)$ reflétant la similarité entre deux segments (temps similaire, objets similaires, etc.). Cela aide à relier les segments **visuellement** proches. Toutefois, dans un contexte **audio–vidéo**, on se focalise sur la **relation** entre un segment vidéo $\mathcal{V}_p$ et un segment audio $\mathcal{A}_r$.

B. Segments audio

En **parallèle**, le **flux audio** se décompose en **sous-séquences** $\{\mathcal{A}_l\}$, qui peuvent correspondre à des intervalles de une seconde, ou à des tranches homogènes détectées selon un critère acoustique (variation de spectre, changement de locuteur). Chaque entité audio $\mathcal{A}_l$ est décrite par un **vecteur** $\mathbf{a}_l$ issu de **features** classiques (MFCC, LPC) ou de **représentations** plus récentes (embeddings neuronaux basés sur des spectrogrammes).

Comme pour la vidéo, on pourrait définir $S_{\mathrm{aud}}(\mathcal{A}_l, \mathcal{A}_m)$ afin de mesurer la similarité interne à la modalité audio (deux segments contenant un son similaire). Néanmoins, dans la logique de

fusion audio–vidéo, on s'intéresse davantage aux **pondérations** $\omega_{\mathcal{V}_p,\mathcal{A}_r}$ entre un segment **vidéo** $\mathcal{V}_p$ et un segment **audio** $\mathcal{A}_r$, symbolisant leur **synchronicité** ou leur **cohérence**.

C. Synergie Inter-Modale entre Segment Vidéo et Segment Audio

Pour associer un **segment vidéo** $\mathcal{V}_p$ et un **segment audio** $\mathcal{A}_r$, on définit une **synergie** $S_{\mathrm{AV}}(\mathcal{V}_p, \mathcal{A}_r)$. Cette fonction reflète typiquement la concordance temporelle (la partie audio coïncide-t-elle avec ce moment visuel ?) et la similarité sémantique (ex. un **aboiement** détecté dans le flux audio alors qu'on **voit** un chien à l'écran). On peut schématiser :

$$S_{\mathrm{AV}}(\mathcal{V}_p, \mathcal{A}_r) = \alpha\, \delta_{\mathrm{time}}(\mathcal{V}_p, \mathcal{A}_r) + \beta\, \mathrm{Sim}_{\mathrm{embedding}}(\mathbf{v}_p, \mathbf{a}_r),$$

où $\delta_{\mathrm{time}}(\mathcal{V}_p, \mathcal{A}_r)$ mesure la superposition temporelle (combien de recouvrement entre l'intervalle vidéo et l'intervalle audio ?), tandis que $\mathrm{Sim}_{\mathrm{embedding}}(\mathbf{v}_p, \mathbf{a}_r)$ quantifie la compatibilité sémantique (ex. un extrait de spectrogramme caractéristique d'un certain son, mis en regard d'un objet reconnu dans la vidéo). Les **coefficients** α et β ajustent l'importance du facteur temporel par rapport à la similarité sémantique.

D. Création d'un SCN Multimodal

Du point de vue du **Synergistic Connection Network**, on dispose de deux **ensembles** d'entités : $\{\mathcal{V}_1, \ldots, \mathcal{V}_{n_v}\}$ pour les segments vidéo, et $\{\mathcal{A}_1, \ldots, \mathcal{A}_{n_a}\}$ pour les segments audio. Pour chaque paire $(\mathcal{V}_p, \mathcal{A}_r)$, la pondération $\omega_{\mathcal{V}_p,\mathcal{A}_r}(t)$ évolue suivant la **règle** DSL (voir la section **2.2.2** pour la mise à jour des poids) :

$$\omega_{\mathcal{V}_p,\mathcal{A}_r}(t+1) = \omega_{\mathcal{V}_p,\mathcal{A}_r}(t) + \eta \left[S_{\mathrm{AV}}(\mathcal{V}_p, \mathcal{A}_r) - \tau\, \omega_{\mathcal{V}_p,\mathcal{A}_r}(t) \right].$$

Si la **fonction** S_{AV} indique une forte correspondance (par exemple, un son de klaxon pendant que la vidéo montre une voiture), on aura $\omega_{\mathcal{V}_p,\mathcal{A}_r}$ qui s'accroît. À la fin des itérations, les **liaisons** $\omega_{\mathcal{V}_p,\mathcal{A}_r}$ établissent un couplage audio–vidéo plus ou moins dense, révélant les "sous-scènes" où l'on observe un parfait accord entre l'image et le son.

E. Interaction Interne aux Modalités

Bien qu'on s'intéresse principalement aux **corrélations** audio–visuelles, il est souvent souhaitable de laisser le **SCN** gérer également des pondérations **intra-modales**, c'est-à-dire $\omega_{\mathcal{V}_p,\mathcal{V}_q}$ entre deux segments vidéo, ou $\omega_{\mathcal{A}_l,\mathcal{A}_m}$ entre deux segments audio. Le fait de maintenir ces liens intra-modaux peut renforcer la **cohérence** globale. Si $\mathcal{V}_p$ et $\mathcal{V}_q$ montrent le même objet en plan rapproché, et si $\mathcal{A}_l$ et $\mathcal{A}_m$ partagent des similarités de sons, la dynamique du DSL peut croiser ces informations pour stabiliser des **clusters** ou sous-groupes audiovisuels plus riches.

8.8.2.2. Le SCN tisse des liens si la synchronie est forte (ex. on voit un chien aboyer, on entend un aboiement)

Dans un système **multimodal** combinant plusieurs **flux** (p. ex. un flux **visuel** et un flux **audio**), le **Synergistic Connection Network (SCN)** met en relation des **entités** issues de canaux différents, et la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ évalue la mesure selon laquelle ces entités s'**accordent** ou se **synchronisent**. Lorsqu'il existe une correspondance temporelle ou sémantique forte (tel un chien dans la vidéo et l'aboiement correspondant dans l'audio), le **DSL** (Deep Synergy

Learning) consolide les pondérations ω reliant ces entités, ce qui produit in fine un **couplage** stable reflétant un **événement** audio–visuel cohérent.

A. Principes de la synchronie multimodale

Une **synchronie** audio–visuelle repose sur deux facteurs, la **coïncidence temporelle** et la **correspondance sémantique** ou causale. Si l'on considère $\mathcal{E}_i$ une entité d'un flux **visuel** représentant un segment d'images montrant un chien ouvrant la gueule et $\mathcal{E}_j$ une entité du flux **audio** correspondant à un segment sonore présentant un aboiement, on peut définir la synergie $S(\mathcal{E}_i, \mathcal{E}_j)$ comme un mélange d'indicateurs. Cette synergie repose sur leur **chevauchement temporel** ainsi que sur leur **rapprochement sémantique**, un chien qui aboie correspondant parfaitement à un son d'aboiement. On peut penser à une **corrélation** ou un alignement temps-fréquence, ou à de l'**information mutuelle** capturant l'occurrence simultanée de l'événement visuel et du motif sonore.

B. Renforcement si la synchronie est forte

La **logique** du SCN (voir la section **2.2.2** sur la mise à jour) stipule que si $S(\mathcal{E}_i, \mathcal{E}_j)$ est **élevée**, la **pondération** $\omega_{i,j}$ reliant l'entité $\mathcal{E}_i$ à l'entité $\mathcal{E}_j$ augmente. De manière concrète, la règle s'écrit :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(\mathcal{E}_i, \mathcal{E}_j) - \tau\,\omega_{i,j}(t)\big].$$

Si, par exemple, on observe un chien remuant la gueule dans le flux **vidéo** et qu'on entend simultanément un **aboiement** dans le flux **audio**, la **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ apparaît **maximale**, menant à une hausse notable de $\omega_{i,j}$. Le SCN se "rappelle" alors que l'entité visuelle "chien aboyant" coïncide avec l'entité audio "aboiement".

Cette **sélection** adaptative des liaisons aboutit, sur le long terme, à un réseau où seuls les couplages réellement significatifs survivent ou s'amplifient — autrement dit, on ne retient que les **paires** dont la synchronie S est jugée forte.

C. Exemples concrets de tissage de liens

Lorsqu'un **chien** aboie, on se trouve en présence d'un motif **visuel** (le chien, la posture du museau ouvert) et d'un motif **sonore** (fréquences de l'aboiement). La synergie $S(\text{chienVis,aboieAud})$ atteint un niveau élevé. Dans la matrice de pondérations $\{\omega_{i,j}\}$, on renforce donc le lien $\omega_{\text{chienVis,aboieAud}}$. De la même façon, si on voit un **chat** miauler et que l'audio capture un "miaou", le SCN consolide $\omega_{\text{chatVis, miaouAud}}$. Les paires (chienVis, miaulementAud) n'ont pas de synchronie, d'où un score bas et une pondération réduite.

Cette **stratégie** ne se cantonne pas aux animaux. Toute corrélation temporelle entre un **objet** ou une **action** visible et un **son** associé (par exemple, une batte frappant une balle + un bruit de frappe) aboutit à un renforcement dans le SCN. Plus la cohérence temporelle est strictement localisée, plus le score de **similarité** ou de **corrélation** calculé dans la fonction S est important.

D. Interprétation mathématique de la fusion visuo-auditive

Dans un cadre **audio–visuel**, la synergie S peut être formalisée de multiples façons. On peut envisager :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \rho(\mathbf{x}_i, \mathbf{x}_j),$$

où $\mathbf{x}_i$ est un **embedding** pour l'entité visuelle $\mathcal{E}_i$ et $\mathbf{x}_j$ un **embedding** pour l'entité audio $\mathcal{E}_j$. Si la corrélation $\rho \approx 1$, la mise à jour de ω avantage fortement ce couplage (voir la référence **2.2.2.1**), stabilisant un **lien** entre flux visuel et flux sonore. À un niveau plus **global**, un ensemble d'entités image–son se regroupe en **cluster** multimodal quand les pondérations ω s'avèrent élevées, formant un sous-réseau cohérent (voir la référence **8.8.2.3** à venir).

E. Avantages pour l'identification d'événements multimodaux

Un tel **SCN** offre la capacité de repérer des **événements** où l'on identifie à la fois le composant **visuel** et son pendant **sonore**. L'**intégration** des signaux devient alors plus robuste. Si le visuel est trop ambigu, l'audio clarifie la scène, et inversement. Les entités "chienVis" et "aboieAud" se retrouvent couplées, illustrant la **causalité** (le chien est la source de l'aboiement). Le SCN, dans sa version DSL, n'exige pas de supervision. La simple récurrence temporelle et la similarité calculée suffisent à rendre $\omega_{\text{chienVis, aboieAud}}$ prépondérant par rapport à d'autres paires moins synchronisées.

8.8.2.3. Exemples : vidéo de scènes diverses, regroupement par type de son

Dans un **SCN** (Synergistic Connection Network) conçu pour la **fusion multimodale**, les flux **audio** et **vidéo** se décomposent en **segments** que l'on associe à des **entités** distinctes. Le **DSL** (Deep Synergy Learning) applique son principe d'**auto-organisation** aux pondérations ω reliant ces entités, de sorte à **faire émerger** des **macro-clusters** cohérents alliant la dimension visuelle (types de scènes) et la dimension sonore (types de sons). La présente section (8.8.2.3) illustre ce processus par l'exemple d'une **vidéo** complexe contenant diverses **scènes** et d'un **flux sonore** parallèlement enregistré, permettant de regrouper automatiquement "scènes de forêt + sons d'oiseaux", "scènes urbaines + bruits de circulation", etc.

A. Segmentation et Représentation des Scènes Vidéo

Un flux **vidéo** est généralement découpé en **séquences** ou **scènes** $\{\mathcal{V}_1, \ldots, \mathcal{V}_{n_v}\}$. Chaque segment $\mathcal{V}_i$ est converti en un **embedding** $\mathbf{v}_i \in \mathbb{R}^d$, que l'on peut obtenir par un **réseau** convolutif ou un **autoencodeur** appliqué à des frames ou à un agrégat de frames. On peut ainsi caractériser des contenus variés, une scène de plage, une scène urbaine, une scène de forêt, etc. D'un point de vue purement **intra-modal**, la fonction de **synergie** $S(\mathcal{V}_i, \mathcal{V}_k)$ reflète la proximité des embeddings $\mathbf{v}_i$ et $\mathbf{v}_k$ (voir la section **8.8.2.1**), ce qui permet de détecter et de regrouper les scènes visuellement proches, par exemple "paysages de montagne" ou "plans en intérieur".

B. Segmentation et Représentation des Segments Audio

Le flux **audio** se découpe pareillement en **segments** $\{\mathcal{A}_1, \ldots, \mathcal{A}_{n_a}\}$. Chaque segment $\mathcal{A}_j$ est décrit par un vecteur $\mathbf{a}_j$, dérivé par exemple de **MFCC** (coefficients cepstraux), de **spectrogrammes** ou de **réseaux** neuronaux spécifiques (embeddings audio). De la même manière, la **synergie** $S(\mathcal{A}_j, \mathcal{A}_m)$ indique la similarité entre deux segments sonores, révélant des types de sons comparables comme les bruits de trafic, les chants d'oiseaux, les voix humaines ou la musique instrumentale. Cette évaluation intra-modale conduit à la formation de **clusters** de segments audio, chacun représentant un registre acoustique homogène.

C. Couplage Audio–Vidéo et Mise en Œuvre SCN

Au-delà de l'**analyse** séparée (vidéo vs. audio), l'intérêt d'un **SCN** multimodal réside dans la capacité à **relier** un segment vidéo $\mathcal{V}_i$ à un segment audio $\mathcal{A}_j$ par une pondération $\omega_{i,j}$. Cette

pondération évolue sous l'effet de la **synergie** $S(\mathcal{V}_i, \mathcal{A}_j)$. La fonction S capture à la fois la **concordance** temporelle (les deux segments se chevauchent dans le temps) et la **correspondance** sémantique (le contenu sonore correspond-il à la scène ?). Une formulation possible :

$$S_{\mathrm{AV}}(\mathcal{V}_i, \mathcal{A}_j) = \alpha\, \delta_{\mathrm{time}}(\mathcal{V}_i, \mathcal{A}_j) + \beta\, \mathrm{Sim}(\mathbf{v}_i, \mathbf{a}_j),$$

où δ_{time} identifie si les segments se superposent (ou se succèdent) d'un point de vue chronologique, et $\mathrm{Sim}(\mathbf{v}_i, \mathbf{a}_j)$ mesure la cohérence sémantique (par ex. spectrogramme d'aboiement associé à la présence d'un chien dans la vidéo). Dans le **SCN**, la pondération $\omega_{i,j}(t)$ se met à jour suivant la **règle** DSL :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S_{\mathrm{AV}}(\mathcal{V}_i, \mathcal{A}_j) - \tau\, \omega_{i,j}(t)\big].$$

Lorsque la synergie est forte (ex. on voit des **oiseaux** dans la vidéo et on entend des **chants d'oiseaux** au même moment), $\omega_{i,j}$ grimpe, entraînant la création d'un **lien** solide entre la scène visuelle correspondante et le son idoine.

D. Émergence de Macro-Clusters Multimodaux

Les pondérations ω ainsi établies ne relient pas seulement "scène vidéo–scène audio", mais exploitent également les liens **intra-modal** (vidéo–vidéo, audio–audio). On obtient in fine un **réseau** structuré où les segments vidéo **similaires** forment des **clusters** internes, tels que des groupes représentant des "scènes de forêt" ou des "scènes urbaines". De la même manière, les segments audio **similaires** se regroupent en catégories distinctes, comme un cluster dédié aux "voix humaines" ou aux "moteurs de voiture". Enfin, les liaisons **cross-modales**, reliant des segments visuels et sonores synchrones ou cohérents, engendrent des **macro-clusters** multimodaux intégrant, par exemple, des "scènes de forêt associées aux sons d'oiseaux" ou des "séquences urbaines combinées aux bruits de circulation".

Au sens **mathématique**, ce phénomène signifie que le **DSL** repère les configurations de pondérations $\{\omega_{i,j}\}$ minimisant l'énergie globale, de telle sorte que de grands ensembles d'entités (audio et vidéo) partagent de fortes liaisons internes s'ils relèvent d'un **contexte** homogène.

E. Exemple Concret de Fusions de Flux

Imaginons qu'un système traite une **longue vidéo** filmée dans plusieurs lieux, avec une scène en **forêt** où apparaissent des arbres et des animaux, une scène en **ville** montrant des immeubles et du trafic, ainsi que diverses **transitions** entre ces environnements. Parallèlement, l'audio comporte des **segments** de chants d'oiseaux, des **bruits** de voiture, et parfois de la **musique**. Lors de l'**auto-organisation**, les segments vidéo représentant une forêt se renforcent mutuellement en raison de similarités en termes de couleurs et de textures, tandis que les segments audio contenant des chants d'oiseaux se regroupent entre eux. Les liens **cross-modaux** deviennent plus marqués sur la période correspondante, établissant une association entre la forêt et le chant d'oiseaux, ce qui aboutit à un **macro-cluster** intégrant ces deux types d'entités. De la même manière, les segments vidéo représentant une ville entrent en synergie avec les segments audio caractérisés par des klaxons ou des bruits de foule.

On voit alors **apparaître** plusieurs **macro-clusters**, tels qu'un bloc "nature" et un bloc "urbain". Une **musique** de fond, si elle se propage dans différentes scènes, peut se lier un peu à plusieurs segments vidéo, sans former un cluster trop spécifique — ce qui dépendra de la force de la synergie calculée.

8.8.3. IA Symbolique–Sub-Symbolique en Multimodal

Dans une approche **multimodale**, un **SCN** (Synergistic Connection Network) est déjà appelé à gérer diverses sources (image, audio, texte, etc.). Lorsqu'on ajoute en plus des **règles logiques** (dimension **symbolique**), la complexité du réseau augmente d'un cran. Au-delà des simples embeddings sub-symboliques (vision/audio), on introduit également un niveau **logique** (ex. "if meowing then cat"). Le **DSL** (Deep Synergy Learning) doit alors organiser un **triple couplage** entre l'**image**, représentée par des vecteurs CNN ou des features extraites d'une scène, l'**audio**, analysé sous forme de spectrogrammes ou d'embeddings acoustiques, et les **règles logiques**, qui permettent d'établir des connexions sémantiques, par exemple en associant le concept "cat" à la détection d'un "meow" ou en validant une règle comme "milk if dairy-product?".

8.8.3.1. Lorsqu'on ajoute des règles logiques (ex. "if meowing then cat" + embeddings), le SCN gère un triple couplage image–audio–règle

Le **DSL** (Deep Synergy Learning), appliqué à un **SCN** (Synergistic Connection Network), peut recevoir comme **entrées** non seulement des **entités** d'origine sub-symbolique (telles que des embeddings d'images ou d'audio) mais également des **règles** de nature **logique** (entités symboliques). L'exemple d'une règle "if meowing then cat" illustre comment, dans une configuration **multimodale**, le **SCN** peut exploiter simultanément l'**information** visuelle (un chat détecté dans l'image), l'**information** sonore (un miaulement capturé par l'audio) et la **règle** logique dictant un lien causal ("si j'entends meow, alors il y a un chat"). La présente section (8.8.3.1) décrit :

- La **représentation** symbolique d'une règle,
- Le **triple couplage** entre image, audio et logique,
- L'**émergence** d'un concept multimodal plus solide ("cat") grâce à cette triangulation.

A. Représentation Symbolique et Synergie

Bien qu'un **SCN** traite essentiellement des **embeddings** sub-symboliques (par ex. des vecteurs issus de CNN ou des spectrogrammes neuronaux), on peut y **inclure** des **entités** symboliques. Une **règle** telle que "if meowing then cat" se voit alors modélisée par un **nœud** $\mathcal{E}_{\text{rule}}$ qui possède lui aussi un **vecteur** ou un **descripteur** dans un espace symbolique ou sémantique. De façon formelle, on attribue à cette entité $\mathcal{E}_{\text{rule}}$ un "embedding logique" $\mathbf{r} \in \mathbb{R}^d$, ou tout au moins un identifiant que l'on peut mettre en correspondance avec des motifs d'**audio** ou des **objets** visuels.

