TouchSteer: Grounding Natural Language in Tactile Perception via Steering Vectors

Guanqun Cao[1], Yongji Fu, Yi Zhou, Gaojie Jin[2], Zhenyu Lu[2], Shan Luo[3]

***Abstract*— Tactile sensing provides robots with direct information about physical properties through contact, yet most existing methods describe tactile data using predefined attribute labels with limited semantic flexibility. Aligning tactile signals with human language enables richer, concept-level representations. In this work, we propose a transformer-based tactile–language framework that structures the shared embedding space as a manipulable concept space using steering vectors. These vectors encode tactile properties as semantic directions, providing explicit semantic control under limited supervision. Experimental results show that the framework effectively retrieves tactile representations from free-form natural language and generates meaningful tactile descriptions grounded in tactile perception, supporting more effective human–robot interaction.**

I. INTRODUCTION

Tactile sensing enables robots to perceive physical properties through direct contact. It provides reliable information about attributes such as softness, texture, and friction, which critically influence manipulation performance. However, characterising these physical properties in a structured and flexible manner remains a challenge. Existing approaches rely on predefined categorical labels, which fail to capture the richness and contextual nuance of human tactile descriptions.

In contrast, humans typically describe tactile experiences using open-ended language, such as *"This cloth feels soft and smooth, yet retains a certain firmness."*. Such descriptions reflect a semantically structured representation of touch, where attributes are expressed in relation to one another within a shared context rather than treated as independent classification targets. When embedded in a continuous representation space, related tactile concepts can be organised geometrically, enabling generalisation across objects with similar properties and supporting compositional reasoning over unseen attribute combinations. This allows robots to interpret novel materials through previously learned tactile concepts, instead of relying on dedicated supervision for each new object. Moreover, grounding tactile perception in natural language provides a natural interface for human-robot interaction, allowing humans to query, instruct, and communicate about tactile properties in intuitive terms.

Existing multimodal alignment approaches typically rely on similarity-based embedding spaces that bring semantically related samples closer together. However, such formulations offer limited semantic control and depend on large-scale labelled data. In tactile learning, physical interaction and careful annotation inherently limit dataset scale. Under such constraints, similarity-based methods struggle to achieve controllable cross-modal generation or strong generalisation.

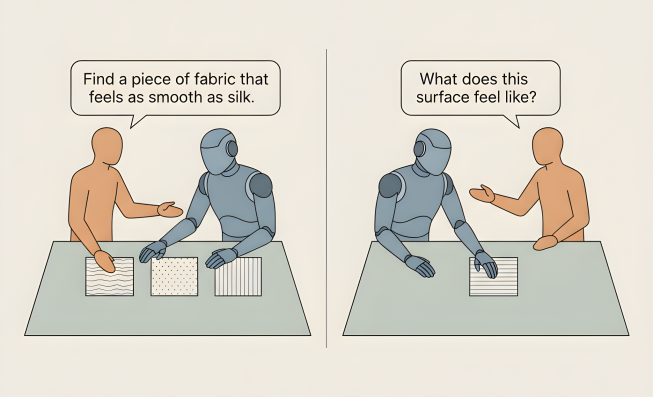


Fig. 1: **Two complementary tasks enabled by the proposed framework.** *Left:* given a free-form natural language query describing desired tactile properties, the robot retrieves the most relevant material from its tactile experience. *Right:* after contacting a surface physically, the robot generates a natural language description of tactile properties.

To address these limitations, we structure the shared tactile and language embedding space as a manipulable concept space. We introduce steering vectors that encode material properties as semantic directions, so that tactile semantics can be manipulated and composed under limited supervision within a structured, navigable concept space. Built upon this, we propose a multimodal alignment framework that retrieves tactile representations from free-form natural language and generates language descriptions grounded in perceived tactile data.

The contributions of this paper are summarised as follows:

1) We present a transformer-based framework, TouchSteer, that enables retrieval of tactile data from natural language queries and tactile-to-text generation, for the first time.
2) The steering vector is applied to enable guided cross-modal retrieval and controllable natural language generation by injecting semantic directions into the shared latent space.
3) Robot experimental results demonstrate that our retrieval and generation framework significantly improves the robot's ability to interpret natural language instructions and provide meaningful tactile feedback during human–robot interaction.

The remainder of this paper is structured as follows: Section II reviews the related works; Section III details the proposed framework for language-tactile retrieval and generation; Section IV describes the datasets and evaluation

protocol; Section V presents the experimental results; finally, Section VI summarises the paper and provides conclusions.

II. Related works

A. Tactile perception and representation

Tactile perception has been widely studied for enabling robots to recognise and classify material properties such as texture, hardness, and slipperiness [1]. Early works focused on classification using sensor arrays and handcrafted features [2]–[4]. More recently, high-resolution vision-based tactile sensors such as GelSight [5] and DIGIT [6] have provided rich contact imagery that facilitates fine-grained property estimation. Deep learning has further enabled associating visual and tactile properties [7], improving grasp prediction through multimodal sensing [8], capturing spatio-temporal patterns in contact sequences [9], and active tactile exploration for material property perception [10].

Self-supervised methods have further advanced the field. Vision-touch representations have been learned for contact-rich manipulation [11], large-scale in-the-wild data collection has broadened generalisation [12], and large-scale self-supervised pre-training has outperformed task-specific training across multiple downstream tasks [13]. Unified multimodal frameworks have also begun to align tactile representations with vision, language, and audio within a shared embedding space [14], and zero-shot material recognition has been achieved through visual-semantic transfer [15]. Despite these advances, most shared tactile or tactile–language embeddings are treated as fixed feature spaces optimised for recognition or alignment, rather than as interpretable spaces that can be adjusted along explicit tactile attributes after training. This limitation is particularly consequential given the high cost of collecting new labelled pairs for each semantic distinction. Grounding tactile representations in natural language offers a path toward more flexible and controllable embeddings.

B. Tactile–language grounding and generation

Early work on tactile–language grounding focused on learning haptic adjectives from physical interaction data, mapping sensor signals to descriptive terms such as *smooth*, *rough*, and *hard* [16]. These methods established that tactile properties can be linked to linguistic descriptors, but relied on fixed label vocabularies.

In the vision–language community, contrastive pre-training [17] and unified retrieval-generation architectures [18], [19] have driven rapid progress in cross-modal alignment. Extending these ideas to the tactile domain, recent works have introduced touch-vision-language datasets with aligned tactile encoders for multimodal representation learning [20], [21] and learned unified tactile representations across heterogeneous sensors [14]. However, the textual supervision in current tactile–language datasets is often sparse, typically consisting of short captions, adjective tags, or coarse attribute labels, which limits the semantic granularity available for alignment.

On the generation side, existing work has focused on producing tactile signals rather than language, such as vision-based haptic rendering [22], controllable visual-tactile synthesis [23], and simulated tactile images for sim-to-real transfer [24]. LLM-based tactile systems such as Octopi [25] target task-specific property reasoning rather than open-ended descriptive captioning. Existing tactile–language systems therefore do not jointly support language-to-tactile retrieval and open-ended tactile-to-language description in one framework. Moreover, even when shared embeddings are learned, they cannot be explicitly controlled at the semantic level after inference.

C. Concept directions and steering vectors

A natural mechanism for post-hoc semantic control is to represent attributes as directions in the embedding space. Testing with Concept Activation Vectors (TCAV) [26] introduced the use of directional vectors to quantify concept sensitivity in classification models. In the language modelling community, steering vectors have been used to control generation behaviour. Activation Addition [27] computes vectors from contrastive prompt pairs and adds them to hidden states at inference time, while Contrastive Activation Addition [28] extends this to scalable behavioural steering. Representation engineering [29] has further formalised the principle that high-level concepts can be identified as linear directions and manipulated accordingly. Most of these methods, however, operate either as analysis tools for concept sensitivity or as intra-modal interventions for decoder behaviour, rather than as mechanisms for steering a shared cross-modal representation.

To our knowledge, steering along semantic directions has not been studied in tactile–language alignment and has not been used to control both retrieval and description generation from the same shared embedding space. TouchSteer addresses this gap by constructing steering vectors from tactile attribute vocabularies and applying them post hoc in the shared tactile–language space, enabling attribute-level control without requiring additional labelled data.

III. Methodologies

As illustrated in Fig. 2, the proposed TouchSteer framework comprises three components: (i) a textual description construction and refinement pipeline that produces tactile attribute descriptions from paired RGB and tactile images (Section III-A); (ii) a steering vector mechanism guides model embeddings toward tactile semantics using the attribute lexicons (Section III-B); and (iii) a unified model with a shared tactile backbone that jointly optimises language-tactile retrieval via contrastive alignment (Section III-C) and tactile description generation via autoregressive decoding (Section III-D).

A. Textual Description Construction and Refinement

High-quality textual descriptions are essential for effective cross-modal alignment, yet producing accurate tactile descriptions is inherently challenging. Unlike visual annotation,

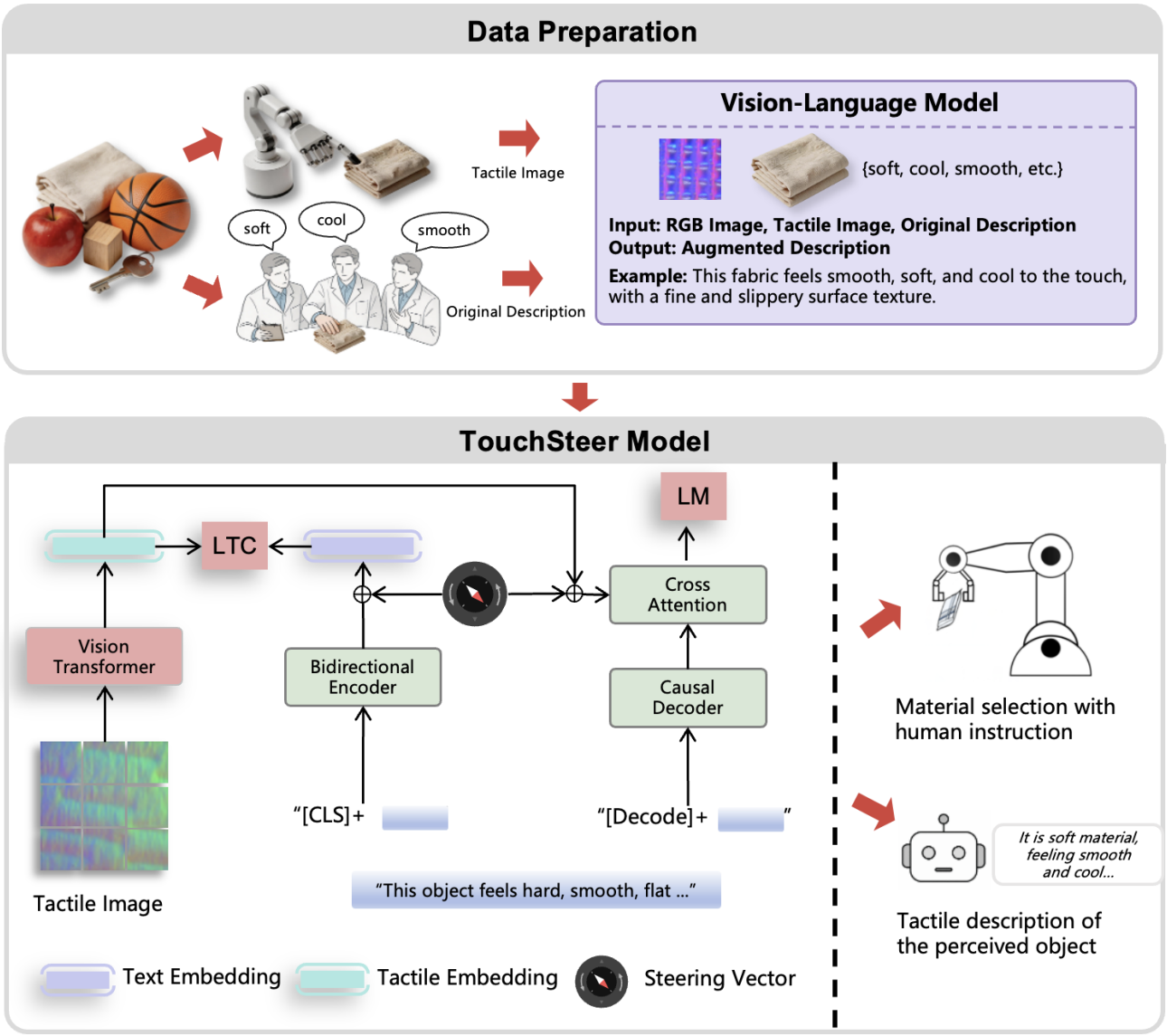


Fig. 2: **Overview of the proposed GRASP framework.** *Top:* Data collection and description refinement pipeline. A robotic arm equipped with a tactile sensor acquires tactile images under varying contact conditions (angle, force, position), while human annotators independently provide original tactile descriptions of the same materials. The RGB image, tactile image, and original description are jointly fed to a Vision–Language Model (VLM) to produce structured present and absent attribute sets. Descriptions with low discriminativeness, identified via human evaluation (Appendix F), are re-annotated by a panel of five experts and re-processed through the VLM in an iterative refinement loop. *Bottom:* Steering vector model. A shared ViT backbone encodes tactile images for two jointly trained tasks: contrastive retrieval via the LTC head and autoregressive description generation via the LM decoder. Steering vectors, constructed from a positive prompt $\mathcal{A}$ and a negative prompt $\mathcal{N}$, can be injected at four positions: the retrieval text encoder (P1), retrieval vision encoder (P2), generation decoder token embeddings (P3), and generation cross-attention visual features (P4).

tactile annotation requires direct physical interaction with the material and careful use of tactile vocabularies to describe properties such as softness, roughness, stiffness, and fine-grained surface texture. This process is labour-intensive and difficult to scale, which explains why existing tactile datasets typically provide only short keyword captions (e.g., "soft, smooth, flat") that lack sufficient semantic richness for fine-grained retrieval. To address this limitation, we propose an iterative refinement pipeline that leverages vision–language models (VLMs) and large language models (LLMs) to construct structured attribute descriptions.

Each tactile sample is associated with an RGB image, a tactile image, an object name, and an original tactile description. Because the original description in most datasets is typically short and semantically sparse, we further enrich it using a two-level attribute representation. Specifically, we treat all tactile samples belonging to the same object as a set and construct a two-level attribute representation for them.

The first level consists of *anchor attributes*, which represent the shared and relatively stable tactile characteristics of the same object across different contact samples. The second level consists of *additional attributes*, which capture the local tactile characteristics specific to an individual contact sample. The former reflects semantic consistency at the object level, while the latter captures sample-level differences caused by contact position, contact angle, contact force, or local surface texture.