La **synergie** $S(\mathcal{E}_{\text{logic}}, \mathcal{E}_{\text{audio}})$ ou $S(\mathcal{E}_{\text{logic}}, \mathcal{E}_{\text{image}})$ reflète le degré de correspondance entre la **règle** symbolique et le **signal** perçu. Par exemple, si la règle "if meowing then cat" s'applique, et qu'on détecte un **miaulement** sur le flux audio, la **compatibilité** est forte. De même, si l'image suggère clairement la forme d'un chat, la **règle** peut valider cette hypothèse via un score positif. Dans le **SCN**, on met à jour $\omega_{\text{rule, audio}}$ ou $\omega_{\text{rule, image}}$ conformément à la règle de mise à jour (voir la section **2.2.2**).

B. Couplage Triple : Image–Audio–Règle

Lorsqu'il existe trois **entités** $\mathcal{E}_{\text{img}}$, $\mathcal{E}_{\text{aud}}$, $\mathcal{E}_{\text{rule}}$, chacune peut entretenir une **synergie** avec les deux autres. La mesure $S(\text{img}, \text{aud})$ évalue l'**accord** entre l'image, par exemple un chat visible à l'écran, et le son, comme un **miaulement** correspondant. La synergie $S(\text{img}, \text{rule})$ détermine dans quelle mesure le **visuel** est en phase avec le concept "chat", sous-jacent à une règle du type "if meowing then cat". Enfin, la relation $S(\text{aud}, \text{rule})$ valide si le segment sonore détecté correspond à la condition évoquée dans la règle, confirmant ainsi une **cohérence tri-modale**.

En un sens, on aboutit à un **triangle** de pondérations dans le SCN avec $\omega_{\text{img,aud}}$, $\omega_{\text{img,rule}}$, $\omega_{\text{aud,rule}}$. Si, par exemple, on constate simultanément un miaulement sur l'audio, la présence d'un chat sur la vidéo et la règle "if meowing then cat" dans le **réseau**, chaque lien se trouve **renforcé**, ce qui accroît la **convergence** vers un **macro-nœud** ou **cluster** incarnant la "situation" où un chat produit un miaulement conformément à la règle logique.

C. Stabilité des Concepts Multimodaux

En **intégrant** un niveau symbolique (les règles) dans le **SCN**, on permet à la **logique** de s'ajouter aux indicateurs sub-symboliques (image, audio). On peut formuler mathématiquement la synergie globale comme :

$$S_{\text{global}}(\text{img}, \text{aud}, \text{rule}) = \alpha\, S(\text{img}, \text{aud}) + \beta\, S(\text{img}, \text{rule}) + \gamma\, S(\text{aud}, \text{rule}).$$

Lorsque la somme $\alpha\, S(\text{img}, \text{aud}) + \beta\, S(\text{img}, \text{rule}) + \gamma\, S(\text{aud}, \text{rule})$ est **élevée**, le **SCN** renforce les liens $\{\omega_{\text{img,aud}}, \omega_{\text{img,rule}}, \omega_{\text{aud,rule}}\}$, ce qui aboutit à un **concept** stable (ex. le "cluster" correspondant à un chat). On dit alors que la **cohérence** du concept "chat" est alimentée par la fois par l'image (l'apparence du chat), l'audio (le miaulement), et la **règle** "if meowing then cat".

Cette **triangulation** favorise une convergence plus rapide et plus sûre qu'un couplage seulement audio–image, car la **règle** logique agit comme un **tuteur** symbolique, clarifiant la relation "un miaulement implique un chat". Lorsque ce triple couplage est validé, le DSL **stabilise** fortement le noeud "cat" et les pondérations afférentes, constituant ainsi un **cluster** multimodal conceptuellement robuste.

D. Avantage : l'Auto-Organisation produit des Concepts plus Stables

La présence, dans le **SCN**, d'un **niveau** symbolique n'est pas nécessairement complexe sur le plan numérique. On peut se contenter d'allouer un **embedding** $\mathbf{r}_k$ pour chaque **règle** $\mathcal{E}_{\text{rule}^k}$, et d'un mécanisme de **match** pour lister les conditions (miaulement, aboiement, etc.) et leurs conséquences (chat, chien). L'**auto-organisation** du DSL fait le reste. On retiendra les points saillants :

- **Convergence plus facile** : la connaissance symbolique "if meowing then cat" complète les indices sub-symboliques (on voit un chat, on entend un "meow"). Les liaisons ω se renforcent plus globalement, permettant au SCN de former un **concept** "chat" de manière plus évidente.

- **Expliquabilité** : une fois convergé, on peut repérer le **rôle** de la règle dans la formation du cluster. L'analyse du SCN indique que les liens image–règle, audio–règle, image–audio se sont mutuellement validés.

- **Extension** : on peut ajouter toute une série de règles analogues (“if barking then dog”, “if crowing then rooster”), ou des énoncés plus complexes décrivant la logique de la scène (par ex. “if flapping wings then bird”).

8.8.3.2. Avantage : l’auto-organisation fait émerger des concepts multimodaux plus stables

La **fusion multimodale** dans un **SCN** (Synergistic Connection Network) piloté par un **DSL** (Deep Synergy Learning) présente un avantage déterminant en permettant de **solidifier** l’émergence de **concepts** ou *macro-nœuds* regroupant plusieurs entités grâce au concours simultané de différents **canaux** visuels, audio, textuels, etc. Lorsque plusieurs modalités corroborent un même groupement, la **synergie** s’en trouve renforcée et la **stabilité** d’un cluster multimodal augmente. Le présent exposé (8.8.3.2) souligne les mécanismes mathématiques et pratiques qui rendent les concepts plus robustes lorsqu’ils reposent sur des indices diversifiés.

A. Rappel du Principe d’Auto-organisation dans un Cadre Multimodal

Un **SCN** multimodal comprend des entités $\mathcal{E}_i$ issues de différentes **sources** (images, sons, textes, éventuellement règles symboliques). La **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ mesure leur **compatibilité** ou **complémentarité**. Dans un cas simplifié, on pourrait définir :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \alpha\, S_{\text{vis}}(i,j) + \beta\, S_{\text{text}}(i,j) + \gamma\, S_{\text{aud}}(i,j).$$

La **pondération** $\omega_{i,j}$ entre $\mathcal{E}_i$ et $\mathcal{E}_j$ suit la mise à jour DSL :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\, \omega_{i,j}(t)\big].$$

Ce mécanisme **auto-organisé** renforce les **paires** (ou les ensembles) d’entités qui affichent une **synergie** élevée, ce qui aboutit à des **clusters** de nœuds ayant un score important. Dans un cadre **multimodal**, si plusieurs canaux confirment la même **association** (ex. un chat visible et un “meow” audible), la **somme** des composantes de $S(i,j)$ favorise une pondération $\omega_{i,j}$ d’autant plus grande.

B. Stabilité Accrue des Concepts Multimodaux

Lorsqu’un **concept** émerge de la conjonction de plusieurs canaux, ce concept se montre **plus stable** qu’un concept ne s’appuyant que sur un seul flux. Supposons un **cluster** combinant $\mathcal{E}_{\text{image}}$ (un embedding visuel), $\mathcal{E}_{\text{audio}}$ (un extrait sonore) et $\mathcal{E}_{\text{text}}$ (un fragment linguistique) . Même si l’un des canaux est **perturbé** (bruit sur l’audio), les deux autres canaux (image et texte) maintiennent la cohésion. D’un point de vue **mathématique**, la pondération ω reste soutenue par la synergie cumulée $\alpha\, S_{\text{vis}} + \beta\, S_{\text{text}} + \cdots$. Cela évite qu’un **bruit** localisé (par ex. un enregistrement audio pollué) ne dissolve entièrement le cluster.

La **mise à jour** $\omega(t+1) = \omega(t) + \eta\,[S - \tau\,\omega]$ bénéficie particulièrement de la **somme** des synergies issues des canaux. Si l’entité $\mathcal{E}_i$ correspond simultanément à des indices visuels et textuels cohérents, la **valeur** de $S(i,j)$ (et donc la force du gradient) est plus grande que dans un cadre *mono-modal*. Ainsi, la pondération $\omega_{i,j}$ atteint plus rapidement et plus fermement un **point** d’équilibre élevé. En conséquence, la **configuration** finale (graphe de pondérations ω) présente des “puits” d’énergie plus profonds, rendant les clusters multimodaux plus **difficiles** à déstabiliser.

Sur un plan **combinatoire**, la présence de multiples canaux pourrait suggérer une complexification, mais en réalité, les **ambiguïtés** se réduisent si un concept doit **convaincre** plusieurs modalités en même temps. Un "faux alignement" qui semblerait plausible en vision peut être **démenti** par l'audio ou le texte. De ce fait, seul un alignement **multi-canal** récoltant l'approbation de tous les canaux parvient à se stabiliser. Cette intégration agit donc comme un **filtre** naturel pour éliminer les configurations ambiguës, entraînant un **concept** plus net (ex. le "chien qui aboie" est soutenu par l'aspect canin et la tonalité d'aboiement, écartant un "chat" ou autre animal).

C. Impact Mathématique sur l'Émergence des Concepts

Dans la logique (chap. 7.2) où la descente d'énergie $\nabla\mathcal{J}(\omega)$ oriente la mise à jour ω, la **partie** "gain" de $\mathcal{J}$ inclut le produit $\omega_{i,j} \times S(i,j)$ (qui s'additionne pour chaque couple). Lorsque la synergie $S(i,j)$ comporte plusieurs **composantes** issues de canaux distincts, la **somme** est d'autant plus élevée si tous les canaux **convergent**. Le **cluster** résultant dispose alors d'un **gain** plus important, le rendant énergétiquement plus **stable** et moins enclin à être perturbé par de petites fluctuations.

Les analyses de **dynamique** dans le SCN montrent que des **clusters** forts (pondérations élevées et mutuellement soutenues) correspondent à des **attracteurs** de la mise à jour $\omega(t+1) = \omega(t) + \cdots$. Dans un cadre multimodal, un "concept" drainant la cohérence de plusieurs flux, s'impose comme un attracteur **plus** stable (ou de gradient plus prononcé). Cela se formalise par un **gradient** où la "force" s'appliquant en faveur d'un concept prend en compte la **somme** $\alpha\, S_{\text{vis}} + \beta\, S_{\text{text}} + \cdots$, creusant un **minimum local** plus profond sur l'espace des pondérations.

8.9. Aspects Évolutifs et Temps Réel

Au sein d'un **DSL** (Deep Synergy Learning) multimodal, il est fréquent que les **données** (images, sons, texte) arrivent de façon **continue** et **évolutive** : on parle alors de **flux** (streams) plutôt que de lots statiques. Dans un tel contexte, le **SCN** (Synergistic Connection Network) doit s'adapter en **temps réel**, en absorbant progressivement de nouveaux segments vidéo ou audio, tout en conservant la structure synergique préexistante. Les sections 8.9.1 à 8.9.3 décrivent comment gérer et visualiser cette dynamique évolutive dans un cadre réellement **multimodal**.

8.9.1. Flux Multimodal en Continu

La **notion** de flux multimodal implique qu'au fil du temps t, on reçoit en **continu** des **extraits** (frames) vidéo, des **segments** audios et éventuellement des **sous-titres** ou des **annotations** textuelles. Le **SCN** se trouve alors confronté à la tâche de **fusionner** ou de **clusteriser** ces nouvelles entités dans la matrice $\{\omega_{i,j}\}$, sans redémarrer l'apprentissage depuis zéro. Les mathématiques du DSL (mise à jour itérative de ω) doivent donc être étendues pour accueillir l'**arrivée incrementale** de données, comme développé en Chap. 7 (sections 7.6 sur l'adaptation continue).

8.9.1.1. Scénarios : streaming vidéo + audio + sous-titres

Dans de nombreux systèmes **multimodaux**, il n'existe pas un lot unique de données à traiter hors ligne, mais plutôt un **flux continu** de contenus (par exemple, un **streaming** issu d'une conférence, d'un cours en ligne ou d'une vidéo en direct) où les canaux **visuels**, **sonores** et **textuels** parviennent simultanément. La présente section (8.9.1.1) envisage un scénario concret où un **flux** vidéo fourni en temps réel, accompagné d'un **flux** audio synchrone et de **sous-titres** ou annotations textuelles intermittentes. L'objectif du **SCN** (Synergistic Connection Network) est d'**intégrer** ces nouvelles entités au fur et à mesure de leur arrivée et de **maintenir** une auto-organisation cohérente permettant de faire émerger des **clusters** (macro-nœuds) multimodaux en temps réel.

A. Exemple de flux continu

Un **service** de streaming peut recevoir :

- Des **frames vidéo** successives $\{F_t\}$,
- Un **flux audio** $\{A_t\}$ enregistré en parallèle,
- Des **sous-titres** (ou tout autre texte) $\{T_t\}$ arrivant par blocs ou au fil de la transcription,
- Des **métadonnées** supplémentaires (type chapitrage, indicateurs d'interactivité, etc.).

Dans un **SCN** (voir la section **2.2.1** sur la définition d'entités), chaque **entité** $\mathcal{E}$ correspond à un segment visuel, une portion audio ou un bloc de texte. L'ensemble de ces entités $\{\mathcal{E}_n\}_{n\in\mathbb{N}}$ grandit au fil du temps. À chaque **pas**, on insère dans le SCN la nouvelle **entité** provenant du flux vidéo $\mathcal{E}_{(\text{vid},t)}$, celle issue du flux audio $\mathcal{E}_{(\text{aud},t)}$ et éventuellement un **chunk** textuel $\mathcal{E}_{(\text{txt},u)}$.

La question centrale est de savoir comment calculer les **pondérations** ω reliant ces nouvelles entités aux entités déjà présentes, et d'éviter que le **coût** de mise à jour ne devienne prohibitif à mesure que le flux s'allonge.

B. Arrivée Incrémentale et Mise à Jour

À chaque itération, au temps τ (pour ne pas confondre avec le τ de décroissance), on reçoit une entité $\mathcal{E}_{\text{new}}$. Dans un **SCN**, on doit alors **connecter** $\mathcal{E}_{\text{new}}$ à un sous-ensemble d'entités déjà existantes. On calcule la **synergie** :

$$S(\mathcal{E}_{\text{new}}, \mathcal{E}_j) = \text{f}(\mathbf{x}_{\text{new}}, \mathbf{x}_j),$$

où $\mathbf{x}_{\text{new}}$ et $\mathbf{x}_j$ sont les embeddings ou les signatures associées à $\mathcal{E}_{\text{new}}$ et $\mathcal{E}_j$. Concrètement, si la nouvelle entité est un frame vidéo, on compare son **embedding visuel** (réseau CNN, par ex.) avec ceux des frames vidéo passés et avec les signatures audio/texte temporellement proches. Cela entraîne la mise à jour :

$$\omega_{\text{new},\,j}(t+1) = \omega_{\text{new},\,j}(t) + \eta\big[S(\mathcal{E}_{\text{new}}, \mathcal{E}_j) - \tau\,\omega_{\text{new},\,j}(t)\big].$$

Dans un streaming de longue durée, le nombre d'entités $\{\mathcal{E}_j\}$ peut croître de façon considérable, rendant le coût $O(n)$ (ou $O(n^2)$ avec la mise à jour croisée) trop grand. Plusieurs **stratégies** limitent l'explosion :

- **Fenêtre glissante** : on ne compare la nouvelle entité qu'aux entités proches dans le temps $\{j \mid |t(\text{new}) - t(j)| \leq \Delta\}$.
- **k-NN** : on recherche les k entités les plus proches en embedding, plutôt que de tout comparer.
- **Parcellisation** : on gère des sous-réseaux ou on agrège les entités plus anciennes en clusters, évitant un survol exhaustif de tous les nœuds (voir chap. 7.6).

C. Problème de Synchronisation

Dans le flux **audio–vidéo**, un **frame** $\mathcal{E}_{(\text{vid},t)}$ correspond au temps t, tandis que le segment audio $\mathcal{E}_{(\text{aud},t')}$ peut se situer un demi-frame plus tard si la synchronisation n'est pas parfaite. Les **sous-titres** $\mathcal{E}_{(\text{txt},u)}$ arrivent souvent par blocs couvrant plusieurs secondes, avec un léger décalage. Le **SCN** doit donc **adapter** la définition de la synergie pour accepter un battement temporel :

$$S(\mathcal{E}_{(\text{vid},t)}, \mathcal{E}_{(\text{aud},t')}) = \text{g}(\,\mathbf{x}_{\text{vid},t}, \mathbf{x}_{\text{aud},t'})\,\delta_{\text{time}}(t, t'),$$

où $\delta_{\text{time}}(t, t')$ rend la synergie presque nulle si $|t - t'|$ est trop grand, ou la pénalise proportionnellement à l'écart. On peut également restreindre $\omega_{\text{vid, aud}}$ à des paires (t, t') jugées proches dans la chronologie.

Lorsque des **chunks** textuels couvrent plusieurs secondes (ex. 2 à 5 s), on associe chaque chunk $\mathcal{E}_{(\text{txt},u)}$ à une plage $[t_0, t_1]$. On évalue la synergie $S(\text{vid}, \text{txt})$ ou $S(\text{aud}, \text{txt})$ en considérant si le segment vidéo (ou audio) intersecte temporellement la fenêtre $[t_0, t_1]$. À nouveau, cela permet de ne pas comparer tout à tout, mais seulement ce qui se chevauche.

D. Implications sur la Synergie Multimodale

Grâce à la **gestion** continue des entités, le **SCN** perçoit au **fil** du temps la corrélation entre frames vidéo et segments audio ou textuels coïncidant dans la chronologie. Les **pondérations**

ω se renforcent lorsqu'une concordance survient (p. ex. une personne s'exprimant à l'écran correspond au segment audio de la même voix). L'auto-organisation fonctionne ainsi **en temps réel**, au lieu d'être un traitement batch a posteriori.

La **fréquence** d'arrivée des frames (p. ex. 30 par seconde) peut être très élevée, rendant la mise à jour ω coûteuse s'il faut sonder la synergie avec toutes les entités passées. Des méthodes d'approximation (voisinage, fenêtre glissante, chap. 7.6) atténuent le problème. Sur le plan mathématique, on suppose souvent un **incrément** Δt fixant la durée de validité d'une comparaison, ou on projette l'entité $\mathbf{x}_{\text{vid},t}$ dans un cluster intermédiaire pour limiter la granularité.

8.9.1.2. Mise à jour incrémentale (Chap. 7.6), insertion progressive de frames, segments audios, etc.

Dans un contexte **multimodal** (vision, audio, texte) où les données **arrivent en continu**, un **SCN** (Synergistic Connection Network) basé sur un **DSL** (Deep Synergy Learning) se doit de gérer l'**insertion** de nouvelles **entités** (frames vidéo, segments audio, sous-titres, etc.) sans tout reprendre à zéro. La section **7.6** souligne précisément les mécanismes d'**insertion progressive** et de **mise à jour incrémentale** des pondérations ω. Appliqués à un scénario de **streaming**, ces méthodes permettent de préserver la structure **auto-organisée** du réseau, tout en intégrant les **nouveaux** flux à la volée.

A. Principes de la mise à jour incrémentale

Lorsque l'on reçoit une **nouvelle** entité $\mathcal{E}_{n+1}$, par exemple un frame vidéo $\mathcal{E}_{(\text{vid},\tau)}$ ou un chunk audio $\mathcal{E}_{(\text{aud},\tau\prime)}$, le **SCN** s'agrandit. La matrice ω passe de taille $n \times n$ à $(n+1) \times (n+1)$. Pour tout noeud $\mathcal{E}_j$ déjà existant, on ajoute deux nouvelles lignes/colonnes $\omega_{(n+1),j}, \omega_{j,(n+1)}$, que l'on peut initialiser à zéro ou à une valeur aléatoire de faible amplitude.

D'un point de vue **mathématique**, l'opération ne nécessite pas de **recalcul** exhaustif sur l'ensemble des paires passées. Les pondérations $\omega_{i,j}$ reliant entre elles des entités déjà présentes (indices $1 \ldots n$) demeurent **inchangées**. On se borne à évaluer la **synergie** $S(\mathcal{E}_{n+1}, \mathcal{E}_j)$ pour un **voisinage** restreint (ou complet, si la charge de calcul le permet). On met alors à jour :

$$\omega_{(n+1),\,j}(t+1) = \omega_{(n+1),\,j}(t) + \eta\big[S(\mathcal{E}_{n+1},\, \mathcal{E}_j) - \tau\, \omega_{(n+1),\,j}(t)\big].$$

Cette formule reflète la **descente locale** (ou l'auto-organisation) évoquée au chapitre **7.2**, mais appliquée uniquement au **nouveau** nœud $\mathcal{E}_{n+1}$ et aux noeuds de son voisinage $\{j\}$.

Pour améliorer l'**efficacité**, la comparaison $S(\mathcal{E}_{n+1}, \mathcal{E}_j)$ est restreinte à un **ensemble** $\mathrm{N}(n+1) \subseteq \{1, \ldots, n\}$ de taille contrôlée, comprenant par exemple les k plus proches voisins ou les entités partageant un intervalle temporel similaire. Par ailleurs, les **pondérations** $\omega_{i,j}$ entre les nœuds déjà existants ($i, j \leq n$) restent **inchangées**, ou bien elles sont mises à jour via la routine standard, sans nécessiter un recalcul complet du réseau.

Le **cluster** global déjà formé se **stabilise** localement, tandis que le nouvel arrivant "trouve sa place" en fonction de la **synergie** calculée avec son voisinage.

Cette approche incrémentale assure que la **topologie** et la **répartition** des macro-clusters déjà formés dans le **SCN** ne soient pas totalement **réinitialisées** à l'arrivée d'un nouvel élément.

Seule la **région** du graphe environnant le noeud $\mathcal{E}_{n+1}$ se réadapte, permettant un **apprentissage continu** où la structure globale se **déploie** graduellement sur l'axe du temps (voir chap. **7.6**).

B. Application à la Fusion Audio-Visuelle

Dans un **stream** combinant image et son, chaque **frame** vidéo (30 images/s) et chaque **bloc** audio (échantillonné différemment) arrive comme une **entité** $\mathcal{E}_{n+1}$. À son arrivée, on calcule la **synergie** $S(\mathcal{E}_{n+1}, \mathcal{E}_j)$ avec, par exemple, les entités $\mathcal{E}_j$ de la même tranche temporelle. Si un segment audio $\mathcal{E}_{n+1}^{(\text{aud})}$ ressemble fortement à un autre (profil spectral identique) ou coïncide temporellement avec des frames vidéo pertinents, $\omega_{(n+1),j}$ se voit **renforcé**. Sinon, la pondération demeure faible.

Outre l'intra-modal (audio-audio, vidéo-vidéo), on peut aussi croiser $\mathcal{E}_{n+1}^{(\text{aud})}$ avec des frames vidéo $\mathcal{E}_k^{(\text{vid})}$ si $\mathcal{E}_k$ tombe dans la même fenêtre temporelle. Le **SCN** (qui gère la **synergie** cross-modale, chap. 8.8) accueille ainsi une **connexion** éventuelle $\omega_{(n+1),k}$ plus élevée si un aboiement sonore correspond à l'apparition d'un chien à l'écran.

C. Mémoire et Approche Incrémentale

Quand on ajoute la nouvelle entité $\mathcal{E}_{n+1}$, on inscrit $\omega_{(n+1),j}$ dans la matrice, avec initialisation $\omega_{(n+1),j} \approx 0$. Ensuite, au moment $t \to t+1$,

$$\omega_{(n+1),j}(t+1) = \omega_{(n+1),j}(t) + \eta\big[S(\mathcal{E}_{n+1}, \mathcal{E}_j) - \tau\, \omega_{(n+1),j}(t)\big].$$

Cette mise à jour localisée assure que la **structure** déjà présente ne subisse pas de re-calcul exhaustif.

Avantages

Le **coût** de calcul est réduit, car seul le **voisinage** immédiat d'une entité (quelques douzaines ou centaines d'éléments) est sondé, plutôt que l'ensemble du **SCN**. La **continuité** est assurée puisque les liens existants $\omega_{i,j}$ pour $i, j \leq n$ ne sont pas supprimés, évitant ainsi de **déstabiliser** les clusters déjà établis. L'**apprentissage** reste **continu**, le SCN s'agrandissant **progressivement** et chaque nouvelle insertion venant s'imbriquer naturellement dans la structure.

Risques

Si les données entrantes (frames, audio, texte) affluent à un rythme très élevé, le nombre d'entités $\mathcal{E}_k$ peut rapidement devenir **ingérable**. Il est alors nécessaire de recourir à des techniques de **fusion** ou de **compression**, comme le concept de **micro-cluster** présenté en **chapitre 7.6.3**, afin de condenser des nœuds anciens ou de désactiver certaines entités trop vieilles. Sans ces mécanismes, la matrice ω risque de **croître indéfiniment**, rendant le SCN **difficilement exploitable**.

D. Observations Mathématiques

La **mise à jour** incrémentale prônée par le chap. 7.6 évite un $O(n^2)$ complet. On se contente de $O(k)$ ou $O(\log n)$ comparaisons si l'on maintient une structure de voisinage (k plus proches, ou recherche de window temporelle). Sur un flux **multimodal**, c'est crucial pour la **scalabilité**.

Le **SCN** ne converge plus au sens classique (puisque de nouvelles entités arrivent sans cesse), mais tend vers un **régime** stationnaire ou quasi-stationnaire où les **clusters** se stabilisent

localement. Chaque **nouveau** nœud se voit insérer dans le cluster le plus pertinent, tandis que les clusters anciens conservent leur cohésion sauf si la nouvelle entité introduit une perturbation suffisamment grande pour déclencher un réajustement local (ch. 7.4 ou 7.5 sur l'inhibition compétitive).

8.9.2. Convergence et Stabilisation

Dans le **contexte multimodal**, une fois que le **SCN** (Synergistic Connection Network) a commencé à agréger les flux (vidéo, audio, texte) et à mettre à jour les pondérations $\omega_{i,j}$, il est nécessaire de définir **comment** et **quand** l'on considère qu'un **cluster** (réunissant plusieurs segments ou entités) se **stabilise**. Cette stabilisation reflète l'idée d'une **convergence** de la dynamique $\{\omega_{i,j}(t)\}$ vers une configuration cohérente, du point de vue **synergique**. Les flux vidéo, audio, et textuel convergent vers un "ensemble" qui se **renforce mutuellement**.

Dans la section 8.9.2, nous étudions plus précisément :

- Les **critères** indiquant qu'un segment multimodal (vidéo+audio+texte) forme un **cluster stable** (§8.9.2.1),
- Le **problème** d'alignement temporel, souvent épineux (§8.9.2.2), puisque l'information visuelle et audio n'est pas toujours parfaitement **synchrone**, et que le texte peut survenir en décalé.

8.9.2.1. Critères pour estimer qu'un segment vidéo+audio+texte forme un "cluster" stable

Dans un **SCN** (Synergistic Connection Network) appliqué à un **DSL** (Deep Synergy Learning) multimodal, plusieurs segments d'un flux vidéo (images ou frames), d'un flux audio (extraits sonores) et, le cas échéant, d'un flux textuel (sous-titres, annotations) peuvent s'assembler pour constituer un **cluster** commun. Le problème se pose alors de déterminer **quand** on considère que ce cluster est "abouti" ou "stabilisé", au sens où les pondérations ω reliant les entités (vidéo, audio, texte) atteignent un état relativement constant. Cette section (8.9.2.1) examine divers **critères** mathématiques et heuristiques pour estimer qu'un ensemble $\mathcal{C}$ de segments multimodaux forme un **cluster** stable.

A. Vue mathématique : convergence des pondérations ω

Chaque **entité** $\mathcal{E}_i$ du SCN représente un **fragment** issu d'un flux ou d'une modalité. Il peut s'agir d'un segment vidéo $\mathcal{V}_a$, d'un segment audio $\mathcal{A}_b$ ou d'un sous-titre $\mathcal{T}_c$. La **pondération** $\omega_{i,j}$ connecte l'entité $\mathcal{E}_i$ à l'entité $\mathcal{E}_j$. L'**auto-organisation** (chap. 2.2.2) met à jour ces pondérations ω en fonction de la **synergie** $S(i,j)$. Un **cluster** $\mathcal{C} \subseteq \{\mathcal{E}_1, \dots\}$ est un sous-ensemble d'entités (p. ex. $\{\mathcal{V}_a, \mathcal{A}_b, \mathcal{T}_c\}$) que le SCN finit par **regrouper** (fortes pondérations) et **isoler** du reste.

On considère qu'un cluster $\mathcal{C}$ se **stabilise** lorsqu'au fil du temps t, les variations $|\omega_{i,j}(t+1) - \omega_{i,j}(t)|$ deviennent négligeables pour toutes les paires $(i,j) \in \mathcal{C}$. Une écriture formelle consiste à exiger :

$$|\omega_{i,j}(t+1) - \omega_{i,j}(t)| \;\leq\; \varepsilon \quad \text{pour } (i,j) \in \mathcal{C} \times \mathcal{C},$$

avec $\varepsilon > 0$ un petit seuil. Cet **arrêt** de l'évolution signale que la dynamique du DSL (règle $\omega_{i,j} \leftarrow \omega_{i,j} + \eta\big[S(i,j) - \tau\,\omega_{i,j}\big]$) ne modifie plus sensiblement liaisons internes au cluster $\mathcal{C}$.

B. Critères de cohésion interne

Au-delà de la **convergence** en dérivée (c.-à-d. les pondérations ne bougent plus), il est possible d'imposer un certain **niveau** de cohérence. Par exemple, on définit :

$$\Omega(\mathcal{C}) = \sum_{(i,j)\in\mathcal{C}\times\mathcal{C}} \omega_{i,j}.$$

Si $\Omega(\mathcal{C})$ dépasse un **seuil** Θ, on considère que le sous-ensemble $\mathcal{C}$ présente une synergie suffisante pour être qualifié de "cluster" stable (la somme des pondérations internes étant conséquente). Dans les cas multimodaux, $\mathcal{C}$ comprend des entités de plusieurs flux (vidéo, audio, texte) dont les pondérations croisées ω sont élevées.

On peut également contrôler la **distribution** des liaisons internes au cluster. Une variance ou un étalement trop grand (certains $\omega_{i,j}$ élevés, d'autres très faibles) pourrait indiquer un cluster encore "en train" de se préciser. Si au contraire la répartition des $\omega_{i,j}$ internes est plutôt **homogène** (toutes fortes ou moyennes), on y voit un indice de **stabilité** et d'unité du groupe.

C. Dynamique multimodale

Lorsqu'un **segment** vidéo $\mathcal{V}_a$ et un segment audio $\mathcal{A}_b$ correspondent au même intervalle temporel (p. ex. la même personne qui parle), la **pondération** $\omega_{\mathcal{V}_a,\mathcal{A}_b}$ tend à augmenter (chap. 8.8.2). Si, en plus, un **texte** $\mathcal{T}_c$ (sous-titre) se révèle lié (le locuteur prononce ce sous-titre au moment t), la **pondération** $\omega_{\mathcal{V}_a,\mathcal{T}_c}$ et $\omega_{\mathcal{A}_b,\mathcal{T}_c}$ augmentent également.

On aboutit à un triplet $(\mathcal{V}_a, \mathcal{A}_b, \mathcal{T}_c)$ formant un **cluster** potentiel. La stabilisation se détecte quand :

- Les pondérations ω entre ces trois entités évoluent peu d'une itération à l'autre,
- Le **score** de cohésion interne $\Omega(\{\mathcal{V}_a, \mathcal{A}_b, \mathcal{T}_c\})$ est supérieur à un certain Θ.

Sur un flux plus long (par ex. conférence), on peut suivre l'évolution des sous-groupes. Tant que "intervenant 1" parle et que la vidéo montre la même personne, le **cluster** $\{\text{video}, \text{audio}, \text{texte}\}$ maintient des liaisons élevées. Dès qu'un changement se produit (nouvelle personne, nouvelle voix, nouveau sous-titre), la **stabilité** de l'ancien cluster chute, et un nouveau cluster commence à se former.

D. Algorithmes de suivi pour déclarer un cluster stable

Une approche pratique (chap. 7.3.1) consiste à calculer :

$$\Delta_{\text{mean}}(\omega, \mathcal{C}, t) = \frac{1}{|\mathcal{C}|^2} \sum_{(i,j)\in\mathcal{C}\times\mathcal{C}} |\,\omega_{i,j}(t+1) - \omega_{i,j}(t)|.$$

Si Δ_{mean} tombe en dessous d'un seuil δ et s'y maintient (disons, sur plusieurs itérations consécutives), on estime que le sous-groupe $\mathcal{C}$ est **convergent**. Numériquement, cela donne un **critère** d'arrêt local.

Une autre idée est de regarder la densité ou la moyenne des **connexions** internes versus les connexions externes (type "modularité"). On peut ainsi vérifier si :

- La somme interne $\Omega_{\text{in}} = \sum_{(i,j)\in\mathcal{C}} \omega_{i,j}$ est grande,
- La somme externe $\Omega_{\text{out}} = \sum_{i\in\mathcal{C},\, j\notin\mathcal{C}} \omega_{i,j}$ est petite.

Un ratio $\mathrm{Q} = \Omega_{\text{in}}/(\Omega_{\text{in}} + \Omega_{\text{out}})$ élevé (> 0.8, par ex.) peut signifier que $\mathcal{C}$ est un cluster nettement séparé.

8.9.2.2. Problème d'alignement temporel (vision, audio ne sont pas toujours synchrones)

Un **SCN** (Synergistic Connection Network) conçu pour un scénario **multimodal** (chap. 8.9) doit généralement gérer le **décalage** ou la **désynchronisation** entre différents **flux** tels que la **vision** (frames vidéo) et l'**audio** (segments sonores). Les données de ces canaux proviennent parfois de dispositifs acquis à des **fréquences** différentes ou subissent des **délais** (latences) inégaux, de sorte que l'image vue à un instant t ne coïncide pas nécessairement avec l'extrait sonore enregistré à cet instant. Cette section (8.9.2.2) développe les enjeux mathématiques et les approches adoptées dans le **DSL** pour gérer l'**alignement temporel** entre vision et audio.

A. Sources d'Asynchronie

La **désynchronisation** entre les flux **vision–audio** peut se manifester pour plusieurs raisons.

Les **différences de fréquences** d'acquisition constituent un premier facteur. Une caméra filmant à 30 images par seconde ne coïncide pas nécessairement avec un signal audio échantillonné à 48 kHz ou découpé en blocs de 20 ms, ce qui peut générer un **décalage** dans la correspondance temporelle entre les deux flux.

Les **délais de capture ou de transmission** introduisent un autre type de désalignement. Il arrive que le son soit enregistré avec un *offset* par rapport à la vidéo, ce qui entraîne un déphasage global entre ce que l'on voit et ce que l'on entend.

Enfin, des **événements asynchrones** peuvent survenir, même lorsque les flux sont **globalement alignés**. Un objet apparaissant soudainement dans une scène visuelle peut ne pas correspondre exactement à l'échantillonnage audio le plus proche, générant ainsi une **distorsion temporelle** entre l'image et le son.

D'un point de vue **mathématique**, si l'on désigne par t l'index vidéo (frame F_t) et par t' l'index audio (bloc audio $A_{t'}$), la condition de synchronie $t \approx t'$ n'est pas forcément exacte. On peut trouver un offset ℓ tel que $t' \approx t + \ell$. Une dérive dans le temps complique encore la chose car ℓ peut varier.

B. Comment modéliser la Synergie temporelle dans le DSL

Dans le **DSL**, la **synergie** $S\left(\mathcal{E}_{\text{vid},t}, \mathcal{E}_{\text{aud},t'}\right)$ ne doit pas être calculée pour tous les couples (t, t'). On introduit souvent une **fenêtre** Δ afin de considérer qu'un segment vidéo à l'instant t et un segment audio à l'instant t' peuvent être corrélés au même événement lorsque $|t - t'| \leq \Delta$. Dans ce cas, $S(\text{vid}, \text{aud})$ est calculée via une métrique (co-occurrence, corrélation, etc.). À l'inverse, si $|t - t'| > \Delta$, on pose $S(\text{vid}, \text{aud}) \approx 0$. Cette **restriction** temporelle évite l'"association" d'un frame vidéo de milieu de séquence avec un segment audio survenant bien plus tard.

Le **décalage** ℓ entre l'audio et la vidéo peut prendre plusieurs formes.

Un **décalage global et constant** se produit lorsque la caméra et le micro introduisent un **offset fixe** ℓ^*. Dans ce cas, une correction simple consiste à imposer que la synergie entre les flux vidéo et audio soit nulle lorsque $|(t + \ell^*) - t'| > \Delta$. Cela garantit que seules les correspondances temporelles respectant une tolérance Δ sont prises en compte.

Un **décalage local et variable** est plus complexe, car il évolue au fil du temps. Un algorithme d'**alignement** tel que la **cross-corrélation** ou le **Dynamic Time Warping** peut estimer, pour chaque portion temporelle, une valeur optimale de $\ell(t)$. Dans ce cas, la fonction de synergie $S(\text{vid}, \text{aud})$ doit intégrer cette dynamique temporelle en tenant compte de $\ell(t)$. Cela complexifie la définition de la synergie, mais permet d'obtenir un **alignement plus précis** lorsque la désynchronisation varie progressivement au cours du temps.

Le **SCN** s'appuie sur la mise à jour $\omega_{(t),(t')} \leftarrow \omega_{(t),(t')} + \eta \left[S(t, t') - \tau\, \omega_{(t),(t')} \right]$. Pour que ω s'élève, il faut un **score** de synergie non nul, ce qui requiert une **fenêtre** temporelle ou un offset local approprié. Dans le cas contraire, la liaison $\omega_{(t),(t')}$ reste proche de zéro, signifiant que le SCN ne perçoit aucune co-occurrence (vision–audio) entre frame t et bloc t'.

C. Dilemme mathématique : Correction globale vs. correction locale

Si l'on a un offset ℓ^* unique (ex. +100 ms sur l'audio), on peut rectifier tous les index audio en posant $\widetilde{t'} = t' + \ell^*$. Cette manipulation "aligne" l'audio sur la vidéo de manière fixe. Le SCN s'en retrouve simplifié car $S(\text{vid}, t, \text{aud}, t')$ se calcule pour $|t - (\widetilde{t'})| \leq \Delta$. Cependant, toute dérive progressive ou changeante n'est pas prise en compte.

Lorsque la désynchronisation n'est pas constante dans le temps, on recourt à des méthodes plus complexes (cross-corrélation glissante, DTW). On obtient alors une fonction $\ell(t)$ où l'audio présente un décalage local de $\ell(t)$. Le SCN doit alors faire $S(\text{vid}, t, \text{aud}, t')$ non nul si $|t' - (t + \ell(t))| \leq \Delta$. D'un point de vue **computational**, cela augmente la difficulté, mais garantit un **alignement** plus fin, crucial pour des scènes variables.

D. Cas de la synergie multimodale à sampling irrégulier

La vidéo peut être échantillonnée en 30 fps, l'audio peut l'être en 16 kHz (ou plus), et des sous-titres arriver par segments d'une seconde. On ne dispose pas d'un index temporel "universel". On manipule donc plusieurs **listes** d'horodatage. On cherche un critère permettant de déterminer si le segment vidéo $\mathcal{E}_{(\text{vid},t)}$ correspond ou non à la portion audio $\mathcal{E}_{(\text{aud},t')}$.

Deux façons de faire :

- **Re-sampling** : on projette tout dans une timeline commune $\{t_k\}$. On assigne à un bloc audio $\mathcal{A}_m$ la référence temporelle $\tilde{t}_m \approx$ moyenne de son intervalle. Idem pour le frame vidéo $\mathcal{V}_n$. On compare ensuite $|\tilde{t}_m - \tilde{t}_n|$.

- **Kernel** : on déclare $S(\text{vid}, t, \text{aud}, t') = \kappa(|t - t'|, \delta)$ (ex. kernel Gaussien, ou nul si $|t - t'| > \delta$). Cela revient à imposer une pondération sur la différence temporelle, la synergie chutant rapidement au-delà d'une fenêtre δ.

E. Impact sur la Qualité d'Apprentissage

Un **mauvais** alignement temporel pénalise fortement la **fusion** multimodale. le SCN associe des frames vidéo à des segments audio inappropriés, ce qui dilue la **synergie** globale et dégrade la reconnaissance d'événements (chap. 8.8.2). À l'inverse, un alignement maîtrisé rend $S(\text{vid}, t, \text{aud}, t')$ élevé pour les paires effectivement synchrones (ex. la parole d'un locuteur et

sa bouche en mouvement). Le **DSL** renforce alors les pondérations adéquates, favorisant un **cluster** stable autour de l'événement commun.