In practice, for each object o, we collect all associated samples into a set $\mathcal{S}_o = \{(I_i^{rgb}, I_i, d_i)\}_{i=1}^{N_o}$, where I_i^{rgb} denotes the RGB image, I_i denotes the tactile image, and d_i denotes the original tactile description of the i-th sample, respectively. We first aggregate the original descriptions of all samples belonging to the same object, and combine them with the corresponding RGB observations and object name n_o to generate object-level anchor attributes: $a_o = VLM_{\text{anchor}}\left(n_o, \{I_i^{rgb}\}_{i=1}^{N_o},\right)$. Then, for each individual sample i, we generate sample-specific additional attributes as $b_i = VLM_{\text{add}}\left(I_i^{rgb}, I_i, n_o, d_i, a_o\right)$. In this way, a_o captures semantic consistency across samples of the same object, while b_i enriches each sample with instance-level tactile variation. The final structured tactile description for sample i is written as $z_i = \{a_o, b_i\}$.

B. Steering Vector Construction

The key mechanism of TouchSteer is a *steering vector* that encodes the direction from non-tactile to tactile concepts in embedding space. For any token w, let $\mathbf{e}(w)$ denote its embedding from the text encoder. We define a *positive lexicon* $\mathcal{A}$ containing words that describe material properties perceivable through touch (e.g., {`soft`, `smooth`, `rough`, `coarse`, `fluffy`}), and a *negative lexicon* $\mathcal{N}$ containing words associated with visual or categorical properties not perceivable through touch (e.g., {`red`, `blue`, `square`, `transparent`}). Both lexicons are fixed, global sets curated independently of the per-sample attributes; the full lists are provided in Appendix G. The steering vector is computed as:

$$\mathbf{s} = \frac{1}{|\mathcal{A}|}\sum_{w\in\mathcal{A}} \mathbf{e}(w) - \frac{1}{|\mathcal{N}|}\sum_{w\in\mathcal{N}} \mathbf{e}(w), \quad \hat{\mathbf{s}} = \frac{\mathbf{s}}{\|\mathbf{s}\|_2}.$$

The vector $\hat{\mathbf{s}} \in \mathbb{R}^d$ is computed once before training and kept fixed throughout. It is injected into both the retrieval and generation branches, as described in the following subsections.

C. Tactile-Language Contrastive Alignment

As shown in the bottom half of Fig. 2, a shared Vision Transformer (ViT) backbone encodes the tactile images for both tasks. Given a tactile image I, the ViT produces a tactile embedding $\mathbf{f}(I) \in \mathbb{R}^d$, which is projected into a shared latent space:

$$\mathbf{v} = \text{normalize}\big(\mathbf{W}_{\text{img}}\,\mathbf{f}(I)\big) \in \mathbb{R}^{d'},$$

where $\mathbf{W}_{\text{img}} \in \mathbb{R}^{d'\times d}$ is a trainable projection matrix and $\text{normalize}(\cdot)$ denotes ℓ_2 normalisation. On the text side, a Transformer text encoder maps an input descriptor T to a pooled representation $\mathbf{g}(T)$, which is projected into the same space via $\mathbf{W}_{\text{text}} \in \mathbb{R}^{d'\times d}$. The steering vector is then projected and added to the text embedding to bias it toward tactile semantics:

$$\hat{\mathbf{s}}_{\text{proj}} = \frac{\mathbf{W}_{\text{text}}\hat{\mathbf{s}}}{\|\mathbf{W}_{\text{text}}\hat{\mathbf{s}}\|_2}, \quad \tilde{\mathbf{t}} = \text{normalize}(\mathbf{W}_{\text{text}}\,\mathbf{g}(T) + \alpha_r\,\hat{\mathbf{s}}_{\text{proj}}),$$

where $\alpha_r \geq 0$ controls the steering strength. This injection is active during both training and inference.

Particularly, we align the steered text and tactile image embeddings via a symmetric contrastive objective based on InfoNCE. For a minibatch of image–text pairs $\{(I_i, T_i)\}_{i=1}^N$, the temperature-scaled cosine similarity between tactile image I_i and text T_j is:

$$s_{ij} = \frac{\mathbf{v}_i^\top \tilde{\mathbf{t}}_j}{\tau},$$

with learnable temperature $\tau > 0$. The contrastive loss optimises both matching directions:

$$\mathcal{L}_{\text{i2t}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(s_{ii})}{\sum_{j=1}^{N}\exp(s_{ij})}, \tag{1}$$

$$\mathcal{L}_{\text{t2i}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(s_{ii})}{\sum_{j=1}^{N}\exp(s_{ji})}, \tag{2}$$

$$\mathcal{L}_{\text{LTC}} = \tfrac{1}{2}\big(\mathcal{L}_{\text{i2t}} + \mathcal{L}_{\text{t2i}}\big).$$

D. Generation of Tactile Descriptions

The generation module employs a Transformer decoder with causal self-attention and cross-attention. The cross-attention layers condition the output on the visual embedding $\mathbf{f}(I)$ from the shared ViT backbone. To bias the decoder toward tactile vocabulary, the steering vector is added to the visual embedding before it enters the cross-attention layers:

$$\tilde{\mathbf{f}}(I) \;=\; \mathbf{f}(I) \;+\; \alpha_g\,\hat{\mathbf{s}},$$

where $\alpha_g \geq 0$ controls the steering strength. Unlike retrieval steering, which operates in the projected space $\mathbb{R}^{d'}$, generation steering operates directly in the ViT embedding space $\mathbb{R}^d$. This injection is also active during both training and inference. Alternative injection positions are explored in the ablation study (Section V-D).

The decoder is supervised with an autoregressive language modelling objective. Given image–text pairs (I_i, T_i) where $T_i = (w_1^{(i)}, \ldots, w_{L_i}^{(i)})$, the loss is

$$\mathcal{L}_{\text{LM}} = -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{L_i}\sum_{t=1}^{L_i}\log P\Big(w_t^{(i)} \mid w_{<t}^{(i)},\, \tilde{\mathbf{f}}(I_i)\Big),$$

where $w_{<t}^{(i)} = (w_1^{(i)}, \ldots, w_{t-1}^{(i)})$ denotes all previously generated tokens and $P(w_t^{(i)} \mid w_{<t}^{(i)}, \tilde{\mathbf{f}}(I_i))$ is the decoder's predicted distribution over the vocabulary, conditioned on the steered visual features.

TABLE I: **Summary of the three tactile evaluation datasets.** RAS comprises high-resolution GelSight images of fabric materials with human-annotated binary attributes. HCT contains in-the-wild DIGIT sensor data of everyday objects with multiple samples per object. SSVTP provides single-sample DIGIT captures of material textures. The datasets span different sensing modalities, object diversity, and evaluation scales.

	FabricVST	**HCT**	**SSVTP**
Domain	Fabric texture	Everyday objects	Material texture
Sensor	GelSight	DIGIT	DIGIT
Total samples	16,250	38,960	4,587
Total classes	50	847	4,587
Text source	Expert-voted attributes	Structured	Structured

E. Joint Training Objective

The model is trained end-to-end by minimising the combined loss:

$$\mathcal{L} = \mathcal{L}_{\text{LTC}} + \mathcal{L}_{\text{LM}}.$$

Both objectives share the same ViT backbone. The contrastive loss encourages the encoder to organise tactile samples in a semantically structured embedding space, bringing matched tactile–text pairs closer while separating mismatched ones. The generation loss, in turn, encourages the encoder to preserve fine-grained surface information needed for attribute-level description. Joint optimisation of the two objectives provides complementary supervision at different levels of representation, encouraging the encoder to capture both global semantic structure and fine-grained descriptive cues.

IV. Experimental Datasets

We evaluate TouchSteer on three tactile datasets that cover different sensing modalities, object categories, and evaluation settings. Table I summarises the key statistics.