8.9.3. Suivi et Visualisation

Dans tout **système multimodal** (chap. 8), la gestion des liaisons $\omega_{i,j}$ entre entités (images, segments audio, tokens textuels, etc.) peut rapidement devenir **complexe** si l'on ne dispose pas d'outils adéquats pour **suivre** l'évolution du réseau et **visualiser** la structure émergente. Le **DSL** (Deep Synergy Learning), en permettant des mises à jour $\omega_{i,j}(t+1)$ fondées sur la synergie $S(i,j)$, suscite la formation de **clusters** multimodaux, la fusion ou la scission de groupes, et des réajustements constants des pondérations.

Cette section (8.9.3) met l'accent sur les méthodes et **outils** permettant de **montrer** ou d'**analyser** :

- Les **changements** dans la matrice ω (ou la structure du SCN) au fil des itérations,
- L'**interaction** entre différentes modalités (images–sons, images–textes, etc.),
- L'**intégration** de ces outils dans un pipeline de classification ou de prise de décision.

8.9.3.1. Outils pour montrer l'évolution de $\omega_{i,j}$ entre images, sons, mots-clés

Dans le **DSL** (Deep Synergy Learning), un **SCN** (Synergistic Connection Network) relie des **entités** issues de divers flux multimodaux (images, segments audio, mots-clés, etc.). Au fil des itérations, les **pondérations** $\omega_{i,j}$ reliant ces entités se modifient suivant la règle de mise à jour, reflétant la **synergie** $S(i,j)$. Il est souvent **crucial** de **montrer** ou de **suivre** l'évolution de ces pondérations $\omega_{i,j}$, afin de comprendre comment le réseau se structure, comment émergent les **clusters** multimodaux, et si l'**auto-organisation** se déroule correctement. La présente section (8.9.3.1) aborde plusieurs **outils** et méthodes de visualisation utiles à cet effet.

A. Matrice Dynamique : Heatmaps et Graphiques Temporels

Une stratégie simple et informative consiste à représenter la **matrice** $\mathbf{\Omega}(t) = [\omega_{i,j}(t)]$ sous forme de **heatmap**. Chaque cellule (i,j) est colorée selon la valeur de $\omega_{i,j}(t)$. Pour un **réseau** de taille n, on obtient ainsi une **matrice** $n \times n$ visualisée. En faisant varier le temps t (ou l'itération) comme un **slider**, on peut "**dérouler**" (playback) la séquence de heatmaps, observant :

- Les liaisons qui **montent** (ex. en passant d'une valeur $\omega_{i,j} = 0.05$ à 0.7)
- Les liaisons qui s'**atténuent**
- Les **clusters** (groupes de nœuds internes) qui se dessinent via un bloc de pondérations fortes dans la heatmap.

Une **analyse** plus quantitative est possible en calculant la norme (ex. ℓ_1, ℓ_2), la **somme** $\sum_{i,j} \omega_{i,j}(t)$ ou la **distribution** $\{\omega_{i,j}(t)\}$ itération par itération. Ces statistiques renseignent sur le taux global de renforcement vs. effacement.

Une variante plus focalisée consiste à tracer, pour **quelques** paires (i, j) d'intérêt (par ex. un segment audio et une image soupçonnée de correspondre), la **courbe** $\omega_{i,j}(t)$ au cours des itérations. Cela met en lumière le **rythme** d'augmentation ou de décroissance ainsi que l'**instant** où le couple (i, j) atteint une valeur élevée et se stabilise.

De manière plus globale, on peut considérer l'évolution de la **cohésion** $\Omega(\mathcal{C}, t)$ (la somme ou la moyenne des $\omega_{i,j}$ internes à un cluster $\mathcal{C}$), révélant la formation d'un **groupe** stable.

B. Visualisation Graphe 2D/3D

Au lieu d'afficher une **matrice**, on peut construire un **graphe** où chaque **noeud** représente une entité (image, mot-clé, segment audio, etc.), et chaque **arête** possède un **poids** $\omega_{i,j}$. Un algorithme de layout (type ForceAtlas, Force-Directed) place les nœuds en 2D ou 3D, éloignant ceux dont $\omega_{i,j} \approx 0$ et rapprochant ceux reliés par un $\omega_{i,j}$ plus fort.

En le faisant évoluer au fil des itérations (chaque itération du DSL correspond à une "image" de ce graphe), on **voit** la structure se **réorganiser** graduellement. Des nœuds "image" et des nœuds "audio" fortement liés se rapprochent, formant des **clusters** multimodaux. Les arêtes trop faibles disparaissent ou restent filiformes (fines, peu visibles), tandis que les arêtes fortes (pondérations élevées) s'épaississent.

Dans un environnement multimodal complet (images, sons, mots, règles, etc.), la **visualisation** globale peut devenir saturée. On peut se **focaliser** sur deux modalités (par ex. *images* vs. *mots*) en ne montrant que les liaisons $\omega_{i,j}$ pour $i \in \text{Images}, j \in \text{Mots}$. Cela revient, en pratique, à extraire un **sous-bloc** de la matrice $\boldsymbol{\Omega}(t)$ ou un **sous-graphe** du SCN, limité aux entités image–texte. On voit alors plus facilement quelles **images** se connectent à quels **mots**.

C. Outils de Focalisation sur Groupes Particuliers

Outre la **sélection** d'une modalité (image, audio, texte), on peut filtrer les $\omega_{i,j}$ trop faibles (en dessous d'un seuil θ). Cela permet de clarifier la **vision** du graphe. Seules les **connexions** qui ont atteint un certain degré de solidité (par ex. $\omega_{i,j} > 0.3$) sont affichées, mettant en évidence un **clustering** mieux délimité.

Si l'on sait qu'un **cluster** $\mathcal{C}$ (ex. un sous-ensemble de frames et de segments audio) s'est formé, on peut **surveiller** la somme interne $\sum_{(i,j)\in\mathcal{C}\times\mathcal{C}} \omega_{i,j}\,(t)$ et la somme externe $\sum_{i\in\mathcal{C}, j\notin\mathcal{C}} \omega_{i,j}\,(t)$. En temps réel, on observe si la cohésion interne du cluster $\mathcal{C}$ continue de croître ou stagne, et si les liaisons vers l'extérieur décroissent ou non, reflétant une **stabilisation** (chap. 8.9.2 sur la stabilité).

D. Exemples concrets

On peut disposer d'un ensemble de **mots-clés** (tokens ou embeddings textuels) et d'un ensemble de **images** (embeddings visuels). La **matrice** $\omega(t)$ reflète la force de correspondance image–mot. En affichant une heatmap, on verra, itération après itération, **qui** se lie à **qui**. Un groupe d'images "chats" se connectera aux mots "cat, kitten, feline", tandis qu'un autre groupe "voiture" pointera vers "car, vehicle, road". On pourra surveiller la formation ou la désagrégation de blocs dans la heatmap.

Dans un **graphe** 2D/3D, chaque nœud audio (ex. segment musical) se retrouvera à proximité des nœuds vidéo où la scène "correspond" (même intervalle de temps, cohérence de contenu). Au fur et à mesure, on voit se constituer des **clusters**. Par exemple, un "cluster urbain" (vidéo

de rues + bruits de moteur + sons de klaxon) et un “cluster nature” (vidéo de forêt + chant d’oiseaux). Les arêtes $\omega_{i,j}$ s’épaississent pour ces entités de forte synergie.

E. Suivi de l’Évolution Temporelle

Pour un **monitoring** en ligne, on **enregistre** $\mathbf{\Omega}(t)$ ou des mesures résumées à intervalle régulier. On **visualise** ensuite l’évolution à l’aide d’un “**player**” temporel de type timeline ou sous forme de courbes représentant $\omega_{i,j}$ en fonction du temps t. On peut croiser ces observations avec les **critères** de stabilisation afin d’identifier le moment où un **cluster** se consolide.

8.9.3.2. Tableaux de bord (dashboard) multimodal, exploitation possible dans un pipeline de classification

Dans un **DSL** (Deep Synergy Learning) appliqué à un **SCN** (Synergistic Connection Network) multimodal, il est essentiel de *monitorer* l’évolution et la structure de la fusion (ch. 8.8 et 8.9) de manière **interne** et en temps réel. Les **tableaux de bord** (ou *dashboards*) répondent à ce besoin. Ils fournissent une **vue** synthétique de l’état du réseau, des **pondérations** ω, des **clusters** formés, et des **tendances** de l’auto-organisation. Cette section (8.9.3.2) décrit comment de tels tableaux de bord peuvent être intégrés dans un **pipeline** plus large, par exemple un **workflow** de **classification** multimodale.

A. Notion de Tableaux de Bord (Dashboards) Multimodaux

Les **tableaux de bord** permettent de **visualiser** et **superviser** l’évolution du SCN. Dans un **contexte** multimodal, ils offrent une vue sur la **structure** des clusters et macro-nœuds reliant des entités d’origines différentes, telles que les images, l’audio et le texte. Ils affichent également les **pondérations** $\omega_{i,j}(t)$ les plus élevées ainsi que leur distribution et permettent de suivre les **dynamiques** de la synergie S ou de la pondération ω au fil du temps.

Dans la pratique, ces dashboards comprennent plusieurs **modules** ou **vues**. Une **vue graphe** représente en 2D ou en 3D les nœuds et leurs arêtes pondérées ω, souvent avec un layout de type “force-directed”. Une **heatmap** ou **matrice** $\mathbf{\Omega}(t)$ permet d’analyser un sous-bloc spécifique, comme les relations image–texte ou image–audio, afin d’observer l’évolution de certaines liaisons. Des **courbes** temporelles illustrent la progression de l’**auto-organisation**, en affichant par exemple la somme de ω ou la fonction d’énergie $\mathcal{J}(\Omega)$ au cours des itérations (chap. 2.2.2 ou 7.2). Enfin, des **indicateurs** tels que la taille des clusters, la densité des liens ou encore l’entropie permettent de suivre la **cohésion** des sous-groupes multimodaux.

B. Exploitation dans un Pipeline de Classification

Dans un **pipeline** de classification (en vision, NLP, etc.), on opère souvent une **fusion** à un niveau “feature” (on concatène des vecteurs) ou à un niveau “decision” (on combine des scores). Avec un **SCN** (voir chap. 8.8), la fusion se produit par **auto-organisation** des entités multimodales, générant **clusters** ou macro-nœuds. On peut alors **extraire** ces **macro-nœuds** comme des **features** pour un classifieur en aval. La structure permet de **déterminer** de manière auto-organisée **quel** segment audio s’attache à **quelle** scène vidéo et d’**exploiter** la matrice ω pour prendre des décisions hiérarchiques.

Le **dashboard** multimodal joue un **double** rôle. Il permet de **monitorer** en temps réel pour vérifier si le SCN regroupe correctement les flux, par exemple en associant les segments vidéo “chien” aux segments audio “aboiement”. Il sert également à **orienter** ou **paramétrer** le pipeline, permettant à un opérateur ou à un module automatique de modifier η, τ ou des

coefficients d'inhibition selon l'état de la structure observée. De plus, on peut réinjecter un "recuit" (voir 7.3.2) en cas de blocage détecté.

Lorsque le **SCN** stabilise un **cluster** $\mathcal{C}$ (par ex. ensemble d'images, audio, mot-clé sémantique), un **classifieur** (apprentissage supervisé, ou un module symbolique) peut en déduire une étiquette (ex. "chat" ou "voiture"). Le tableau de bord montre aussi la **progression** de cette classification, indiquant si le cluster $\mathcal{C}$ correspond à "chat" (forte présence de miaulements et de la forme féline détectée).

C. Intégration de Paramètres et Feedback

Un **tableau de bord** peut comprendre des **curseurs** permettant d'ajuster les paramètres η, τ et le "taux" d'inhibition. Il peut également inclure un **bouton** de recuit "light" ou "heavy" pour intervenir en cas de **blocage** (minima local) dans la formation des clusters.

Le pipeline de classification en aval reçoit la **nouvelle** structure ω mise à jour. Cette interaction signifie qu'on injecte un **contrôle** humain (ou algorithmique) dans les hyperparamètres du DSL, se basant sur la **visualisation** pour décider quand intervenir.

Dans certains cas, on peut disposer d'un **feedback** partiel (par ex. un label correct pour un segment vidéo–audio), permettant de **valider** ou **invalider** un cluster formé par le SCN. On peut afficher dans le dashboard la **précision** ou la **réussite** de la classification multimodale associée, renforçant la **compréhension** du niveau de confiance.

D. Cas Pratique

Supposons un pipeline où le **SCN** reçoit des frames vidéo pour reconnaître la personne qui parle, des segments audios pour identifier la voix et du texte pour analyser les sous-titres. Un **dashboard** permet alors de visualiser plusieurs éléments clés :

- Un **graphe 2D** représentant les **connexions** et la pondération ω entre segments vidéo affichant "personne A", segments audio correspondant à "voix A" et sous-titres mentionnant "M. A".
- Des **courbes** illustrant la somme des liens $\sum_{(\text{vid,aud})} \omega_{i,j}$, mettant en évidence la force des correspondances audio–visuelles.
- Une **liste des clusters actuels**, par exemple, "Intervenant 1 (vidéo #2, audio #5, texte #1)".

L'utilisateur ou un module supervisé peut alors **intervenir** si un cluster "Intervenant 1" est incorrect, en ajustant un paramètre ou en injectant une correction.

Le pipeline final (classification "Qui parle en ce moment ?") s'appuie sur ce cluster auto-organisé pour étiqueter "c'est la voix de M. A, vue à l'écran".

Sans tableau de bord, on a un SCN qui s'auto-organise en **boîte noire**. On ignore si certains liens ω se sont **excessivement** solidifiés à cause d'un bruit ou d'une confusion (ex. un segment audio d'oiseau relié à un plan vidéo de chat). Le dashboard **révèle** visuellement ces erreurs potentielles. On peut alors corriger ou ajuster le pipeline (ch. 7.4 sur l'inhibition ciblée).

8.10. Limites, Défis et Pistes de Recherche

Même si le **DSL multimodal** (voir chapitres précédents) offre de nouvelles capacités pour intégrer, associer et organiser des flux variés (image, audio, texte, etc.), il se heurte inévitablement à des **contraintes** liées à la **complexité** des données réelles, à la **qualité** parfois imparfaite des embeddings, et à des enjeux de **sécurité** et de **biais**. La présente section (8.10) met en relief ces **limites** et **défis**, tout en esquissant des **pistes de recherche** pour aller plus loin.

8.10.1. Complexité

L'un des principaux obstacles auxquels fait face un **SCN** (Synergistic Connection Network) en mode multimodal repose sur le **nombre** d'entités à traiter. Segmenter un flux vidéo en frames, un flux audio en unités ou fonctions, un texte en phrases ou en tokens peut entraîner une explosion de la quantité totale d'éléments $\{\mathcal{E}_i\}$. Cela se traduit par une **dimension** potentiellement gigantesque pour la matrice $\{\omega_{i,j}\}$, d'où un **coût** de mise à jour et de recherche en $O(n^2)$.

8.10.1.1. Le nombre d'entités explose si on segmente un flux vidéo (frames), audio (fonctions) et texte (phrases)

Dans un **SCN** (Synergistic Connection Network) destiné à la **fusion multimodale** (vision, audio, texte, etc.), la quantité d'**entités** à traiter peut croître de façon considérable dès lors qu'on segmente finement chaque flux. Les sections précédentes (chap. 8.8, 8.9) mettent en évidence l'intérêt de la segmentation pour associer segments vidéo à segments audio. La présente section (8.10.1.1) détaille la **problématique** qui en découle, à savoir l'**explosion** du nombre de nœuds à gérer dans le **DSL**, et donc d'un potentiel $O(n^2)$ liaisons $\{\omega_{i,j}\}$.

A. Segmentation Vidéo

Un **flux vidéo** de durée D (quelques minutes, voire plus) peut être découpé en **centaines** ou **milliers** de frames si l'on conserve la plupart des images (ex. 25 ou 30 fps). Chaque **image** devient alors une **entité** $\mathcal{E}_i^{(\text{vid})}$. De surcroît, si l'on opte pour une segmentation plus granulaire (patchs par frame, détection d'objets locaux), la multiplication des entités s'accroît davantage.

Avec F frames et P patchs par frame, on obtient $n = F \times P$ entités vidéo. Si F s'élève à plusieurs milliers et P n'est pas négligeable, la taille n devient rapidement **massive**. Dans un **SCN**, le nombre de liaisons ω peut atteindre $O(n^2)$. À mesure qu'on traite des vidéos plus longues ou plus riches en segments, ce volume de nœuds et de liens menace de **saturer** la mémoire et le temps de calcul.

B. Segmentation Audio

Le **flux audio** s'analyse généralement via des **blocs** d'une durée de quelques millisecondes (10, 20, 40 ms) ou via des unités (segments phonétiques, etc.). Chaque bloc se convertit en un **vecteur** de features (MFCC, spectrogramme) puis en une entité $\mathcal{E}_j^{(\text{aud})}$. Comme la fréquence d'échantillonnage peut être élevée (ex. 16 kHz, 48 kHz), le **nombre** de segments audio grimpe vite à des **milliers** ou plus, tout au long d'un enregistrement de plusieurs minutes.

Un exemple concret illustre cette problématique. Dix minutes d'audio, segmenté en **fenêtres** de 5 ms (non superposées), dégage 120 000 blocs. Cela engendre autant d'entités dans le **SCN** si aucun pré-regroupement n'est opéré. La mise à jour en $O(n^2)$ devient alors problématique.

C. Segmentation Textuelle

Les **flux** de texte (ex. sous-titres, transcription, documents associés) peuvent aussi générer de très **nombreuses** entités. On segmente souvent le texte en **phrases** ou en **tokens** (mots, sous-mots). Une conférence, un livre ou un corpus volumineux conduit potentiellement à des **dizaines** voire **centaines de milliers** de tokens.

Au sein du **SCN**, chaque **token** ou **phrase** serait un noeud $\mathcal{E}_k^{(\mathrm{txt})}$. Or, ajouter une telle masse de nœuds alourdit énormément la structure, gonflant la matrice $\{\omega_{i,j}\}$. Lorsque le **DSL** traite la synergie **image–texte** (chap. 8.8.1) ou **audio–texte** (subtitling), l'explosion de n'entités nuit à la faisabilité d'une mise à jour naïve en $O(n^2)$.

D. Fusion Multimodale : le cumul des entités

Lorsque le **SCN** veut simultanément gérer :

- Les entités **vidéo** (n_{vid}) extraites des frames ou patchs,
- Les entités **audio** (n_{aud}) issues de la segmentation sonore,
- Les entités **textuelles** (n_{txt}) (phrases, tokens, etc.),
- Éventuellement d'autres sources (capteurs, règles symboliques),

on obtient un volume total $n_{\mathrm{vid}} + n_{\mathrm{aud}} + n_{\mathrm{txt}} + \cdots$. Ce total n peut atteindre rapidement le **million**, rendant impossible toute gestion brute de $n(n-1)/2$ pondérations ω. Les algorithmes DSL, s'ils ne prévoient pas de **stratégies** de limitation, risquent d'**exploser** en mémoire et en temps de calcul.

8.10.1.2. Nécessité d'heuristiques (k plus proches voisins, cluster gating, etc.)

Lorsque la **taille** du **SCN** (Synergistic Connection Network) croit significativement, la charge de calcul en $O(n^2)$ devient inacceptable. Il faut envisager des **stratégies** d'allégement pour gérer les **pondérations** ω entre n entités sans manipuler toutes les paires (i,j). Les sections précédentes (8.10.1.1) ont montré comment le **nombre** d'entités peut exploser en multimodal (images + audio + texte), ce qui **impose** l'emploi d'**heuristiques**. L'idée principale est de **restreindre** la mise à jour $\omega_{i,j}$ à un **sous-ensemble** limité de paires. Cette section (8.10.1.2) présente deux familles d'approches, le "k plus proches voisins" (k-NN) et le "cluster gating".

A. Heuristique du "k plus proches voisins" (k-NN)

L'approche k-NN **restreint** la recherche de **synergie** $S(i,j)$ à un **voisinage** $\mathrm{N}_k(i)$ autour de l'entité $\mathcal{E}_i$.

On **définit** un critère (souvent basé sur des embeddings, distances ou similarités initiales) pour **classer** les autres entités $\{\mathcal{E}_j\}_{j\neq i}$ par ordre de proximité vis-à-vis de $\mathcal{E}_i$.

On **choisit** les k plus proches d'entre elles (ou un nombre restreint équivalent).

La **mise à jour** $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \cdots$ ne s'applique **qu'**à ces k voisins (et possiblement, à quelques nouveaux candidats si des événements indiquent que la distance a évolué). Sinon, $\omega_{i,j}$ reste figé à 0 ou une valeur ancienne.

Sur le plan **mathématique**, l'algorithme DSL se limite ainsi à $O(n\,k)$ ou $O(n\log n)$ si l'on actualise régulièrement les k plus proches voisins via une structure comme un **kd-tree** ou **ball-tree**. Cela ramène la complexité de $O(n^2)$ à $O(n\,k)$, ce qui devient gérable si $k \ll n$.

Les **avantages** de cette approche résident principalement dans l'**économie** en temps de calcul, puisque seules les k entités les plus pertinentes sont explorées pour chaque nœud du réseau. Cela permet d'éviter l'explosion combinatoire du nombre de paires à évaluer tout en préservant les liens significatifs. De plus, un **filtrage** automatique s'opère sur les connexions les plus faibles ou improbables, ce qui limite la saturation du réseau et garantit une organisation plus claire et plus efficace du **SCN**.

Toutefois, cette approche présente certaines **limites**. Un phénomène de **cloisonnement** peut apparaître lorsque, dans un premier temps, une entité $\mathcal{E}_i$ est classée parmi les k plus proches voisins d'une autre entité $\mathcal{E}_j$, mais que l'inverse ne se produit pas. Cela introduit un **biais** structurel, où certaines relations potentielles ne peuvent pas émerger naturellement à cause d'un voisinage restreint. De plus, il existe un risque d'**erreurs** de partition lorsque des liaisons pertinentes, bien que caractérisées par une distance initialement grande, sont écartées trop tôt. Une solution consiste à prévoir un **rafraîchissement** périodique du voisinage, permettant ainsi à des liens **long-courriers** mais sémantiquement valides d'apparaître progressivement et d'être intégrés au sein du **SCN**.

B. Cluster Gating : Filtrage sur la Base de Clusters Émergents

Une seconde approche consiste à **grouper** (même de façon approximative) les entités en **clusters** (ou super-nœuds), puis à n'autoriser la mise à jour $\omega_{i,j}$ qu'entre entités dans un même cluster ou entre clusters voisins. On parle de "**cluster gating**". Cela peut être vu comme un **gating** binaire $G(i,j) \in \{0,1\}$:

$$\omega_{i,j}(t+1) = G(i,j)\left[\omega_{i,j}(t) + \eta\left(S(i,j) - \tau\,\omega_{i,j}(t)\right)\right],$$

de sorte que si $G(i,j) = 0$, la liaison $\omega_{i,j}$ n'est pas mise à jour (reste 0 ou inchangée). Ce "gating" s'active $(= 1)$ pour les paires (i,j) jugées **compatibles**. i et j appartiennent au même cluster ou à des clusters proches dans l'arbre hiérarchique (chap. 6).

Le **bénéfice** du **cluster gating** est de réduire considérablement le **nombre** de liaisons évaluées. Une fois qu'un **gros** cluster $\mathcal{C}_\alpha$ s'est stabilisé, on peut limiter l'évolution $\omega_{i,j}$ aux entités à l'intérieur de $\mathcal{C}_\alpha$ ou aux entités proches, sans s'occuper de paires très distantes.

Le **risque** est qu'un cluster ou super-nœud **verrouille** excessivement la structure, rendant difficiles les re-liaisons inter-clusters. On peut envisager un **rafraîchissement** occasionnel, réexaminant certaines paires inter-clusters pour autoriser des réassignations.

C. Pourquoi de telles Heuristiques ?

La **motivation** principale est la **scalabilité**. En se passant d'un traitement exhaustif $O(n^2)$, on ramène le calcul à $O(n\,k)$ ou $O(n\,c)$ selon la méthode. Les heuristiques veillent à ne pas brider trop la découverte de **nouveaux** liens potentiellement significatifs.

8.10.2. Qualité de la Synergie

Dans un **SCN** (Synergistic Connection Network) multimodal, la pertinence de la **synergie** $S(i,j)$ dépend étroitement de la **qualité** des **représentations** (embeddings) associées à chaque modalité (image, texte, audio, etc.). Si ces embeddings sont imparfaits — bruités, mal calibrés ou incomplets —, il en résulte un calcul de synergie moins fiable, susceptible de **tromper** le réseau dans la formation de clusters ou de super-nœuds inappropriés. La présente section (8.10.2) vise à examiner comment **évaluer** ces embeddings, puis comment les **calibrer** ou les **normaliser** pour conserver un niveau de synergie cohérent à travers plusieurs canaux.

8.10.2.1. Évaluation des embeddings multimodaux : s'ils sont imparfaits, le SCN peut se tromper

Un **SCN** (Synergistic Connection Network) déployé dans un **DSL** (Deep Synergy Learning) multimodal s'appuie fortement, pour son calcul de **synergie** $S(i,j)$, sur des **embeddings** sub-symboliques (vision, audio, texte) censés traduire la similarité ou la correspondance entre entités. Lorsque ces embeddings sont **imparfaits** (qualité sous-optimale, biais, surapprentissage…), la mesure $S(\mathcal{E}_i, \mathcal{E}_j)$ en est faussée. De telles erreurs d'estimation engendrent une **auto-organisation** trompeuse. Le SCN peut créer ou renforcer des liens $\omega_{i,j}$ là où la **réalité** (sémantique) ne le justifie pas, ou, à l'inverse, négliger des connexions pertinentes. La présente section (8.10.2.1) discute comment l'**embedding** imparfait se répercute sur la fiabilité du SCN, et pourquoi il importe de **surveiller** la qualité des embeddings avant ou pendant l'exécution du DSL.

A. Impact d'un Embedding Imparfait

Dans un **SCN** multimodal, chaque **entité** $\mathcal{E}_i$ (image, texte, segment audio, etc.) se voit associé un **vecteur** $\mathbf{v}_i$. La **synergie** $S(i,j)$ est souvent calculée comme une **similarité** (cosinus, produit scalaire, distance exponentielle) ou une mesure statistique (information mutuelle) dérivée des embeddings. Si les embeddings sont **imparfaits** — par exemple, mauvaise généralisation, confusion dans les features, absence de capture des nuances sémantiques — la **valeur** $S(i,j)$ risque de ne plus refléter la véritable proximité ou compatibilité sémantique des entités $\mathcal{E}_i$ et $\mathcal{E}_j$.

Considérons :

- $\mathbf{v}_i^{(\text{image})}$: embedding tiré d'un **CNN** trop peu entraîné, confondant parfois un chien et un chat,
- $\mathbf{v}_j^{(\text{txt})}$: embedding textuel issu d'un modèle à couverture insuffisante (les mots "cat" et "car" se révèlent trop proches).

Le **SCN**, au moment de calculer $S\left(\mathcal{E}_i^{(\text{image})}, \mathcal{E}_j^{(\text{txt})}\right)$, est induit en erreur si l'**embedding** cosinus $\cos(\mathbf{v}_i, \mathbf{v}_j)$ s'avère artificiellement élevé pour la paire (chien, "car") ou trop faible pour la paire (chien, "dog"). On verra alors le **DSL** renforcer $\omega_{(chien),(car)}$ et ignorer $\omega_{(chien),(dog)}$, ce qui fausse le regroupement (cluster incorrect).

B. Conséquences Mathématiques dans le SCN

La dynamique DSL :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$$

s'opère à chaque itération, où $S(i,j) = \mathrm{sim}(\mathbf{v}_i, \mathbf{v}_j)$. Si $\mathbf{v}_i, \mathbf{v}_j$ sont mal positionnés dans l'espace d'embedding, $\mathrm{sim}(\mathbf{v}_i, \mathbf{v}_j)$ peut être **trop** grand ou **trop** petit par rapport à la réalité. Cela engendre un **renforcement** de liens $\omega_{i,j}$ qui ne devraient pas l'être et un **abandon** ou une **sous-estimation** de liaisons qui, dans un embedding correct, seraient révélées comme pertinentes.

On obtient alors un **réseau** ω dont la topologie n'est pas conforme à l'authentique structure sémantique, et qui risque de converger vers un **minimum local** "faux".

Dans un DSL, les **clusters** émergents (groupes de nœuds fortement connectés) correspondent en principe à des **concepts** ou des ensembles sémantiques partagés (p. ex. "chien + aboiement + mot dog"). Mais si l'embedding fait confondre "chien" et "chat", le SCN peut aboutir à un cluster (chien, chat, meowing) ou (chien, "cat"), un bloc **incohérent**. Les pondérations internes ω se stabilisent néanmoins parce qu'aucun autre signal n'est venu **corriger** l'erreur.

En somme, le **SCN** se fiant aux embeddings donne un "garbage in, garbage out". Si la similarité S est trompeuse, l'organisation ω l'est aussi.

C. Nécessité d'une Évaluation et d'une Calibration des Embeddings

Avant d'utiliser un embedding (image, audio, texte) dans le SCN, il est prudent de **contrôler** sa qualité. Par exemple, si on a un jeu de test (images + labels), on vérifie si la distance cosinus reflète la proximité de labels. Ou dans le texte, on regarde si les embeddings distinguent bien "cat" et "car". L'échec de ce test incite à **réentraîner** ou **raffiner** l'embedding.

Si on ne peut pas réentraîner, on peut recourir à une **calibration** statistique, par ex. un **scale** ou un **offset** tenant compte des distributions courantes, un alignement sémantique (coordonner l'échelle de l'espace image et l'espace texte). Des méthodes *fine-tuning cross-modale* (ex. style CLIP) permettent parfois de réduire le fossé entre embeddings hétérogènes.

Dans certains cas (voir chap. 8.6–8.7), on peut injecter un **feedback** top-down. Si le SCN détecte des incohérences massives dans un cluster, on soupçonne un problème d'embedding ; on envoie un signal ou on "réapprend" localement le mapping pour mieux aligner $\mathbf{v}_i, \mathbf{v}_j$. Cela suppose un DSL unifié ou couplé à l'étape d'entraînement de l'embedding.

8.10.2.2. Approches pour calibrer $S(i,j)$, ex. normalisation cross-modal

Lorsque l'on applique un **SCN** (Synergistic Connection Network) à un **DSL** (Deep Synergy Learning) dans un **contexte multimodal** (images, audio, textes, etc.), la **fonction** $S(i,j)$ quantifiant la synergie entre deux entités $\mathcal{E}_i$ et $\mathcal{E}_j$ se heurte à des **hétérogénéités** de **modalités** et d'**échelles**. Les différentes **sources** (vision, audio, texte…) ne délivrent pas des vecteurs ou mesures directement comparables. Cette section (8.10.2.2) détaille les méthodes de **calibrage** et de **normalisation** visant à **unifier** ou du moins **harmoniser** les scores de similarité/distance, afin de rendre $S(i,j)$ **cohérent** à l'échelle du SCN tout entier.

A. Problématique de la Normalisation Cross-Modal

Chaque **modalité** (image, audio, texte, etc.) produit des embeddings $\mathbf{v} \in \mathbb{R}^d$ selon des **dimensions** et des **échelles** variables. On peut par exemple avoir :

- Un embedding visuel $\mathbf{x}_i^{(\mathrm{img})} \in \mathbb{R}^{2048}$ provenant d'un CNN,

- Un embedding audio $\mathbf{x}_j^{(\text{aud})} \in \mathbb{R}^{256}$ résumant un segment sonore,
- Un embedding textuel $\mathbf{x}_k^{(\text{txt})} \in \mathbb{R}^{768}$ d'un modèle transformer.

Comparer directement $\| \mathbf{x}_i - \mathbf{x}_j \|$ ou $\cos(\mathbf{x}_i, \mathbf{x}_j)$ sans **calibrer** l'échelle (dimension, distribution, amplitude) peut poser problème. Une **modalité** pourrait avoir des valeurs de norme plus élevées par construction, ou la distribution statistique des embeddings diffère drastiquement, conduisant à des **scores** S difficilement comparables.

Même **au sein** d'une modalité unique, on peut générer plusieurs **scores**. Un pour la forme globale, un autre pour la texture, un troisième pour la scène audio globale, un quatrième pour la voix localisée, etc. Dans un environnement **multimodal**, on obtient un éventail de **fonctions** $S^{(m)}$, qu'il faut **fusionner** en un **score** global $S(i,j)$. Pour éviter qu'une **mesure** (avec amplitude plus élevée) ne domine, on doit **normaliser** ou **pondérer** chacun de ces scores.

B. Formulations Mathématiques de Normalisation

Une **méthode** répandue est de **centrer–réduire** les valeurs de similarité d'une modalité. Supposez qu'on considère la distribution $\{S^{(\text{mod})}(i,j)\}$ sur l'ensemble des paires (i,j) de cette modalité, on définit :

$$\tilde{S}^{(\text{mod})}(i,j) = \frac{S^{(\text{mod})}(i,j) - \mu_{\text{mod}}}{\sigma_{\text{mod}}}$$

pour un offset μ_{mod} (moyenne) et une échelle σ_{mod} (écart-type). On peut ensuite le **ramener** à $[0,1]$ par un min–max ou une sigmoïde. Ainsi, on obtient des **scores** comparables entre modalités. Sur le plan **mathématique**, on veille à ce que chaque flux ait une distribution de similarités homogène, évitant qu'un flux "embeddings plus longs" dépasse systématiquement les autres.

Si on a M modalités, on peut écrire :

$$S(i,j) = \sum_{m=1}^{M} \alpha_m \ \tilde{S}^{(m)}(i,j),$$

où $\tilde{S}^{(m)}(i,j)$ est la **version** normalisée intramodale de la similarité $S^{(m)}$, et α_m (tel que $\sum_m \alpha_m = 1$) reflète l'importance de chaque modalité. C'est un **mélange** convexe. On s'assure ainsi que la synergie finale $S(i,j)$ est **équilibrée**.

Plus **complexe**, on peut laisser α_m **varier** dans le temps, ou être optimisé par un mini-algorithme. Le **SCN** peut, par exemple, estimer qu'une modalité est moins fiable (captor bruité) et réduire α. Cette **auto-adaptation** augmente le **degré** de plasticité dans le DSL, puisqu'on ne se borne plus à un paramétrage statique.

C. Exemples de Cross-Modal Normalisation

Dans des embeddings textuels (ex. BERT), on emploie souvent la **similarité** cosinus, bornée entre $[-1,1]$. Pour un embedding image basé sur des distances euclidiennes, on la convertit en une similarité (ex. $S_{\text{img}} = \exp(-\alpha \| \mathbf{x}_i - \mathbf{x}_j \|)$) bornée entre $(0,1]$. On peut ensuite **rééchelonner** ces deux scores pour qu'ils aient des **moyennes** ou des **intervalles** comparables.

Dans certains flux (ex. intensités spectrales audio), on peut recourir à un **log** ou à une **sigmoïde** pour "compresser" l'éventail de valeurs. Par exemple,

$$\hat{S}^{(m)}(i,j) = \text{sigmoid}\left(\gamma\left(S^{(m)}(i,j) - \mu^{(m)}\right)\right),$$

de sorte à limiter les **extrêmes** et ramener le score dans $[0,1]$. Cela se révèle utile quand la **distribution** des mesures dans une modalité présente de larges écarts ou comporte des outliers.

D. Intérêt pour le DSL

Une fois la fonction de synergie $S(i,j)$ normalisée, la règle DSL

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$$

part d'une **base** plus fiable, on ne risque pas de voir une modalité "alpha" dominer simplement parce que ses valeurs S se situaient dans un registre d'amplitude plus élevé. Chaque canal **contribue** de façon plus proportionnée, ce qui **renforce** la cohérence et la **stabilité** des clusters multimodaux (chap. 8.9, 8.10).

Si, dans un flux, la similarité se situe **toujours** ~0.9 (surdimension), alors que dans un autre, elle varie ~0.2–0.5, le SCN penche en faveur du premier, aboutissant à un **biais**. Un calibrage cross-modal évite ces écarts amplifiés et assure une **convergence** plus homogène.

8.10.3. Sécurité et Biais

Au-delà des aspects d'intégration multimodale et d'optimisation, un **DSL** (Deep Synergy Learning) appliqué à des **flux** (texte, images, audio, etc.) doit aussi s'interroger sur la **sécurité** et la **neutralité** des données traitées. Les **biais** présents dans les corpus textuels, visuels ou sonores risquent de **déformer** la formation des clusters (donnant lieu à des regroupements stéréotypés ou injustes), tandis que des flux malveillants (ex. un **fake audio**) peuvent compromettre la **confiance** dans la synergie calculée.

8.10.3.1. Biais dans le texte (ou les images) → clusters déformés

Lorsqu'un **SCN** (Synergistic Connection Network) travaille sur un **DSL** (Deep Synergy Learning) multimodal, il s'appuie sur des **embeddings** ou des **représentations** sub-symboliques issus de **corpus** divers (textuels, visuels, etc.). Or, si les **données** textuelles ou visuelles sont entachées de **biais** (culturels, genrés, ethniques…), la **synergie** calculée $S(i,j)$ se retrouve biaisée. Le **DSL**, au fil de l'auto-organisation, construit alors des **clusters** potentiellement **déformés**, ancrant les stéréotypes du corpus dans la structure même du réseau. La présente section (8.10.3.1) explore la manière dont ces biais se manifestent et leurs conséquences sur la formation de **clusters**.

A. Origine des Biais dans les Données

De nombreux corpus textuels (provenant de l'internet, d'archives, de données historiques) véhiculent des **stéréotypes** ou des **asymétries**. Par exemple, "médecin" pourrait être corrélé à un genre masculin, "infirmière" à un genre féminin, ou d'autres liens discriminants relatifs aux métiers, aux groupes ethniques, etc. D'un point de vue **mathématique**, les **embeddings** (Word2Vec, GloVe, BERT…) enregistrent ce déséquilibre, aboutissant à des **similarités** entre mots reflétant des biais latents.

Dans les bases d'images (pour la **vision**), un **déséquilibre** peut se manifester dans la représentation de certains groupes (ex. plus d'images d'hommes que de femmes dans un métier donné, ou d'un groupe ethnique particulier). Un algorithme de **reconnaissance** ou d'**embedding** entraîné sur ces données assimile alors des notions partielles ou sur-spécifiques, impactant la **synergie** $S(\text{img}_i, \text{img}_j)$ et la façon dont il associe images à concepts.

Dans un SCN multimodal, ces biais "visuels" peuvent **déformer** l'idée qu'a le réseau de la diversité d'apparences, menant à des confusions ou à un alignement trop strict avec l'échantillonnage du dataset.

B. Impact sur la Formation des Clusters

Le **DSL** fonctionne en renforçant $\omega_{i,j}$ lorsque $S(i,j)$ est jugé élevé. Mais si $S(i,j)$ est déjà corrompu par un **biais** de l'embedding (par ex. "femme" corrélé à "infirmière" beaucoup plus qu'à "médecin"), alors le SCN va **créer** ou **stabiliser** un cluster qui associe "femme" et "infirmière", tout en écartant "femme" de "médecin". De même, le cluster "médecin-homme" peut se **cristalliser**, reproduisant un stéréotype.

À mesure que la **dynamique** DSL se déroule, ces "petits biais" dans S s'accumulent et **créent** des macro-nœuds reflétant la distribution biaisée du corpus plutôt que la **réalité** qu'on voudrait représenter. Le SCN peut ainsi *sur*-associer "ingénieur" à "homme" ou "voile" à une ethnie spécifique. Ces **déformations** sont d'autant plus graves qu'elles peuvent renforcer d'autres liaisons (effet de rétroaction), menant à des **clusters** encore plus stéréotypés (boucle d'amplification).

C. Exemples

Imaginons un SCN multimodal image–texte où "chien" est très souvent décrit par un mot "dog" associé à un adverbe "cute". Le SCN finira par sur-pondérer $\omega_{\text{"dog","cute"}}$. Mais si le corpus n'a que peu d'exemples de "chien" "agressif", l'association "dog–aggressive" reste faible, faussant la pluralité possible des chiens. Pire, s'il y a un biais "cat = female, dog = male" dans le corpus, le SCN entérine des **clusters** genrés d'animaux.