A. FabricVST

The FabricVST dataset [15] comprises 50 fabric materials collected using a GelSight tactile sensor mounted on a UR5 arm. The GelSight sensor captures high-resolution (640×480) RGB images of surface deformation through an elastomer membrane with a 1.5×1.1 cm perception field. For each material, the robot presses the sensor at 225 uniformly spaced contact points within an 8×8 cm area with a controlled force of 15 N. Each material is annotated with 24 binary tactile attributes (e.g., *soft*, *rough*, *thick*, etc.) through a consensus process in which five trained expert raters independently assess each attribute and the final label is determined by majority voting. The dataset is partitioned into training and testing subsets using a 7:3 ratio.

B. HCT Dataset

The HCT (Human Collected Tactile) dataset, drawn from the Touch-Vision-Language (TVL) benchmark [20], comprises in-the-wild visual–tactile data of 1,492 everyday objects. The data were collected by five human operators using a handheld device equipped with a DIGIT tactile sensor, which synchronously records visual and tactile observations to ensure cross-modal alignment. Each object is represented by multiple tactile images captured under varying contact conditions, with an average of approximately 45 images per object. The dataset provides a predefined train/test partition, comprising 38,604 training samples across 847 objects and 356 test samples across 279 unique objects. The multi-sample-per-object structure enables both object-level and sample-level retrieval evaluation.

C. SSVTP Dataset

The SSVTP (Self-Supervised Visuo-Tactile Pretraining) dataset [30], targets material texture classification. It contains 4,587 visual–tactile pairs collected by a UR5 robot equipped with a DIGIT sensor in a laboratory environment. The robot first captures a top-down image of objects on a work surface, then presses the sensor onto the corresponding location. Each image corresponds to a distinct material sample. The dataset provides a predefined split of 4,541 training and 46 test samples. Unlike HCT, each sample in SSVTP represents a unique material with no repeated object instances.

V. Experimental Results and Analysis

We evaluate TouchSteer on three tactile datasets spanning different sensors, object categories, and annotation styles. The experiments are organised around four questions: (1) Does TouchSteer outperform existing tactile–language models on cross-modal retrieval? (2) Can it generate accurate tactile descriptions from contact data? (3) How do steering injection position and strength affect performance? (4) What is the role of the text representation in retrieval and generation task?

A. Experimental Setup

Model architecture. The architecture of TouchSteer includes a ViT-B/16 vision encoder with a 12-layer Transformer text encoder for retrieval and a 12-layer Transformer decoder for generation.

Steering configuration. We inject the steering vector at two positions simultaneously, namely P1 (retrieval text encoder) with α=0.02 and P4 (generation cross-attention) with α=0.5, using the $\mathcal{A}-\mathcal{N}$ prompt design described in Section III. The two strengths reflect the different dimensionalities of the injection spaces. P1 operates in the projected retrieval space $\mathbb{R}^{d'}$, where small perturbations suffice, while P4 operates in the higher-dimensional raw embedding space $\mathbb{R}^{d}$, requiring a larger magnitude. We also explored the steering vector with different locations and strengths. These values are determined via ablation as detailed in Section V-D.

Evaluation metrics. For retrieval, we report Recall@K (R@K) for $K \in \{1, 5\}$, averaged over 3 evaluation runs. For generation, we report BLEU-1 to measure unigram overlap between generated and reference tactile descriptions. We do not report higher-order n-gram metrics (BLEU-4) or consensus-based metrics (CIDEr), as tactile descriptions primarily consist of unordered sets of adjectives (e.g., "soft, smooth, rough") where word order carries little semantic

TABLE II: **Comparison with baseline methods for language-tactile retrieval across three datasets (%).** Our proposed method TouchSteer achieves the best performance across all three datasets. Best results per column are in **bold**. ZS: zero-shot; FT: fine-tuned.

	FabricVST		HCT		SSVTP	
Method	R@1	R@5	R@1	R@5	R@1	R@5
UniTouch, ZS	2.0	4.0	0.28	1.12	4.3	6.5
TVL, ZS	2.0	8.0	2.53	7.87	2.2	10.9
UniTouch, FT	52.0	64.0	25.28	53.09	21.7	63.0
TVL, FT	44.0	58.0	39.61	76.40	8.7	43.5
CLIP	87.0	89.0	42.65	81.36	31.8	68.7
TouchSteer (Ours)	**96.0**	**96.0**	**50.28**	**88.76**	**37.0**	**73.9**

TABLE III: **Tactile description generation quality measured by BLEU-1 across all datasets.** TouchSteer is compared against three generative baselines, namely GIT-Base, ViT-GPT2, and CoCa ViT-B-32. The HCT column reports Mode 2 text descriptions. Higher BLEU-1 indicates stronger unigram overlap with the reference tactile descriptions.

	FabricVST	**HCT**	**SSVTP**
GIT-Base	67.32	46.88	37.86
ViT-GPT2	63.53	43.12	36.13
CoCa ViT-B-32	62.11	58.40	49.37
TouchSteer (Ours)	73.6	62.3	54.7

significance, making order-sensitive metrics unreliable indicators of description quality.

B. Tactile retrieval using natural language description

We compare against retrieval and generation baselines separately. For retrieval, we include (i) **CLIP** [17], fine-tuned with symmetric contrastive loss; (ii) **UniTouch** [14], an ImageBind-based tactile foundation model, evaluated in both zero-shot and fine-tuned settings; (iii) **TVL** [20], a ViT-Small tactile encoder aligned to OpenCLIP ViT-L-14, also evaluated zero-shot and fine-tuned. We do not report CLIP in a zero-shot setting because it is pretrained on image–text pairs rather than tactile data, making direct zero-shot evaluation under this modality mismatch less informative.

Table II presents the retrieval performance of TouchSteer and three baselines across different tactile datasets, evaluated on the test set of each dataset. In the zero-shot setting, both UniTouch and TVL perform close to chance level across all datasets, indicating that representations learned from their pretraining data do not generalise well to free-form tactile–language retrieval without task-specific adaptation.

Fine-tuning yields substantial improvements for both models. The relative ranking between fine-tuned UniTouch and fine-tuned TVL is dataset-dependent. TVL benefits more from fine-tuning on HCT, where its ViT-Small architecture generalises well to diverse everyday objects, whereas UniTouch shows stronger performance on SSVTP, suggesting that its ImageBind-based representation captures material texture properties more effectively.

TouchSteer achieves the best performance on all three datasets, consistently outperforming both fine-tuned foundation models and the CLIP baseline. The gains are especially notable at R@1, showing that TouchSteer improves not only coarse retrieval but also top-ranked matching accuracy. This suggests that the proposed steering mechanism provides a more effective way to align tactile and language representations at the attribute level, allowing the model to better capture the semantic distinctions required for free-form tactile–language retrieval across diverse domains.

C. Tactile Description Generation

A distinctive advantage of TouchSteer over retrieval-only baselines is its ability to generate natural-language descriptions from tactile images through textdecoder. We evaluate generation quality using BLEU-1, which measures whether the model produces the correct tactile description. We deliberately omit higher-order n-gram metrics such as BLEU-4 and CIDEr, because tactile descriptions are inherently *bag-of-adjectives* where a surface described as "soft, smooth, slightly rough" is equally valid as "smooth, slightly rough, soft." Order-sensitive metrics penalise such semantically equivalent permutations and therefore do not reliably reflect generation quality in this domain.

Table VIII reports the BLEU-1 scores for tactile description generation on FabricVST, HCT, and SSVTP. TouchSteer achieves the best performance on all three datasets, with scores of 73.6 on FabricVST, 62.3 on HCT, and 54.7 on SSVTP. These differences likely reflect not only variations in description format, but also differences in annotation granularity and lexical diversity across datasets. Despite these variations, TouchSteer maintains a clear advantage over all baselines in every setting. Overall, the results demonstrate that TouchSteer generalises effectively across datasets with different description styles and provides the strongest overall generation performance among the compared methods.