Dans un flux de visages, le SCN peut surreprésenter un certain groupe ethnique dans le cluster "client souriant", en ignorant d'autres facettes faute d'exemples ou d'embeddings adaptés. Les pondérations ω se hiérarchisent alors selon un prisme biaisé, privant le réseau d'une vision équitable.

D. Approches pour Détecter/Atténuer ces Biais

On peut **analyser** la matrice $\{\omega_{i,j}\}$ ou les macro-clusters issus du SCN en cherchant des indicateurs de regroupement stéréotypé. D'un point de vue **mathématique**, on peut définir un **indice** de concentration d'un certain groupe (genre, ethnie, etc.) dans un cluster particulier et voir s'il diverge de la distribution globale.

Une option consiste à **rebaisser** les scores $S(i,j)$ susceptibles de véhiculer un stéréotype, en introduisant un terme de "debias" :

$$S'(i,j) = S(i,j) - \Delta_{\text{bias}}(i,j).$$

Si $\mathcal{E}_i$ et $\mathcal{E}_j$ contiennent des attributs genrés, on peut injecter un correctif statistique pour diminuer la **corrélation** artificielle.

À un niveau plus **amont**, on peut **réentraîner** l'embedding (image, texte) sur un corpus plus équilibré ou recourir à des techniques d'**augmentations** de données pour réduire le biais au stade de l'ingestion. Cela diminue le besoin de correction tardive dans le SCN.

E. Enjeux Éthiques et Sociétaux

Le **biais** dans un SCN multimodal n'est pas qu'un souci technique, il peut consolider des représentations discriminantes, perpétuer des stéréotypes, influencer des **décisions** injustes (cf. chap. 8.10.3.2 peut-être). Sur le plan mathématique, le SCN n'est pas en tort s'il s'appuie sur un $S(i,j)$ biaisé par l'embedding, mais d'un point de vue **responsabilité** et **équité**, il revient à l'ingénierie du DSL de **détecter**, **comprendre** et **corriger** ces défauts.

8.10.3.2. Sécurité : si un flux audio est "fake", confusion potentielle ?

Dans un **SCN** (Synergistic Connection Network) multimodal où cohabitent des **flux** visuels (vidéo), **audio** (enregistrements sonores) et **textes** (sous-titres, annotations), la **qualité** et l'**authenticité** de chaque flux conditionnent la **fiabilité** de la **synergie** $S(i,j)$. Un flux **audio** "fake" (ex. deepfake vocal) risque de **leurrer** le SCN, car il peut être *artificiellement* cohérent avec un flux visuel ou textuel. Cette section (8.10.3.2) explore comment un audio "falsifié" peut semer la **confusion** dans la structure ω du DSL et quelles **approches** sécuritaires on peut envisager pour mitiger ce risque.

A. Contexte : le "fake audio" dans un DSL multimodal

Un **fake audio** consiste en un signal sonore **généré** ou **modifié** (ex. imitation d'une voix, ajout de propos inexistants) dans le but de **duper** un système. Sur le plan **mathématique**, l'embedding $\mathbf{v}_{\text{fake}}$ d'un tel flux peut **ressembler** à celui d'un audio légitime $\mathbf{v}_{\text{legit}}$ si l'algorithme de deepfake est efficace. Le **DSL**, dans sa phase de calcul $S(\text{aud}, \text{vid})$ ou $S(\text{aud}, \text{txt})$, peut alors **renforcer** $\omega_{\text{fake,autres}}$ à un haut niveau, croyant que la correspondance est réelle.

Si le **SCN** adopte la règle de mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$$

et que l'audio "fake" est **superficiellement cohérent** avec un flux vidéo (ex. un mouvement labial), la similarité $S(\text{fake}, \text{vid})$ sera jugée élevée et les liaisons ω s'intensifient. Cela conduit le SCN à intégrer cette entité audio "fake" au sein d'un **cluster** multimodal (e.g., un macro-nœud représentant la scène ou le locuteur). La **structure** du DSL s'en voit altérée, ce qui peut détériorer la **reconnaissance** d'événements, car un flux sonore trompeur est pris en compte. Cela peut également induire une **fausse** correspondance entre la vidéo et le contenu sonore, tout en contaminant le flux textuel, par exemple si l'audio falsifié est aligné à des sous-titres fallacieux.

B. Mécanismes : confusion potentielle dans la synergie

Le DSL renforce $\omega_{\text{fake,legit}}$ dès lors que $S(\text{fake}, \text{legit})$ est élevé, ce qui peut arriver si l'embedding audio "fake" $\mathbf{v}_{\text{fake}}$ se trouve **proche** de $\mathbf{v}_{\text{locuteur}}$ dans l'espace des features (ex. même timbre, même intonation). Cela **leurre** la règle DSL en faisant croire à un score de similarité fort. L'**auto-organisation** se base sur un "garbage in" partiel, provoquant un cluster "faux" où la voix simulée est acceptée comme partie légitime de la scène.

Une fois que $\omega_{\text{fake,locuteur}}$ est fort, l'entité audio "fake" reçoit plus de **validité** dans le SCN, influençant la construction de macro-nœuds (chap. 8.6). Si, par exemple, ce locuteur est lié à un sous-titre **texte**, l'audio "fake" peut s'immiscer dans un cluster "Texte–Image–Audio" consolidé. Il en résulte une "**confusion**". Le flux "fake" se retrouve **validé** par le réseau, induisant potentiellement des **actions** erronées en aval (ex. classification trompée, décision erronée).

C. Pistes de Sécurisation et Contrôle

Une première **défense** consiste à exiger plus qu'une simple correspondance audio–vidéo. On multiplie les **tests** de cohérence (contrôle labial précis, tonalité vocale), ou on confronte le flux audio à d'autres indicateurs (cf. recuit simulé, chap. 7.3). Si une entité "fake" ne parvient pas à maintenir une cohérence **simultanée** avec d'autres flux ou d'autres moments du réseau, la synergie chute, évitant la consolidation du lien ω.

Sur le plan **mathématique**, on peut intégrer un module "**score** de suspicion" $\text{suspect}(\mathcal{E}_{\text{fake}}) \in [0,1]$ dans la fonction de **synergie** :

$$S'(\text{fake}, j) = \big(1 - \text{suspect}(\text{fake})\big) \times S(\text{fake}, j).$$

Si on détecte (via un algorithme externe) un fort risque de deepfake, $\text{suspect}(\text{fake}) \approx 1$, annihilant la pondération ω. Cela **empêche** l'entité falsifiée de piéger le SCN.

Si l'on relève une incohérence à un certain stade (ex. la vidéo indique un locuteur différent de ce que l'audio signale), on peut **amplifier** l'inhibition (chap. 7.4) dirigée contre ce flux audio suspect. Dans ce cas, le DSL diminue rapidement les liaisons $\omega_{\text{fake},\cdot}$.

8.10.4. Recherche Future

L'exploration du **DSL** (Deep Synergy Learning) multimodal ne se limite pas aux démos ou prototypes actuels ; il ouvre un vaste champ pour des **déploiements** plus ambitieux, tant en volume de données qu'en diversité sensorielle. Dans cette sous-section (8.10.4.1), nous soulignons la **perspective** d'appliquer le DSL à des **datasets massifs**, comportant potentiellement des **millions** de vidéos ou de documents, et d'y gérer des **synergies** à une échelle inédite.

8.10.4.1. Appliquer le DSL multimodal à de très grands datasets (millions de vidéos / documents)

Dans un **SCN** (Synergistic Connection Network) mis en œuvre pour un **DSL** (Deep Synergy Learning) multimodal, le **nombre** d'entités $\{\mathcal{E}_i\}$ peut exploser dès qu'on envisage des **datasets** à l'échelle de millions (ou dizaines de millions) d'éléments, par exemple dans la gestion de **vidéos** massives (YouTube, TikTok) ou de **documents** textuels volumineux. Cette section (8.10.4.1) aborde :

- Les **défis** de l'échelle et de la complexité,
- Les **infrastructures** HPC (High-Performance Computing) ou de calcul distribué qui s'imposent,

- Les **approches** hybrides et approximatives permettant de maintenir la logique DSL sans être submergé par un coût $O(n^2)$.

A. Échelle et Complexité

Lorsque l'on traite des **millions** de vidéos (ou documents), on se retrouve avec un **nombre** d'entités $n \approx 10^6$, voire plus. Le **SCN** stocke $\{\omega_{i,j}\}$, et le nombre de paires (i,j) peut monter à $O(n^2) \approx 10^{12}$. Cette **matrice** ω est impossible à manipuler pleinement en mémoire standard, et la mise à jour DSL en $O(n^2)$ par itération devient irréalisable.

Pour éviter cet écueil, on adopte des stratégies de **sparsification** (k plus proches voisins, gating de clusters) qui **limitent** la mise à jour $\omega_{i,j}$ aux seules paires (ou sous-blocs) jugées pertinentes. Les sections précédentes (8.10.1.2) évoquent ces heuristiques. Sur le plan **mathématique**, cela ramène le coût à $O(n\,k)$ ou $O(n\log n)$ selon l'indexation, plus gérable pour de larges n.

B. Infrastructure et HPC

Traiter des **millions** de vidéos/documents implique souvent d'avoir accès à des **clusters** HPC (High-Performance Computing) ou du **cloud** distribué. Cela requiert un **partitionnement** de la grande **matrice** ω :

- On divise l'ensemble d'entités $\{1, \dots, n\}$ en $\{\mathcal{V}_1, \dots, \mathcal{V}_m\}$.
- Chaque partition $\mathcal{V}_p$ gère localement $\omega_{i,j}$ pour $(i,j) \in \mathcal{V}_p$.
- Les paires $\left(i \in \mathcal{V}_p, j \in \mathcal{V}_q\right)$ supposent une **synchronisation** inter-bloc (messagerie, verrous épisodiques).

D'un point de vue **théorique**, on peut modéliser cela comme des **sous-graphes** partiels, rassemblés périodiquement. Du fait de la nature **distribuée**, la mise à jour DSL devient asynchrone ou semi-synchrone, et la convergence doit être considérée dans un cadre distribué (chap. 7.5 sur la parallélisation).

Une autre manière de décomposer le traitement est de **segmenter** (batcher) la base en sous-lots plus petits (ex. 10^4 entités). On applique la mise à jour DSL localement sur chaque lot, puis on fusionne les résultats ou on effectue une étape de **raccord**. Cela introduit une forme de **mini-batch** auto-organisé, analogue à ce qui se pratique dans l'apprentissage profond classique.

C. Approches Hybrides

Si le dataset provient d'un **flux** continu (ex. nouveau contenu vidéo chaque jour), on mélange :

- La **segmentation** en mini-lots (batches),
- La **recherche** de voisins (k-NN) restreinte,
- L'**insertion incrémentale** (voir chap. 9.1 sur la dynamique en flux).

Ce dispositif garantit qu'on ne fasse **jamais** d'itération $O(n^2)$ complète, mais plutôt un ensemble d'opérations locales $O(n\,k)$.

Une pratique courante consiste à définir un **seuil** θ dans l'espace d'embeddings de la modalité., tel que

$$S(i,j) = \begin{cases} \text{compute}, & \text{si } \| \mathbf{x}_i - \mathbf{x}_j \| \leq \theta, \\ 0, & \text{sinon.} \end{cases}$$

Cela évite de calculer des distances $\| \mathbf{x}_i - \mathbf{x}_j \|$ pour toutes paires, et impose qu'on stocke $\omega_{i,j}$ seulement si $\| \mathbf{x}_i - \mathbf{x}_j \| \leq \theta$. Dans la pratique, on peut user de techniques d'**approximate nearest neighbor** (ANN, hashing, etc.) pour localiser les candidats (j) dans une boule $\| \mathbf{x}_i - \mathbf{x}_j \| \leq \theta$.

D. Potentiel en Recherche

Du point de vue **théorique**, on peut étudier le comportement d'un SCN à **millions** d'entités en s'inspirant d'**objets** comme les **graphons** en théorie des graphes à grande échelle, ou en adoptant des arguments de **limite** $n \to \infty$. On examinerait alors la **formation** de clusters comme une transition de phase, avec un niveau de similarité moyen.

On se demande aussi comment, mathématiquement, la **fusion** des canaux (image–audio–texte) demeure stable ou cohérente quand n grossit. S'il existe un flux surreprésenté, peut-il dominer ? Des formalismes de grande dimension montrent que certains canaux peuvent saturer la matrice ω si on ne prévient pas la surexposition.

E. Exemple de Scénario

Supposons qu'on veuille **analyser** 10^6 vidéos courtes, par exemple des extraits de **TikTok** ou **YouTube Shorts**. On **extrait** pour chaque vidéo un embedding visuel $\mathbf{x}_i^{(\text{vid})}$, un embedding audio $\mathbf{x}_i^{(\text{aud})}$ et éventuellement un embedding textuel $\mathbf{x}_i^{(\text{txt})}$. On se retrouve ainsi avec $n = 10^6$ entités vidéo, audio ou texte, voire plus. Afin d'éviter une complexité de $0(n^2) = 10^{12}$ calculs par itération, une **heuristique** de type k-NN est appliquée, restreignant les connexions de chaque entité à ses $k = 100$ plus proches voisins. La mise à jour est alors distribuée sur un cluster HPC, chaque nœud traitant un sous-bloc, tandis que la partie inter-bloc des pondérations ω est synchronisée régulièrement. Après un certain nombre d'itérations, la structure **auto-organisée** révèle des **clusters** ou **macro-nœuds** $\{1, \dots, C\}$ qui regroupent des vidéos aux thématiques similaires, telles que "sports", "comédie" ou "vlog personnel", identifiables par la convergence locale des pondérations ω.

8.10.4.2. Fusion Multi-Sensorielle en Robotique plus Avancée (Capteurs LiDAR, etc.)

La **fusion** de multiples **capteurs** (caméras, audio, LiDAR, etc.) constitue un volet crucial de la **robotique** moderne, où la diversité de sources (capteurs de profondeur, inertiels, radars, etc.) accroît la **richesse** des informations mais soulève des **défis** d'intégration. Au sein d'un **SCN** (Synergistic Connection Network) piloté par un **DSL** (Deep Synergy Learning), les **entités** issues de capteurs variés peuvent être reliées par des **pondérations** ω, ajustées en continu par la **synergie** S. La présente section (8.10.4.2) s'attache plus spécifiquement à la **fusion** impliquant un **capteur LiDAR** (Light Detection and Ranging), soulignant ses apports et la manière dont un SCN multimodal peut en tirer parti pour une **compréhension** plus fine de l'environnement.

A. Rappels et Enjeux de la Fusion LiDAR

Le **LiDAR** envoie des impulsions laser pour mesurer la distance des obstacles via leur temps de vol. On obtient ainsi un **nuage de points** 3D décrivant la géométrie de l'environnement autour du robot. Sur le plan **mathématique**, on peut représenter chaque **scan** LiDAR comme

un ensemble $\{\mathbf{p}_k\} \subset \mathbb{R}^3$. Pour alimenter un **SCN**, il est possible de **définir** une **entité** $\mathcal{E}_i$ comme un cluster ou un voxel parmi ces points, en fonction d'une segmentation ou d'une agrégation 3D spécifique. Une fois cette entité formée, on lui associe des **features** pertinentes telles que la **position moyenne**, la **normale** dominante ou encore la **densité** de points, permettant ainsi d'enrichir la représentation et d'optimiser la synergie entre les différentes entités du réseau.

Un capteur LiDAR se révèle **complémentaire** des caméras (2D) en fournissant un **relief** 3D précis, tandis que l'image apporte la **couleur** ou la **texture**.

Le **DSL** (Deep Synergy Learning) opère sur un **réseau** $\{\omega_{i,j}\}$, où $\omega_{i,j}$ relie deux **entités** (par ex. segment LiDAR vs. patch visuel). La **synergie** S quantifie leur cohérence spatiale ou sémantique (ex. un nuage 3D localisé devant la caméra correspond à la forme perçue en 2D). L'**auto-organisation** $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right]$ renforce ou affaiblit ces liens en continu, permettant au robot de **consolider** une représentation multimodale plus fiable.

B. Modélisation Mathématique de la Synergie LiDAR + Autres Capteurs

Plutôt que de traiter directement des milliers ou millions de points **LiDAR**, on effectue une **segmentation** 3D à l'aide d'algorithmes comme **DBSCAN** ou **RANSAC**, permettant d'identifier des volumes sous forme de clusters ou de surfaces homogènes. Une fois cette segmentation réalisée, on procède à une **agrégation** en entités $\{\mathcal{E}_i^{(\mathrm{LiDAR})}\}$, où chaque entité représente un **objet partiel** ou un **voxel** volumique, facilitant ainsi l'analyse et l'exploitation des données au sein du **SCN**.

Chacune de ces entités $\mathcal{E}_i$ dispose d'un vecteur $\mathbf{x}_i \in \mathbb{R}^d$ (features géométriques). Un algorithme plus fin peut extraire la **normale moyenne**, la **couleur estimée** (si un alignement avec la caméra est disponible), etc.

Si un robot embarque à la fois un LiDAR et une caméra, on peut définir :

$$S(\mathcal{E}_{\mathrm{LiDAR}}, \mathcal{E}_{\mathrm{cam}}) = \alpha\ \mathcal{C}_{\mathrm{proj}}(\mathcal{E}_{\mathrm{LiDAR}}, \mathcal{E}_{\mathrm{cam}}) + \beta\ \mathcal{C}_{\mathrm{color}}(\mathcal{E}_{\mathrm{LiDAR}}, \mathcal{E}_{\mathrm{cam}}) + \cdots$$

– $\mathcal{C}_{\mathrm{proj}}$ estime la **cohérence** spatiale en vérifiant que la projection 3D du cluster LiDAR vers l'image coïncide avec la forme 2D détectée.

– $\mathcal{C}_{\mathrm{color}}$ compare éventuellement la **colorimétrie** si on a mappé la texture de la caméra sur le nuage LiDAR, etc.

Le **SCN** exploitera cette synergie pour **renforcer** ω entre entités "correspondantes" (même objet capté par la caméra et le LiDAR) et **éliminer** celles qui ne correspondent pas (faible S).

On peut étendre la logique à d'autres **capteurs**, en intégrant notamment des sources **audio**, où l'emplacement sonore est estimé par **beamforming**, ou en exploitant la position 3D d'un événement capté par **LiDAR**. De même, les capteurs **inertiels** (**IMU**) permettent d'évaluer la **pose** d'un robot et d'en déduire un **alignement spatio-temporel**, modélisé par une synergie $S(\mathrm{LiDAR}_{t+1}, \mathrm{LiDAR}_t)$ qui exprime la continuité entre deux instants successifs. Dans chaque cas, le **DSL** établit et ajuste dynamiquement les **liaisons** ω afin de renforcer la **cohérence** du réseau et d'optimiser l'agrégation des signaux issus de diverses modalités.

C. Avantages et Problématiques de la Fusion LiDAR + DSL

L'**auto-organisation** DSL procure une **flexibilité** en évitant une procédure figée de fusion. Au lieu d'une projection LiDAR vers caméra prédéfinie, le **réseau** ajuste dynamiquement ω si la configuration spatiale évolue ou si une partie du nuage LiDAR ne trouve pas d'équivalent

visuel. Les entités $\mathcal{E}_i$ LiDAR peuvent être partiellement **isolées** (faible ω vers les caméras), ou au contraire, fortement liées si l'**image** confirme l'objet 3D.

Inévitablement, un **nuage** LiDAR volumineux (plusieurs dizaines de milliers de points par scan) engendre beaucoup d'entités dans le SCN. Cela conduit à privilégier des approches telles que la **segmentation** ou la **voxelisation**, permettant de réduire la **granularité** des données et d'optimiser leur traitement. Une autre stratégie repose sur l'usage d'**heuristiques** basées sur les **k plus proches voisins** (k-NN), comme discuté en détail dans le **chapitre 8.10.1.2**. L'objectif est d'éviter un calcul exhaustif des pondérations ω entre chaque segment issu du **LiDAR** et l'ensemble des segments issus de la **caméra**, ce qui serait trop coûteux en termes de complexité computationnelle.

En **robotique**, le LiDAR scanne l'environnement en continu, donnant naissance à des entités $\{\mathcal{E}_{(\text{LiDAR},t)}\}$. Le DSL agit donc de manière **incrémentale** (chap. 9.1), chaque nouveau scan actualisant ω. Sur le plan **mathématique**, la mise à jour $\omega_{(n+1),\,j} \leftarrow \omega_{(n+1),\,j} + \cdots$ s'applique à un ensemble restreint de voisins si on adopte une approche k-NN ou un gating.