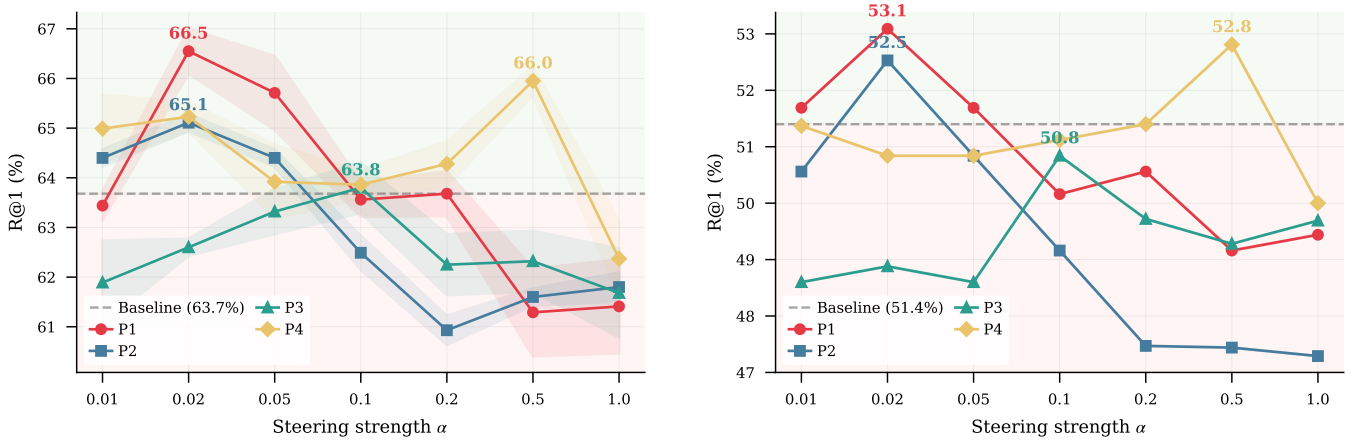
D. Steering Vector Injection Ablation

In this subsection, we evaluate how the steering strength α and the injection position affect retrieval and generation performance. The HCT dataset is selected for this ablation study. As shown in Fig. 4, the steering vector is explored to be injected at four positions within the model, each corresponding to a different embedding pathway:

- **P1** applies steering to the projected text embedding $\mathbf{t}$ in the shared retrieval space $\mathbb{R}^{d'}$.
- **P2** applies steering to the projected tactile image embedding $\mathbf{v}$ in the shared retrieval space $\mathbb{R}^{d'}$.
- **P3** applies steering to the token embeddings input to the autoregressive decoder in $\mathbb{R}^{d}$.
- **P4** applies steering to the visual features fed into the decoder cross-attention layers in $\mathbb{R}^{d}$.

P1 and P2 are applied in the projected latent space, where they directly affect contrastive alignment, whereas P3 and P4 are applied in the decoder embedding space, where they primarily influence generation behaviour.

Particularly, the multi-sample-per-object structure of the dataset enables retrieval evaluation at two levels of granularity, including *object-level retrieval* and *sample-level retrieval*.



(a) Object level retrieval

(b) Instance level retrieval

Fig. 3: **Effect of steering injection position and strength on R@1.** Four injection positions are evaluated: P1 (text encoder), P2 (tactile encoder), P3 (text decoder token embeddings), and P4 (cross-attention tactile features), across different steering strengths $\alpha \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1.0\}$.

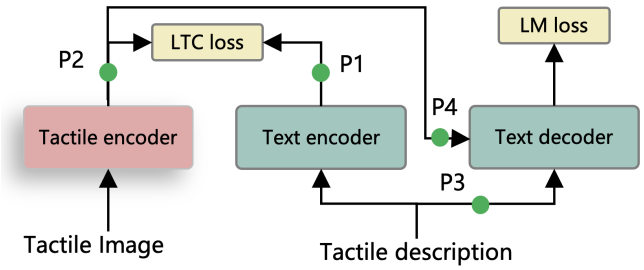


Fig. 4: **Position of steering vectors.**The green dots indicate the candidates of injection positions for steering vectors.

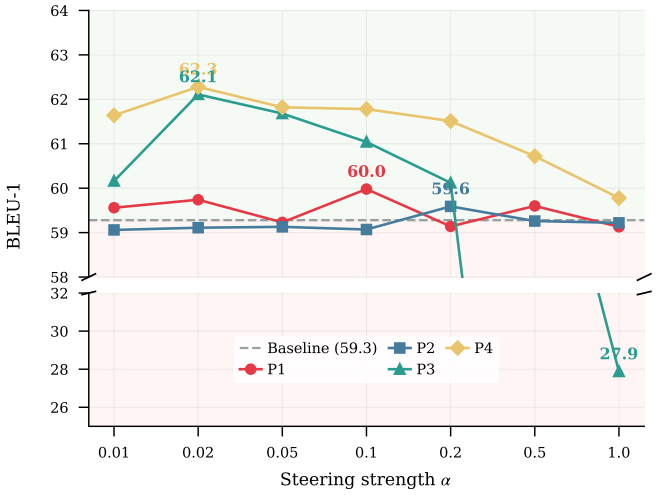


Fig. 5: **Effect of steering injection position and strength on generation quality.** BLEU-1 is plotted for each injection position across different steering strengths. The unsteered baseline is denoted by the grey dashed line. Peak values are annotated for each position.

In the object-level setting, a retrieval is considered correct if the returned sample belongs to the correct object, even if it is not the exact paired sample. In the sample-level setting, each test sample is used as a query, and the model must retrieve the exact matching sample rather than any sample belonging to the same object.

Figure 3 shows that retrieval performance is most sensitive to steering applied in the shared retrieval space. Among all positions, P1 gives the strongest and most reliable gains, achieving the best R@1 at α=0.02 in both settings. This suggests that steering the text embedding directly is the most effective way to improve tactile–language alignment. Although both P1 and P2 yield improvements, we use only P1 as the default setting. Since both positions act in the same projected retrieval space, their effects are likely to be partially redundant, and joint injection may complicate the intervention. Given that P1 provides the strongest and most stable improvement, it is the most suitable default choice.

A second clear trend is that the effect of steering is highly strength-dependent. For both P1 and P2, performance improves at small values of α and then declines as α increases, producing a clear non-monotonic pattern. This indicates that moderate steering helps bias the representation toward more retrieval-relevant semantics, whereas overly strong perturbations begin to disrupt the structure of the shared embedding space. The fact that P1 and P2 peak at the same value, α=0.02, is also consistent with both operating in the same projected retrieval space $\mathbb{R}^{d'}$.

By contrast, P3 provides little consistent improvement, which is unsurprising given that decoder token embeddings are not directly involved in contrastive retrieval. P4 shows a different pattern, where its best result is reached only at a much larger steering strength of 0.5. This suggests that steering the cross-attention tactile features influences retrieval more indirectly than steering the retrieval embeddings. Overall, these results support using P1 with α=0.02 as the default retrieval configuration, while P4 is better regarded as a complementary position when generation is also considered.

TABLE IV: **Retrieval and generation results using different text representation variants on HCT (%).** Mode 3 gives the best retrieval and generation performance under both evaluation settings. Best results are in **bold**.

Text mode	Object-level		Instance-level		Generation
	R@1 ↑	R@5 ↑	R@1 ↑	R@5 ↑	BLEU-1 ↑
Mode 1	4.06	11.23	3.65	10.39	19.25
Mode 2	62.01	91.64	50.28	88.76	62.30
Mode 3	**79.21**	**99.28**	**62.92**	**96.35**	**67.92**
Mode 4	53.29	83.39	41.85	79.21	50.11

Figure 5 evaluates the effect of steering on tactile description generation using BLEU-1 on the HCT dataset. P1 and P2 preserve generation quality across the full range of tested strengths, with BLEU-1 remaining within ± 1 point of the unsteered baseline of 59.3. This indicates that steering in the projected retrieval space does not noticeably interfere with the text decoder. By contrast, P3 is much more sensitive to steering strength. It performs well at small values of α, but degrades sharply once the perturbation becomes stronger, with BLEU-1 dropping to 44.8 at α=0.5 and 27.9 at α=1.0. This suggests that directly perturbing decoder token embeddings can destabilise generation. P4 shows a milder pattern, remaining competitive at low and moderate strengths and declining more gradually as α increases. Although P4 achieves its highest BLEU-1 at a smaller steering strength when applied alone, we use α=0.5 in the final configuration because it provides a better trade-off in combination with P1 when jointly considering retrieval and generation.

E. Tactile Description Variants

Following the tactile description refinement pipeline described in Section III-A, we use Qwen-VL [31] to generate tactile descriptions for training. The original annotations are short keyword-style captions (e.g., "soft, smooth, flat"), which provide limited linguistic diversity and weak semantic structure. To study the effect of textual formulation on tactile–language alignment, we design four text description modes for tactile features:

- **Mode 1**: the original keyword caption only.
- **Mode 2**: a flexible natural-language description including both anchor and dynamic attributes.
- **Mode 3**: Mode 2 with the object or material name added as a prefix.
- **Mode 4**: Mode 1 with the object or material name added as a prefix.

During training, we apply text augmentation to improve lexical diversity and reduce overfitting to fixed phrasings. Specifically, we use keyword shuffling, synonym replacement with probability $p = 0.3$.

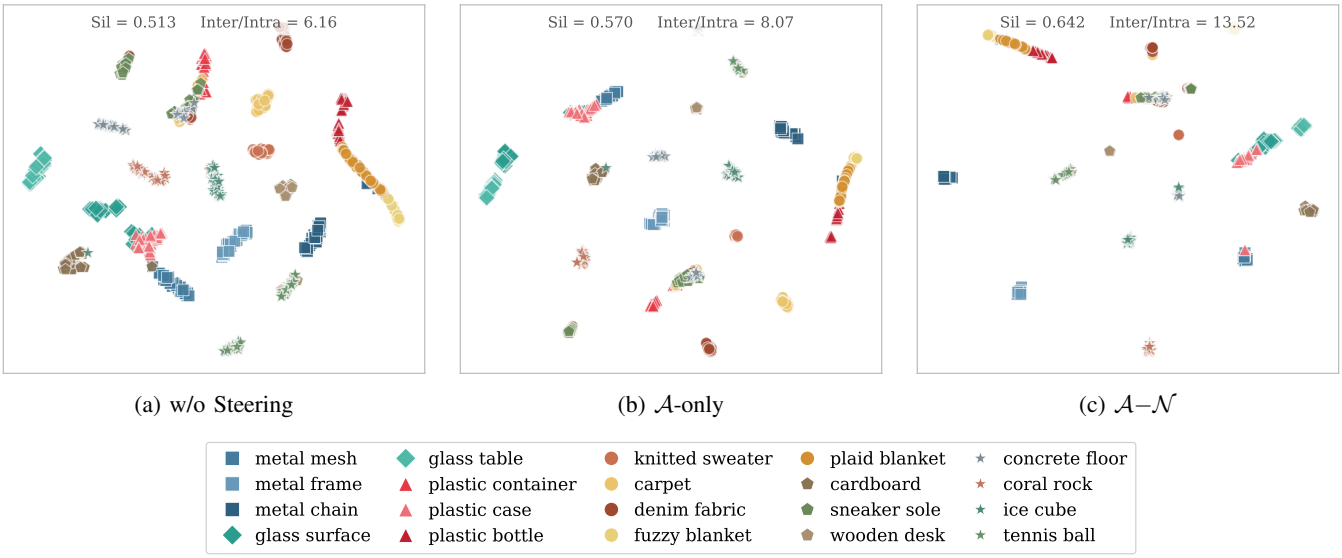
Table IV shows that retrieval performance is sensitive to text representation. Mode 1, which uses only keyword captions, performs poorly in both object-level and instance-level evaluation, suggesting that sparse annotations are insufficient for robust tactile–language alignment. In contrast, Mode 2 substantially improves all metrics, showing that structured natural-language descriptions provide much stronger supervision.

Mode 3 performs best across all settings, achieving 79.21%/99.28% R@1/R@5 at the object level and 62.92%/96.35% at the instance level. This suggests that adding the object or material name to a structured description provides complementary information beyond attributes alone. Mode 4 also outperforms Mode 1, but remains clearly below Modes 2 and 3, indicating that object identity is helpful but cannot substitute for richer textual structure. Nevertheless, Mode 3 is best regarded as an optional enhancement, since object or material names are not always available. For datasets without such labels, or for material selection scenarios where object identity is unknown, Mode 2 provides a more practical default. Moreover, the result is different from the cross-dataset observation that more diverse tactile descriptions generally make retrieval harder. The difference is that Table IV compares supervision quality within a fixed dataset rather than overall difficulty across datasets. Although richer descriptions increase linguistic complexity, they can still improve performance when they provide clearer and more informative alignment signals.

F. Embedding Space Visualisation

To qualitatively assess how steering vectors reshape the cross-modal embedding space, we visualise image–text similarity distributions using t-SNE, as shown in Figure 6. We select 20 representative object categories spanning metal, fabric, glass, plastic, and wood, sampling $\sim$50 tactile images per category for a total of 978 images. For each image, we compute its cosine similarity with all 20 class-level text descriptions and obtain a 20-dimensional cross-modal similarity vector. This vector captures the model's confidence distribution over all categories. A well-steered model concentrates probability mass on the correct class, producing more compact and separated clusters.

We compare three configurations: (a) no steering as the baseline, (b) $\mathcal{A}$-only steering using only the positive prompt, and (c) $\mathcal{A}-\mathcal{N}$ steering based on the difference between positive and negative prompt embeddings. As shown in Figure 6a, without steering, categories with similar tactile properties, such as glass and plastic surfaces, produce overlapping clusters, indicating that the model conflates fine-grained tactile semantics. $\mathcal{A}$-only steering, shown in Figure 6b, improves cluster separation, with the Silhouette score (higher values indicate better intra-class compactness and inter-class separation) increasing from 0.513 to 0.570. It demonstrates that anchoring the embedding toward tactile attributes already provides a useful inductive bias. Moreover, full $\mathcal{A}-\mathcal{N}$ steering, shown in Figure 6c, yields the most compact and well-separated clusters, with the Silhouette score further improving to 0.642 and the inter-class to intra-class distance ratio increasing from 6.16 to 13.52. This shows that subtracting non-tactile attribute embeddings sharpens decision boundaries by suppressing irrelevant dimensions, such as colour and shape. Table V shows the same trend quantitatively. Under identical training conditions, $\mathcal{A}-\mathcal{N}$



(a) w/o Steering (b) $\mathcal{A}$-only (c) $\mathcal{A}-\mathcal{N}$

Fig. 6: **t-SNE visualisation of cross-modal similarity vectors for 20 HCT object categories spanning metal, fabric, glass, plastic, and wood.** Marker shapes encode material categories (see legend). Each point represents a 20-dimensional similarity distribution of a tactile image over all category-level text descriptions. (a) Without steering, categories with similar tactile properties overlap substantially, with Silhouette = 0.513 and Inter/Intra = 6.16. (b) With $\mathcal{A}$-only steering using the positive prompt only, clusters show modest improvement, achieving Silhouette = 0.570 and Inter/Intra = 8.07. (c) With $\mathcal{A}-\mathcal{N}$ steering, i.e., positive minus negative prompt, clusters become markedly more compact and well separated, with Silhouette = 0.642 and Inter/Intra = 13.52.