D. Extensions : Surfaces 3D et Sémantique Avancée

Le LiDAR est parfois converti en **maillage** (mesh) ou en **surface** polygonale. Dans un SCN, on peut représenter ces **surfaces** comme des entités plus haut niveau, facilitant la correspondance géométrique avec la caméra. La **synergie** $S(\text{mesh}, \text{cam})$ se définit alors via la superposition 2D/3D.

Par ailleurs, le robot peut disposer de **modèles** de reconnaissance (objets, classes sémantiques) alimentant le SCN d'un volet "symbolique" (chap. 8.7.4). On associe à chaque **cluster** LiDAR un label potentiel (ex. "voiture stationnée"). Le DSL vérifie la **cohérence** de ce label via la caméra ou la carte textuelle.

8.11. Conclusion et Ouverture

Après avoir parcouru ce chapitre 8 consacré au **DSL multimodal**, nous arrivons maintenant à la **conclusion** et aux pistes d'ouverture. L'objectif était de montrer comment le **Deep Synergy Learning** s'étend à des flux variés (images, audio, texte...), tout en préservant la philosophie d'**auto-organisation** au sein du **SCN** (Synergistic Connection Network). Dans ce qui suit (8.11.1 et 8.11.2), nous récapitulons les points marquants du chapitre, puis nous anticipons les liens avec les chapitres ultérieurs (9 et 10), avant de souligner la valeur générale du DSL multimodal (8.11.3).

8.11.1. Récapitulatif du Chapitre

8.11.1.1. On a défini comment le DSL peut s'étendre à plusieurs modalités (image, audio, texte)

Au terme de l'exploration menée dans ce **chapitre 8**, nous avons montré que le **DSL** (Deep Synergy Learning), jusque-là appliqué à des environnements mono-modaux (par exemple, seulement images ou seulement texte), peut être **élargi** à des cadres où **plusieurs modalités** — visuelles, auditives, textuelles, voire d'autres types de capteurs — coexistent et sont traitées au sein d'un **SCN** (Synergistic Connection Network) unifié. Cette extension repose sur plusieurs aspects fondamentaux garantissant la robustesse et l'adaptabilité du modèle. Tout d'abord, elle intègre des **fonctions de synergie** spécifiques ou hybrides, permettant d'évaluer la **pertinence** et la **compatibilité** entre des entités provenant de différents canaux, qu'il s'agisse de relations image–image, image–texte ou encore audio–texte. Ensuite, la **dynamique** du DSL suit une règle de mise à jour uniforme, définie par

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta \left[S(i,j) - \tau\, \omega_{i,j}(t) \right]$$

indépendamment de la modalité des entités concernées. Si la synergie $S(i,j)$ est jugée élevée, alors la pondération $\omega_{i,j}$ se trouve renforcée, tandis qu'en cas de faible synergie, elle décroît progressivement. Enfin, l'ensemble du réseau conserve une **cohérence** globale grâce aux mécanismes d'**inhibition**, détaillés dans le **chapitre 7.4**, ainsi qu'à la possibilité d'**insertion** incrémentale, discutée dans le **chapitre 9**. De plus, des techniques de **calibrage** des embeddings, présentées dans le **chapitre 8.10.2**, permettent d'assurer que les différentes modalités restent alignées et comparables au sein du SCN.

Cette conclusion met en évidence la **généralité** de l'approche DSL. Son cadre synaptique local, basé sur l'évolution des **pondérations** ω et la mesure de **synergie** S, s'adapte de manière **naturelle** à des **sources** de données variées. L'un des aspects clés de cette flexibilité réside dans la **définition appropriée** de la **fonction** $S(\mathcal{E}_i, \mathcal{E}_j)$, qui doit refléter la correspondance ou la distance entre entités, même lorsqu'elles appartiennent à des **espaces** ou **dimensions** fondamentalement dissemblables.

A. Principe d'Intégration Multimodale

L'**intégration multimodale** repose sur le fait qu'un **SCN** ne se limite plus à des entités issues d'un **espace unique** $\mathbb{R}^d$. Désormais, les entités **visuelles** $\mathcal{E}_i^{(\text{img})}$ sont représentées par un **embedding visuel** $\mathbf{v} * i \in \mathbb{R}^{d*\text{img}}$, tandis que les entités **sonores** $\mathcal{E}_j^{(\text{aud})}$ sont décrites par un **embedding acoustique** $\mathbf{a} * j \in \mathbb{R}^{d*\text{aud}}$. De même, les entités **textuelles** $\mathcal{E}_k^{(\text{txt})}$ possèdent une

représentation linguistique sous la forme d'un **embedding sémantique t** $* k \in \mathbb{R}^{d*\text{txt}}$. D'autres modalités, telles que les **capteurs LiDAR** (voir chap. 8.10.4.2) ou les **représentations symboliques** (voir chap. 8.7.4), peuvent être incorporées à cette structure.

Le **DSL** organise alors un **réseau de pondérations** $\omega_{i,j}$ qui connecte ces entités, sans nécessiter un **alignement strict** de leurs espaces respectifs. Pour ce faire, il utilise des **fonctions de synergie** adaptées à chaque modalité, telles que la **similarité cosinus**, la **distance exponentielle**, ou encore une **co-occurrence probabiliste**. En présence de plusieurs modalités combinées, il est possible de recourir à un **assemblage pondéré** sous la forme d'une combinaison linéaire $\alpha, S_{\text{vis}} + \beta, S_{\text{aud}}$, permettant ainsi de moduler l'impact de chaque **canal d'information** selon son importance relative au sein du réseau.

Sur le plan **mathématique**, la même règle de descente

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S\left(\mathcal{E}_i, \mathcal{E}_j\right) - \tau\, \omega_{i,j}(t)\right]$$

assure que, si plusieurs modalités **confirment** une association (par ex. l'image d'un chat, l'audio "meow", le mot "cat"), le **score** S en devienne élevé, renforçant $\omega_{i,j}$. Ainsi, un **cluster** auto-organisé se forme, réunissant entités $\{\text{img\_chat}, \text{audio\_miaulement}, \text{texte\_cat}\}$. D'autres liaisons (incohérentes) finissent par s'**étioler** (pondérations retombant vers 0).

B. Mise en Œuvre au Niveau du SCN

Le **SCN** demeure une **matrice** $\{\omega_{i,j}\}_{1\le i,j\le n}$. La "nouveauté" réside dans le fait que $S(i,j)$ prend en compte l'identifiant de la modalité. Lorsque deux entités appartiennent au même canal, comme image–image ou audio–audio, la synergie S est calculée selon les méthodes classiques d'un **DSL mono-modal**, en fonction de la distance dans l'espace des embeddings ou d'autres critères de similarité propres à la modalité. En revanche, lorsqu'il s'agit d'une comparaison **cross-modale** (par exemple, image–texte ou audio–texte), on utilise une fonction S spécifique et adaptée à cette correspondance, telle que

$$S\left(\text{image}_i, \text{texte}_j\right) = f\left(\cos\left(\mathbf{v}_i^{(\text{img})}, \mathbf{v}_j^{(\text{txt})}\right)\right)$$

ou bien un mélange convexe, comme décrit dans la section **8.10.2.2**, lorsque plusieurs descripteurs sont disponibles pour affiner la mesure de synergie entre les modalités.

Une fois la mise à jour enclenchée, le **DSL** repère **naturellement** les associations fréquentes ou récurrentes. Un *macro-nœud* se construit progressivement, regroupant un sous-ensemble d'images, de segments audio et de tokens textuels qui partagent une forte synergie. Sur le plan **algorithmique**, on finit par voir un **bloc** dans la matrice ω où $\omega_{i,j}$ est élevé, indiquant une synergie inter-modalité solide (ex. "chien + aboiement + mot dog + mot puppy").

C. Gestion des Conflits et Incohérences

Il arrive que le SCN détecte une synergie contradictoire (ex. un segment audio "oiseau" mal aligné sur une image "voiture"). Si la **valeur** S reste faible ou fluctuante, la pondération $\omega_{i,j}$ ne se consolide pas. L'**auto-organisation** agit comme un **filtre** inhibant la croissance de liens incohérents.

Les règles d'**inhibition** (chap. 7.4) permettent de limiter la **prolifération** de liaisons entre modalités trop hétérogènes. S'il s'avère qu'un flux audio "n'a rien à voir" avec l'image, $\omega_{i,j}$ se voit réduit, évitant la formation d'un cluster erroné.

D. Pertinence pour l'Analyse et la Classification

Exemples d'applications

- **Indexation multimédia** : on peut ranger ou annoter des **vidéos** selon les sons correspondants et les mots-clés détectés. Le SCN classe naturellement ce qui converge (image + sous-titres + audio).
- **Recherche** cross-modal : on cherche dans des documents la correspondance (texte–image). Un SCN fusionne les entités, créant des macro-nœuds signant la parenté sémantique.
- **Clustering** auto-organisé : un ensemble de contenus (podcasts, transcriptions, images d'illustration) se groupent par thèmes émergents.

Au plan **théorique**, la **convergence** se comprend comme la minimisation d'une fonction d'énergie $\mathcal{J}(\omega)$ (chap. 7.2), où l'on pèse la **synergie** $S(i,j)par\omega i,j\omega_{i,j}$. Les entités dont la **similarité** (même modalité) ou **complémentarité** (modalités différentes, mais sémantiquement alignées) est forte forment un **minimum local** stable, c'est-à-dire un **cluster** consolidé dans la matrice $\{\omega_{i,j}\}$.

8.11.1.2. On a vu la dynamique auto-organisée gérer des liens hétérogènes, créant des clusters multimodaux

Au fil du **chapitre 8**, nous avons exploré la façon dont un **DSL** (Deep Synergy Learning) — appliqué à un **SCN** (Synergistic Connection Network) — traite simultanément des entités provenant de **différentes modalités** (image, audio, texte, etc.) en s'appuyant sur la **même** dynamique ω. Le **résultat** essentiel est la capacité à gérer des **liens** hétérogènes tout en **faisant émerger** des **clusters** multimodaux, c'est-à-dire des sous-groupes où des entités de canaux dissemblables (vision, son, mots, etc.) se retrouvent rassemblées selon leur **synergie**. La présente section (8.11.1.2) récapitule les aspects fondamentaux de cette auto-organisation multimodale et insiste sur l'hétérogénéité des liens traités par la mise à jour DSL.

A. Hétérogénéité des Liens et Mesures de Synergie

Un **SCN** multimodal regroupe des **entités** $\{\mathcal{E}_i\}$ issues de canaux variés, par exemple :

- $\mathcal{E}_i^{(\text{img})}$ pour les **images**,
- $\mathcal{E}_j^{(\text{aud})}$ pour l'**audio**,
- $\mathcal{E}_k^{(\text{txt})}$ pour du **texte**,
- Ou encore d'autres capteurs (LiDAR, signaux biométriques…).

Les **liens** $\omega_{i,j}$ reliant ces entités diffèrent selon qu'on considère un **couple** image–image, image–texte, audio–texte, etc. La **synergie** $S(i,j)$ doit donc s'adapter à la **nature** des deux modalités. Par exemple, pour audio–audio, on pourra prendre un score de corrélation spectrale ; pour image–texte, un embedding cross-modal ; pour texte–texte, une similarité sémantique, etc.

Formellement, si l'on note $\mathcal{M}$ l'ensemble des **modalités** (image, audio, texte, etc.), pour deux entités $\mathcal{E}_i^{(\alpha)} \in \alpha$ et $\mathcal{E}_j^{(\beta)} \in \beta$ (avec $\alpha, \beta \in \mathcal{M}$), la **fonction** $S^{(\alpha,\beta)}(i,j)$ mesure leur **pertinence**. On insère alors la règle DSL :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S^{(\alpha,\beta)}(i,j) - \tau\,\omega_{i,j}(t)\right].$$

Même si S varie d'une paire de modalités à l'autre, la **dynamique** de mise à jour ω (fondée sur η et τ) demeure identique et **auto-organisée**.

B. Émergence de Clusters Multimodaux

Grâce à cette **auto-organisation** hétérogène, le **SCN** peut relier un **segment audio** à une **image** s'ils sont cohérents (ex. le son "aboiement" correspond à la scène "chien"), ou un fragment **texte** décrivant la même scène. Ces liens ω se renforcent, menant à un **cluster** où cohabitent entités image–audio–texte.Ce cluster se révèle **multimodal**, traversant les différents canaux et reflétant une **unité sémantique**, un concept, une scène ou un objet.

Les liens ne s'appuient pas seulement sur la **similarité** (proches dans la même modalité), mais aussi sur la **complémentarité** (par ex. le texte et l'image se répondent, ou l'audio "coïncide" avec la vidéo). Le DSL autorise cette **mixité** car S peut intégrer des scores de **co-occurrence**, de **correspondance** spatiale ou temporelle, etc.

C. Conséquences sur la Dynamique Auto-Organisée

En prenant en compte plusieurs types de liens (image–image, image–audio, audio–texte, etc.), le **SCN** acquiert une **structure** plus dense, où chaque entité peut être reliée à d'autres d'une **modalité** différente. Les **clusters** qui en émergent ne sont plus purement "une classe d'images" ou "un ensemble de phrases", mais **un** sous-groupe **multi-canaux** (chap. 8.10.4.2 sur la robotique, par exemple).

Cette capacité à gérer la **diversité** des liens donne au DSL plus de **robustesse**, car si un flux, comme l'audio, se dégrade ou se révèle insuffisant, la complémentarité d'autres flux, tels que le texte ou la vision, peut **soutenir** la cohérence du cluster. Par ailleurs, ce mécanisme favorise la **découverte** de nouvelles associations "long-courrier" (ex. on détecte qu'un son "type jazz" se recoupe souvent avec des images de "concert nocturne"), renforçant une connaissance globale du contenu.

8.11.2. Liens vers Chapitres Suivants

L'exploration que nous avons menée dans ce **Chapitre 8** sur le **DSL multimodal**, traitant de la fusion entre la vision, le langage, les sons et d'autres modalités, trouve des prolongements naturels dans les **chapitres** qui suivent. Nous examinerons la capacité du **DSL** à évoluer dans des contextes encore plus dynamiques dans le **Chapitre 9**, ainsi que la mise en œuvre de **boucles de feedback** plus élaborées dans le **Chapitre 10**.

8.11.2.1. Chap. 9 : Évolutions Temps Réel et Apprentissage Continu

Après avoir établi, dans le **Chapitre 8**, les fondements d'un **SCN** (Synergistic Connection Network) étendu à plusieurs **modalités** (image, audio, texte, etc.), la suite logique consiste à aborder le cas d'un **environnement** en constante évolution, où de **nouvelles** entités

multimodales apparaissent (ou d'anciennes disparaissent). Le **Chapitre 9** traite précisément des mécanismes d'**apprentissage continu**, d'**insertion** de nouvelles données, et de **dynamique** jamais totalement figée dans le temps. Cette section (8.11.2.1) introduit la transition vers ce prochain chapitre en rappelant les grands enjeux et l'esprit des méthodes qui y seront discutées.

A. Flux Dynamiques et Contexte Évolutif

Lorsque l'on observe un **SCN** soumis à un **flux** incessant (comme un **stream** vidéo, audio, textuel, ou un ensemble hétéroclite de capteurs), le nombre d'**entités** $\mathcal{E}_i$ croit au fil du temps, ou certaines entités sont jugées obsolètes et retirées. Sur le plan **mathématique**, la pondération $\omega_{i,j}(t)$ n'atteint pas nécessairement un **équilibre** final, mais se **modifie** constamment à chaque "tick" ou à chaque **itération** où de nouvelles entités arrivent. Le **Chapitre 9** se focalise donc sur :

- Des **protocoles** d'insertion incrémentale (addEntity($\mathcal{E}_{\text{new}}$)) dans le réseau,
- Des **stratégies** de mise à jour **online**, permettant d'éviter la nécessité de tout recalculer,
- Des **mécanismes** de "recuit local" ou de "mini-bursts de bruit" quand on craint une "cristallisation" trop rapide du SCN.

Ces aspects soulignent la **flexibilité** d'un **DSL**, qui ne se limite pas à une unique passe d'apprentissage mais entretient un **processus** continu, capable de gérer la **distribution** variable des flux sensoriels au fil du temps.

B. Adaptation Incrémentale pour le Multimodal

Les **scénarios** multimodaux "réels" impliquent souvent des **sources** distinctes qui arrivent à des rythmes différents. Par exemple, on reçoit 25 images par seconde pour la vidéo et un flux audio à 16 kHz, tandis que des blocs textuels peuvent être déclenchés de manière asynchrone en fonction d'événements spécifiques. Chaque **entité** nouvellement créée (frame, segment audio, chunk textuel) doit alors :

- **Déterminer** un **voisinage** dans le SCN existant (chap. 8.10.1.2 sur k-NN ou gating),
- **Calculer** la synergie $S(\text{new}, j)$ vis-à-vis de ces voisins,
- **Actualiser** $\omega_{\text{new},j}$ localement (insertion incrémentale),
- Éventuellement, **réajuster** quelques liens existants si la nouvelle entité modifie le paysage sémantique.

Le **Chapitre 9** décrit ces algorithmes incrémentaux, établissant les conditions pour **maintenir** la **cohérence** multimodale malgré l'arrivée d'entités multiples.

C. Intégration avec la Multi-Échelle

Nous avons précédemment (Chap. 8) évoqué la **fusion** à différents **niveaux** (features brutes, embeddings plus abstraits, concepts symboliques). Dans un environnement qui **change** en continu, il importe de **préserver** cette structure multi-niveau sans qu'un réapprentissage intégral (coût prohibitif) ne soit imposé à chaque nouveau lot de données. Le **Chapitre 9** montrera comment :

- On peut **gérer** un SCN "ouvert" où la **dynamique** ω s'élargit aux entités apparues plus tard.

- On évite la redondance ou la saturation via des **mécanismes** de suppression, d'inhibition (ex. liens sous un seuil),
- La **synergie** multimodale demeure un **fil** conducteur pour relier (ou non) les nouvelles entités.

D. Perspective

Le **DSL** multimodal gagne en **robustesse** lorsque, au fil du temps, il est capable d'adopter un **paradigme** d'**apprentissage continu**. Le réseau ne se fige pas, mais *se met à jour* avec un flux incluant la vidéo, l'audio, le texte ou des capteurs multiples. Du point de vue **mathématique**, on peut voir cela comme une **suite** $\{\omega(t)\}_{t\in\mathbb{N}}$ potentiellement infinie, où l'équilibre local évolue constamment avec les nouvelles **données**. Les petits **recuits** ou "mini-bursts de bruit" parfois introduits aident à éviter les **minima** locaux trop rigides. Le **Chapitre 9** discutera en détail de ces procédures, prolongeant la théorie multimodale du Chap. 8 par une **dimension** de **flux temps réel** et d'**adaptation** permanente.

8.11.2.2. Chap. 10 : Feedback Coopératif dans le DSL

Les sections précédentes (Chap. 8 sur la multimodalité, Chap. 9 sur l'apprentissage continu) ont montré comment un **SCN** (Synergistic Connection Network) peut intégrer **plusieurs modalités** (images, textes, audio, capteurs divers) et gérer l'**arrivée** ou la **disparition** d'entités en temps réel. Le **Chapitre 10** se consacre à un **nouveau** niveau de complexité. Il traite de la **coopération** entre sous-SCN ou sous-modules via des **boucles** de **rétroaction** (feedback), permettant aux modalités de **s'influencer** mutuellement et d'obtenir une **harmonisation** globale plus fine qu'une simple juxtaposition. Le présent paragraphe (8.11.2.2) introduit la **logique** de ces "feedbacks coopératifs" et la façon dont ils s'inscrivent dans la dynamique du **DSL** (Deep Synergy Learning) multimodal.

A. Circulation de l'Information Multimodale et Boucles de Rétroaction

Dans un **DSL** multimodal, il est fréquent de disposer de **sous-SCN** ou de "modules" dédiés à chaque modalité. Un module **image**, un module **audio**, un module **texte**, etc. sont ainsi définis, chacun opérant sur ses propres entités et liens internes. Chacun opère sa propre **auto-organisation** locale (calcul de synergies S "intra-modal" et mise à jour des liaisons ω correspondantes). Cependant, ces modules ne travaillent pas en vase clos. Un **méta-niveau** ou un **super-nœud** global peut réunir leurs résultats, notamment les clusters partiels, les scores et les pondérations agrégées, afin de **rendre** un jugement "inter-modal".