TABLE V: **Effect of steering vector design on retrieval and generation (%).** Three configurations are compared: no steering, positive-prompt-only ($\mathcal{A}$-only), and positive-minus-negative ($\mathcal{A}-\mathcal{N}$). $\mathcal{A}-\mathcal{N}$ yields the highest R@1 on both retrieval and generation task. Best results per column are in **bold**.

Design	Object Level R@1	Instance Level R@1	BLEU-1
w/o Steering	63.7	51.4	59.3
$\mathcal{A}$-only	63.4	51.1	**62.3**
$\mathcal{A}-\mathcal{N}$	**66.5**	**52.8**	**62.3**

TABLE VI: **Tactile retrieval accuracy with robots (%).** The evaluation contains 90 prompts, including 55 2-way queries and 35 3-way queries. In each run, each candidate material is represented by one of its first three tactile observations, and the reported results are averaged over three runs. Higher is better.

Method	Overall	2-way	3-way
TouchSteer (Ours)	**65.56**	**70.91**	57.14
CLIP, FT	52.22	60.00	40.00
UniTouch, ZS	12.22	14.55	8.57
UniTouch, FT	54.44	56.36	51.43
TVL, ZS	8.89	10.91	5.71
TVL, FT	58.89	54.55	**65.71**

achieves the best retrieval performance, improving per-object R@1 from 63.7% to 66.5%, whereas $\mathcal{A}$-only steering remains close to the unsteered baseline. For generation, steered variants improve BLEU-1 from 59.3% to 62.3%.

G. *Material selection and tactile description with robots*

To further assess whether the learned tactile–language alignment is useful in an interactive robot setting, we built a compact real-robot-style evaluation of different kinds of fabrics. The evaluation contains 90 prompts, including 55 2-way selection tasks and 35 3-way selection tasks. The prompts are written in application-oriented language, such as *choosing fabrics for baby clothing, winter garments, or breathable daily wear, rather than using isolated attribute words alone.* This setup better reflects how a human would issue requests to a robot assistant during material selection.

For each query group, we repeat the retrieval three times and report the average accuracy. In each run, each candidate material is represented by one of its first three tactile observations. This procedure reduces the sensitivity of the evaluation to any single contact. The candidate pool is restricted to two or three materials per query, following the intended robot interaction protocol in which the robot helps a user choose from a small set of available fabrics. We evaluate the same curated prompt set for all methods and report accuracy for the overall set, the 2-way subset, and the 3-way subset separately.

Table VI shows that TouchSteer achieves the best overall accuracy on this interactive selection benchmark, outperforming the strongest baseline by 6.67 percentage points overall. The advantage is most evident in the 2-way setting, where TouchSteer reaches 70.91%, indicating that steering the shared embedding space helps the model respond to preference-style natural language queries that implicitly describe tactile requirements. In the 3-way setting, TVL, FT achieves the highest accuracy, suggesting that multi-choice

ranking with an additional distractor remains challenging and that different backbones may favour different candidate-pool structures. Even so, TouchSteer delivers the strongest overall trade-off across both difficulty levels, which supports its use as the default model in our robot-facing retrieval pipeline.

VI. Conclusion

We presented TouchSteer, a unified tactile–language framework that jointly supports cross-modal retrieval and natural-language description generation from tactile images. The key contribution is a steering vector mechanism that biases shared vision–language embeddings toward tactile semantics by injecting directional signals derived from contrastive tactile and non-tactile vocabulary prompts. Steering is applied at the retrieval text encoder (P1) with a small strength (α=0.02) to sharpen cross-modal alignment, and optionally at the generation cross-attention pathway (P4) with a larger strength (α=0.5) to enhance tactile vocabulary in generated descriptions.

Experiments on three tactile datasets (RAS, HCT, and SSVTP) spanning two sensor modalities demonstrate that TouchSteer outperforms four baselines, including fine-tuned CLIP and two tactile foundation models, on in-distribution retrieval across all datasets, while also achieving the best out-of-distribution generalisation on unseen materials. Ablation studies confirm an inverted-U relationship between steering strength and performance, with moderate steering providing beneficial inductive bias and excessive steering degrading accuracy. The $\mathcal{A}-\mathcal{N}$ (positive-minus-negative) steering design consistently outperforms positive-only steering by suppressing non-tactile confounders in the embedding space.

The framework has several limitations that suggest future directions: the global steering vector treats all materials uniformly, and adaptive, category-specific steering may further improve performance; the current evaluation is restricted to two sensor types, and broader sensor coverage would strengthen the generalisability claims; and the generation evaluation relies on unigram metrics, which do not fully capture semantic correctness. Extending the steering mechanism to larger vision–language backbones (e.g., BLIP-2) and incorporating multi-touch or active exploration strategies to address sensor-level ambiguities are promising avenues for future work.

References

[1] S. Luo, J. Bimbo, R. Dahiya, and H. Liu, "Robotic tactile perception of object properties: A review," *Mechatronics*, vol. 48, pp. 54–67, 2017.

[2] S. Luo, W. Mou, K. Althoefer, and H. Liu, "Novel tactile-sift descriptor for object shape recognition," *IEEE Sensors Journal*, vol. 15, no. 9, pp. 5001–5009, 2015.

[3] S. Luo, W. Mou, K. Althoefer, and H. Liu, "iclap: Shape recognition by combining proprioception and touch sensing," *Autonomous Robots*, vol. 43, no. 4, pp. 993–1004, 2019.

[4] H. Liu, X. Song, J. Bimbo, L. Seneviratne, and K. Althoefer, "Surface material recognition through haptic exploration using an intelligent contact sensing finger," in *2012 IEEE/RSJ international conference on intelligent robots and systems*, pp. 52–57, IEEE, 2012.

[5] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 2762, 2017.

[6] M. Lambeta, P.-W. Chou, S. Tian, B. Yang, B. Maloon, V. R. Most, D. Stroud, R. Santos, A. Byagowi, G. Kammerer, D. Jayaraman, and R. Calandra, "Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 3838–3845, 2020.

[7] W. Yuan, S. Wang, S. Dong, and E. Adelson, "Connecting look and feel: Associating the visual and tactile properties of physical materials," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4494–4502, 2017.

[8] R. Calandra, A. Owens, M. Upadhyaya, W. Yuan, J. Lin, E. H. Adelson, and S. Levine, "The feeling of success: Does touch sensing help predict grasp outcomes?," in *Conference on Robot Learning*, pp. 314–323, PMLR, 2017.

[9] G. Cao, Y. Zhou, D. Bollegala, and S. Luo, "Spatio-temporal attention model for tactile texture recognition," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9896–9902, IEEE, 2020.

[10] W. Yuan, Y. Mo, S. Wang, and E. H. Adelson, "Active clothing material perception using tactile sensing and deep learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4842–4849, IEEE, 2018.

[11] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, "Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8943–8950, IEEE, 2019.

[12] F. Yang, C. Ma, J. Zhang, J. Zhu, W. Yuan, and A. Owens, "Touch and go: Learning from human-collected vision and touch," in *Advances in Neural Information Processing Systems*, vol. 35, pp. 24130–24141, 2022.

[13] C. Higuera, A. Sharma, C. K. Bodduluri, T. Fan, P. Lancaster, M. Kalakrishnan, M. Kaess, B. Boots, M. Lambeta, T. Wu, *et al.*, "Sparsh: Self-supervised touch representations for vision-based tactile sensing," *arXiv preprint arXiv:2410.24090*, 2024.