Ce **niveau** global joue un **rôle** fondamental de **coordination** en orchestrant l'interaction entre les différents modules du **SCN**. Il fonctionne en trois étapes complémentaires.

Tout d'abord, il reçoit en **bottom-up** les informations issues de chaque module, incluant les **clusters internes**, les **liaisons fortes** et les **indices de confiance** associés aux entités détectées.

Ensuite, il procède à une **évaluation de la cohérence** entre ces modules, en vérifiant si les différentes modalités identifient un même objet ou événement à travers des canaux distincts, ce qui permet d'assurer une convergence interprétative.

Enfin, il agit en **top-down** en renvoyant un **feedback** correctif ou un **contrôle coopératif**, visant à ajuster localement les pondérations ω d'un module particulier.

Cet ajustement favorise une meilleure **harmonisation** des informations entre modalités, en maximisant la **concordance inter-modules** et en réduisant d'éventuelles contradictions ou incohérences.

On peut structurer cette **communication inter-modulaire** en deux **boucles complémentaires**, une **boucle ascendante** (*bottom-up*) et une **boucle descendante** (*top-down*), qui permettent une interaction continue entre les modules et le niveau global du **SCN**.

Dans la **boucle ascendante**, chaque module $\mathcal{M}_{\text{img}}, \mathcal{M}_{\text{aud}}, \mathcal{M}_{\text{txt}}$, ... transmet des informations au **nœud global** $\mathcal{N}_{\text{global}}$. Ces informations incluent les **liaisons internes**, les **clusters détectés** et des **scores résumés** sur la cohérence interne du module. Mathématiquement, cette étape correspond à la collecte des matrices de pondérations $\mathbf{\Omega}^{(m)}$, ou d'autres indicateurs dérivés pour chacun des m modules.

Dans la **boucle descendante**, le **niveau global** analyse ces contributions et évalue la **cohérence inter-modules** à l'aide d'une fonction Ψ. Il génère ensuite un **feedback correctif** Δ_{down} à destination de chaque module, ce qui permet d'**affiner** les liaisons $\omega_{i,j}$ en fonction de la compatibilité observée entre les canaux. Sur le plan **DSL**, cette correction se traduit par un **terme additionnel** dans la mise à jour des pondérations, donné par :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\left[S(i,j) - \tau\,\omega_{i,j}(t)\right] + \Delta_{\text{down}}^{(\text{global})}(i,j).$$

Le rôle du **terme correctif** $\Delta_{\text{down}}^{(\text{global})}(i,j)$ est de moduler dynamiquement l'évolution des liens. Un **terme positif** $\Delta_{\text{down}} > 0$ renforce la pondération $\omega_{i,j}$ lorsque plusieurs canaux confirment une association pertinente, améliorant ainsi la robustesse du SCN. À l'inverse, un **terme négatif** $\Delta_{\text{down}} < 0$ affaiblit le lien lorsqu'une incohérence est détectée, favorisant une meilleure spécialisation des clusters.

B. Coordination et Coopération entre Modules

Ce **feedback** coopératif permet aux canaux **image**, **audio**, **texte**, etc. de s'**aligner** plus efficacement.Si, à l'échelle globale, une correspondance cohérente est détectée entre plusieurs modalités, par exemple une **image identifiée comme "chien"** et un **segment audio reconnu comme "aboiement"**, le **niveau global** du SCN peut alors émettre un **signal correctif** en direction des modules concernés, renforçant ainsi leur cohésion. Ce signal incite à **augmenter la pondération des liens pertinents** pour assurer une meilleure structuration du réseau.

Dans le **module vision**, ce signal se traduit par une **hausse des pondérations internes** impliquant le concept visuel "chien", ce qui renforce la liaison $\omega_{\text{chien,…}}$ avec d'autres images de chiens similaires.

Dans le **module audio**, un effet similaire se produit, amplifiant les connexions entre le son "aboiement" et d'autres segments acoustiques apparentés, ce qui accroît la pondération $\omega_{\text{aboiement,…}}$.

Enfin, le **feedback global** agit également au niveau des **liaisons cross-modales** reliant image et audio, consolidant ainsi la relation entre la représentation visuelle et le signal sonore correspondant. Cela conduit à un **renforcement automatique** de la **synergie multimodale** entre les entités associées, stabilisant progressivement la structure auto-organisée du SCN.

À l'inverse, un **conflit** (ex. l'audio prétend "miaulement", alors que la caméra et le texte concordent "chien") peut conduire le niveau global à pénaliser la liaison $\omega_{\text{audio,chien}}$. Le sous-

module audio, recevant ce **feedback**, s'ajuste en conséquence. D'autres pondérations, comme $\omega_{\text{miaulement},\ldots}$, pourraient se voir inhibées, redirigeant ainsi l'**auto-organisation** locale.

C. Formulation Mathématique du Feedback

On peut formaliser ce **feedback** descendante comme un **terme** $\Delta^{(m)}_{\text{down}}(i,j)$ pour chaque module m. Celui-ci dépend de la **cohérence** globale repérée entre divers modules $\{m' \neq m\}$. Par exemple :

$$\Delta^{(m)}_{\text{down}}(i,j) = \alpha \sum_{m' \neq m} \text{compat}\left(\omega^{(m)}_{i,j}, \omega^{(m')}_{p,q}\right),$$

où compat évalue le degré de compatibilité entre la structure interne du module m et celle du module m'. Si la compatibilité est élevée, $\Delta^{(m)}_{\text{down}}(i,j)$ peut renforcer $\omega^{(m)}_{i,j}$. Sinon, on la réduit.

Cette mécanique rend le **SCN coopératif**, chaque module local recevant à la fois sa dynamique DSL standard et un **feedback** global, censé maintenir la cohérence inter-modules. Sur le plan **algorithmique**, on a une **boucle locale** qui correspond à la descente **DSL classique** et une **boucle globale** qui assure la réception, l'analyse et la distribution du feedback.

D. Avantages et Considérations

Sans ce **feedback** coopératif, chaque module peut demeurer cohérent en soi, mais ignorer les signaux corroborant ou réfutant ses hypothèses dans d'autres canaux. Le feedback **unifie** la perception en renforçant les clusters multi-canaux déjà plausibles et en corrigeant ou fragilisant ceux qui paraissent anormaux lorsqu'ils sont analysés dans leur globalité.

D'un point de vue **mathématique**, l'introduction de boucles de feedback rend la **dynamique** plus complexe, on ne manipule plus un simple $\omega(t)$, mais un ensemble $\{\omega^{(m)}(t)\}$ couplé par des interactions top-down. Il faut veiller à la **stabilité** et à éviter des oscillations ou divergences entre modules. Les heuristiques (recuit, inhibition, etc.) aident à canaliser ce phénomène.

8.11.3. Synthèse sur la Valeur du DSL Multimodal

La **convergence** de flux multiples (vision, langage, audio, etc.) au sein d'un **Deep Synergy Learning** (DSL) multimodal présente un **intérêt** majeur pour la **cohésion** et la **richesse** des représentations. En effet, la synergie $S(\mathcal{E}_i, \mathcal{E}_j)$, lorsqu'elle englobe des **dimensions** hétérogènes (caractéristiques visuelles, sémantiques, acoustiques), autorise une **mise en correspondance** beaucoup plus fine et adaptative entre des entités supposément "différentes" (une image, une phrase, un extrait sonore), tout en préservant la **dynamique** d'auto-organisation propre au DSL.

8.11.3.1. Le DSL multimodal offre une "cohésion" adaptative entre vision, langage, son, tirant profit des principes d'auto-organisation

Les sections précédentes (Chap. 8) ont établi que le **DSL** (Deep Synergy Learning), appliqué à un **SCN** (Synergistic Connection Network), peut prendre en charge **plusieurs modalités** (image, texte, audio, etc.) en définissant pour chaque couple (i,j) une **synergie** $S(i,j)$ reflétant la correspondance ou la complémentarité entre les entités $\mathcal{E}_i$ et $\mathcal{E}_j$. Cette **synergie**, lorsqu'elle

est jugée forte, **renforce** la pondération $\omega_{i,j}$ via la règle DSL, ce qui conduit à la formation **auto-organisée** de **clusters** multimodaux. La présente section (8.11.3.1) récapitule la façon dont le DSL multimodal parvient à établir une **cohésion adaptative** entre différentes sources (vision, langage, son) et à exploiter les principes d'**auto-organisation** en leur conférant une structure unifiée et dynamique.

A. Principe de Base

Dans un **SCN** multimodal, chaque **entité** $\mathcal{E}_i$ (image, segment textuel, extrait audio, etc.) se dote d'un vecteur (embedding) ou d'une représentation plus complexe. La **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ peut regrouper diverses composantes, chacune correspondant à une **modalité** donnée :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \alpha_{\text{vis}}\ S_{\text{vis}}(i,j) + \alpha_{\text{txt}}\ S_{\text{txt}}(i,j) + \alpha_{\text{aud}}\ S_{\text{aud}}(i,j) + \cdots$$

Si, par exemple, $\mathcal{E}_i$ est un **patch d'image** et $\mathcal{E}_j$ un **segment textuel**, la composante S_{vis} n'intervient qu'à l'intérieur de la modalité "visuel" (image–image), et la composante S_{txt} s'applique à la modalité "texte". Pour le cross-modal "image–texte", on définit une **version** $S_{\text{vis,txt}}(i,j)$, etc. Chaque **pondération** $\omega_{i,j}$ est alors mise à jour par la règle :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta\big[S(i,j) - \tau\,\omega_{i,j}(t)\big],$$

ce qui confère à la **dynamique** DSL (chap. 2.2.2) sa **flexibilité** d'extension à plusieurs canaux.

Cette **cohésion** entre modalités ne repose pas sur une procédure rigide, mais sur la **dynamique** locale d'**auto-organisation**. Dès que deux entités, comme une image représentant un "chat" et le mot "cat", se révèlent cohérentes par une similarité sémantique ou une co-occurrence avérée, leur pondération ω se **renforce**. À l'inverse, si d'autres paires, telles qu'une image de "voiture" et un audio de "miaulement", ne présentent pas de correspondance pertinente, leurs liaisons restent faibles ou s'annulent naturellement. Ainsi, le **réseau** DSL s'ajuste **en continu** à la cohérence perçue entre flux.

B. Auto-Organisation et Formation de Clusters

Au fil des itérations, des **sous-graphes** fortement connectés, appelés **clusters**, **émergent** progressivement. Ces clusters peuvent regrouper des images, des segments textuels, des morceaux audio ou d'autres entités multimodales, tous reliés par des pondérations ω élevées. Cela traduit une **unité** sémantique ou contextuelle (ex. "scène de plage", "concert de rock", "textes parlant de chat + images félines + bruits de miaulement"). Le **DSL** n'a pas besoin d'une supervision extérieure imposant ces regroupements. La **dynamique** elle-même, gouvernée par S et ω, les fait **surgir** de façon non supervisée.

Grâce à son **dynamisme**, le **DSL** multimodal demeure **robuste**. Si l'un des canaux, par exemple l'audio, est bruité ou manquant, les autres modalités, telles que le visuel et le textuel, prennent le relais et assurent la cohésion globale du cluster. On obtient souvent une **structure** plus stable que dans un cadre mono-modal, car chaque lien $\omega_{i,j}$ peut être soutenu par plusieurs composantes $\alpha_m\, S^{(m)}$ (fusion de scores). De plus, si de **nouvelles** entités surviennent (chap. 9.1), on recalcule localement les synergies, évitant un réapprentissage exhaustif de toute la base.

C. Illustration : un Score Composite

Pour illustrer, prenons la fusion tripartite **image–texte–audio**. La **synergie** $S(\mathcal{E}_i, \mathcal{E}_j)$ pourrait être :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \alpha_{\text{vis}}\ \text{vis\_score}(\mathcal{E}_i, \mathcal{E}_j) + \alpha_{\text{txt}}\ \text{txt\_score}(\mathcal{E}_i, \mathcal{E}_j) + \alpha_{\text{aud}}\ \text{aud\_score}(\mathcal{E}_i, \mathcal{E}_j),$$

où :

- vis_score compare l'**embedding** d'images (ou de vidéos),
- txt_score compare l'**embedding** textuel (similitude cosinus, par ex.),
- aud_score compare l'**embedding** sonore (spectrogramme, MFCC).

Un **couple** (i, j) pourrait n'être pertinent que dans une seule modalité ou dans plusieurs. L'**auto-organisation** DSL, identique à la formule standard, assure la convergence vers des liaisons élevées lorsque les **entités** coïncident sur un ou plusieurs canaux.

D. Portée et Intérêt

On peut élargir ce **modèle** à d'autres capteurs (LiDAR, signaux inertiels), ou à des **représentations** plus symboliques (chap. 8.7.4). Chacune génère une **composante** $S^{(m)}$ dans la synergie globale, permettant une **unification** graduelle de sources diverses. Le DSL s'avère donc apte à gérer une **multitude** de types d'entrées, pourvu que l'on définisse la **fonction** S propre à chaque canal ou paire de canaux.

De nombreux systèmes de **multimédia** (annotation, clustering, recommandation) ou de **robotique** (fusion sensorielle, perception 3D) bénéficient de la **cohésion** adaptative induite par le DSL. L'**absence** de supervision stricte ouvre des perspectives de **découverte** de correspondances, d'**auto-correction** en présence de bruit, et de **fusion** continue en cas de flux évolutifs (chap. 9).

8.11.3.2. Perspectives pour la Future IA Multimédia, la Robotique Sensorielle, etc.

En conclusion du **Chapitre 8**, il est clair que l'**approche** DSL (Deep Synergy Learning), et en particulier la mise en œuvre d'un **SCN** (Synergistic Connection Network) multimodal, ouvre des **perspectives** passionnantes pour un large éventail d'applications. Les **principes** d'**auto-organisation**, de **synergie** et de **multi-échelle** s'appliquent aussi bien à la **fusion** de données multimédia (texte, image, audio) qu'à des environnements plus "physiques" comme la **robotique sensorielle**, où de multiples capteurs (LiDAR, caméras, microphones, etc.) doivent coopérer.

A. IA Multimédia de Prochaine Génération

Les systèmes multimédia ne se contentent plus de **séparer** les canaux, tels que les images, les sons et les textes, pour les fusionner a posteriori. L'**avenir** envisage une **fusion** *évolutive* qui se construit progressivement au fil de la réception des données. Le **DSL** multimodal offre une **auto-organisation** organique où chaque entité, qu'il s'agisse d'un segment audio, d'un patch d'image ou d'une phrase textuelle, s'intègre progressivement au réseau, comme détaillé dans le chapitre 9 sur l'insertion incrémentale. La **cohérence** du système est maintenue par la règle DSL, qui renforce $\omega_{i,j}$ lorsque la synergie $S(i, j)$ est élevée, tandis que les liens faibles s'estompent naturellement au fil des itérations.

Cette **logique** évite le cloisonnement des canaux et permet de **déceler** des liens transversaux (ex. entre une vidéo donnée et un sous-titre partiellement correspondant, un timbre musical et une scène visuelle).

Au-delà de la simple correspondance surface (ex. cosinus d'embeddings), on peut imaginer des **niveaux** (micro → macro) où le SCN reconstruit des **concepts** ou **thèmes** (chap. 6). Le DSL s'enrichit ainsi de :

- **Micro-liaisons** : patch d'image ↔ token textuel (façon "vision–langage local"),
- **Macro-liaisons** : grand cluster multimodal associant plusieurs flux autour d'un thème sémantique (ex. "actualité sur le climat").

Cette capacité multi-niveau profite à la **classification** (découvrir des catégories émergentes) et à la **navigation** (indexer des contenus, extraire des motifs).

Les **applications** du DSL multimodal sont nombreuses et diversifiées. En **réalité augmentée**, un SCN multimodal permet d'ajuster dynamiquement les annotations visuelles, textuelles ou sonores en fonction de l'image capturée par la caméra, en exploitant les liaisons ω qui unissent ces différentes modalités. Les **agents conversationnels enrichis** bénéficient également de cette approche en intégrant simultanément les informations issues de la caméra, du son et du texte, ce qui améliore leur capacité à comprendre l'environnement et les gestes de l'utilisateur. Enfin, dans l'**analyse de grands corpus** multimédias, qu'il s'agisse de vidéos ou de documents, le DSL génère un **clustering auto-organisé unifié**, surpassant les méthodes traditionnelles qui compartimentent artificiellement les données en clusters distincts pour l'image, le texte ou l'audio.

B. Robotique Sensorielle et Coordination Synergique

En **robotique**, un système auto-organisé multimodal prend en charge les mesures issues de différentes sources, notamment le **LiDAR**, la **caméra** pour l'analyse visuelle, les **capteurs inertiels** et les **microphones**. Le **SCN** établit des liaisons entre les entités **LiDAR** et **visuelles** lorsque leur synergie spatiale et temporelle est confirmée. Il filtre ou ignore les associations incohérentes et ajuste progressivement la pondération des connexions ω en fonction de l'évolution du flux de données, conformément aux principes du **DSL** appliqués aux systèmes dynamiques (chap. 9).

La **dynamique** DSL confère au robot un mécanisme de **fusion** souple, sans imposer un pipeline figé (LiDAR↔camera obligatoire). Au besoin, si un capteur se dégrade, le réseau auto-organisé **bascule** plus de poids ω vers d'autres capteurs.

Plus ambitieusement, on peut coupler la **fusion** sensorielle à la **structure** motrice dans un **SCN** commun. Les nœuds représentant les **capteurs** tels que le **LiDAR**, la **caméra** ou les **microphones** s'auto-organisent en fonction de leur synergie intrinsèque. Parallèlement, les nœuds associés aux **moteurs**, qu'il s'agisse de joints ou d'actionneurs, reçoivent des retours établissant un lien direct entre la perception et l'action. L'**auto-organisation** assure ainsi un couplage **sensorimoteur flottant**, permettant à chaque entité motrice d'**apprendre** quels capteurs sont les plus pertinents pour sa commande et, inversement, aux capteurs d'adapter leur association aux actions qu'ils influencent.

On obtient ainsi une **architecture** fractale ou multi-niveau, unifiant la **perception** et l'**action** au sein d'un même **réseau** ω.

Dans le cadre de la **robotique multi-agent**, notamment pour des **flottes de drones** ou des **essaims de robots**, le **DSL multimodal** établit une répartition dynamique des **liens** $\omega_{i,j}$. Ces connexions ne se limitent pas aux capteurs d'un unique robot, mais s'étendent à **plusieurs** robots si une synergie pertinente est détectée. Lorsque des objectifs communs, des localisations

proches ou des signaux partagés renforcent ces interactions, des **macro-clusters** inter-agents émergent naturellement. Ce mécanisme favorise une **coopération distribuée**, évitant ainsi la nécessité d'une supervision centrale qui imposerait un schéma rigide d'interaction.

C. Vers une IA Synergique et Fractale

Les **scénarios** de l'IA multimédia et de la **robotique sensorielle** convergent dans la nécessité de gérer simultanément **plusieurs flux**, qu'il s'agisse d'images, de sons, de capteurs 3D ou de textes. Cette gestion implique l'**assemblage** de ces flux à **différents niveaux**, allant des micro-liaisons locales aux macro-liaisons sémantiques. L'approche repose sur un **paradigme d'auto-organisation**, évitant toute contrainte a priori sur la manière dont ces informations doivent être fusionnées, permettant ainsi une intégration dynamique et adaptative des signaux issus de diverses modalités.

Le DSL multimodal satisfait ces critères grâce à l'**universalité** de la règle DSL (pondérations ω renforcées localement, inhibition et recuit optionnels, chap. 7.3–7.4).

Dans des environnements évolutifs (nouveaux flux multimédias, nouveau capteur, nouveau robot), le **DSL** s'adapte **progressivement** (chap. 9.1). L'**inhibition** évite la prolifération indiscriminée de liens, tandis que le **recuit** (ou mini-bursts de bruit) prévient l'enlisement dans de faux minima. Cette **combinaison** d'éléments mathématiques (synergie, inhibition, recuit, partition, etc.) ouvre la voie à des **réseaux** robustes face à la quantité et la variabilité de données.