[14] F. Yang, C. Feng, Z. Chen, H. Park, D. Wang, Y. Dou, Z. Zeng, X. Chen, R. Gangopadhyay, A. Owens, *et al.*, "Binding touch to everything: Learning unified multimodal tactile representations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26340–26353, 2024.

[15] G. Cao, J. Jiang, D. Bollegala, M. Li, and S. Luo, "Multimodal zero-shot learning for tactile texture recognition," *Robotics and Autonomous Systems*, vol. 176, p. 104688, 2024.

[16] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. Martinez Perez-Tejada, M. Jarber, S. Selber, T. Hollis, K. Tenney, *et al.*, "Robotic learning of haptic adjectives through physical interaction," *Robotics and Autonomous Systems*, vol. 63, pp. 279–292, 2015.

[17] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, pp. 8748–8763, PmLR, 2021.

[18] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *International conference on machine learning*, pp. 12888–12900, PMLR, 2022.

[19] J. Li, D. Li, S. Savarese, and S. Hoi, "BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *Proceedings of the 40th International Conference on Machine Learning*, vol. 202 of *Proceedings of Machine Learning Research*, pp. 19730–19742, PMLR, 2023.

[20] L. Fu, G. Datta, H. Huang, W. C.-H. Panitch, J. Drake, J. Ortiz, M. Mukadam, M. Lambeta, R. Calandra, and K. Goldberg, "A touch, vision, and language dataset for multimodal alignment," in *Forty-first International Conference on Machine Learning*, 2024.

[21] N. Cheng, J. Xu, C. Guan, J. Gao, W. Wang, Y. Li, F. Meng, J. Zhou, B. Fang, and W. Han, "Touch100k: A large-scale touch-language-vision dataset for touch-centric multimodal representation," *Information Fusion*, p. 103305, 2025.

[22] G. Cao, J. Jiang, N. Mao, D. Bollegala, M. Li, and S. Luo, "Vis2hap: Vision-based haptic rendering by cross-modal generation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 12443–12449, IEEE, 2023.

[23] R. Gao, W. Yuan, and J.-Y. Zhu, "Controllable visual-tactile synthesis," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7040–7052, 2023.

[24] D. F. Gomes, P. Paoletti, and S. Luo, "Generation of gelsight tactile images for sim2real learning," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 4177–4184, 2021.
[25] S. Yu, K. Lin, A. Xiao, J. Duan, and H. Soh, "Octopi: Object property reasoning with large tactile-language models," in *Proceedings of Robotics: Science and Systems (RSS)*, 2024.
[26] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres, "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV)," in *International Conference on Machine Learning*, pp. 2668–2677, PMLR, 2018.
[27] A. M. Turner, L. Thiergart, G. Leech, D. Udell, J. J. Vazquez, U. Mini, and M. MacDiarmid, "Steering language models with activation engineering," *arXiv preprint arXiv:2308.10248*, 2023.
[28] N. Rimsky, N. Gabrieli, J. Schulz, M. Tong, E. Hubinger, and A. Turner, "Steering llama 2 via contrastive activation addition," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, Association for Computational Linguistics, 2024.
[29] A. Zou, L. Phan, S. Chen, J. Campbell, P. Guo, R. Ren, A. Pan, X. Yin, M. Mazeika, A.-K. Dombrowski, S. Goel, N. Li, M. J. Byun, Z. Wang, A. Mallen, S. Basart, S. Koyejo, D. Song, M. Fredrikson, J. Z. Kolter, and D. Hendrycks, "Representation engineering: A top-down approach to AI transparency," *arXiv preprint arXiv:2310.01405*, 2023.
[30] J. Kerr, H. Huang, A. Wilcox, R. Hoque, J. Ichnowski, R. Calandra, and K. Goldberg, "Self-supervised visuo-tactile pretraining to locate and follow garment features," *arXiv preprint arXiv:2209.13042*, 2022.
[31] S. Bai, Y. Cai, R. Chen, K. Chen, *et al.*, "Qwen3-vl technical report," *arXiv preprint arXiv:2511.21631*, 2025.
[32] N. Subramani, N. Suresh, and M. Peters, "Extracting latent steering vectors from pretrained language models," in *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 566–581, Association for Computational Linguistics, 2022.
[33] K. Li, O. Patel, F. Viégas, H. Pfister, and M. Wattenberg, "Inference-time intervention: Eliciting truthful answers from a language model," in *Advances in Neural Information Processing Systems 36 (NeurIPS)*, 2023.
[34] Y. Zhou, M. Xu, J. Shi, Q. Li, and J. Chen, "Collaborative representation learning for alignment of tactile, language, and vision modalities," *arXiv preprint arXiv:2511.11512*, 2025.
[35] R. Feng, J. Hu, W. Xia, T. Gao, A. Shen, Y. Sun, B. Fang, and D. Hu, "Anytouch: Learning unified static-dynamic representation across multiple visuo-tactile sensors," *arXiv preprint arXiv:2502.12191*, 2025.
[36] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," in *Proceedings of the 38th International Conference on Machine Learning*, vol. 139 of *Proceedings of Machine Learning Research*, pp. 4904–4916, PMLR, 2021.
[37] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11975–11986, 2023.
[38] Q. Sun, Y. Fang, L. Wu, X. Wang, and Y. Cao, "EVA-CLIP: improved training techniques for CLIP at scale," *arXiv preprint arXiv:2303.15389*, 2023.
[39] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," in *Advances in Neural Information Processing Systems*, vol. 34, pp. 9694–9705, 2021.
[40] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, "Coca: Contrastive captioners are image-text foundation models," *Transactions on Machine Learning Research*, 2022.

APPENDIX

A. Training Details

Hardware. All experiments were conducted on a single workstation equipped with an NVIDIA GeForce RTX 5090 GPU (32 GB VRAM) and an AMD Ryzen 7 7800X3D 8-core CPU (16 threads).

Table IX summarises all training hyperparameters. Shared settings (optimiser, weight decay, learning rate schedule, gradient clipping, mixed precision, and seed) are common across all fine-tuned models and datasets. Per-model settings, including learning rate, warmup steps, and epochs, were independently optimised for each dataset via grid search on the respective validation splits. Zero-shot models (UniTouch ZS, TVL ZS) require no training and are evaluated directly with their pretrained weights. All augmentations are disabled during evaluation.

TABLE VII: **Out-of-distribution retrieval on the RAS dataset (%).** Ten held-out fabric materials, unseen during training, are evaluated against a candidate pool of 10 OOD materials. Results are averaged over 3 evaluation runs. TouchSteer achieves the highest R@1 and R@5, demonstrating that the steering mechanism enhances generalisation to novel materials. Best results are in **bold**. ZS: zero-shot; FT: fine-tuned.

Method	R@1	R@5
UniTouch, ZS	13.3±4.7	50.0±0.0
TVL, ZS	10.0±0.0	53.3±4.7
UniTouch, FT	30.0±8.2	60.0±0.0
TVL, FT	26.7±9.4	66.7±12.5
CLIP, FT	23.7±12.5	50.3±8.2
TouchSteer (Ours)	**33.3±9.4**	**73.3±4.7**

TABLE VIII: **Tactile description generation quality measured by BLEU-1 across all datasets.** TouchSteer is compared against three generative baselines, namely GIT-Base, ViT-GPT2, and CoCa ViT-B-32. The HCT column reports Mode 2 text descriptions. Higher BLEU-1 indicates stronger unigram overlap with the reference tactile descriptions.

	RAS-ID	**RAS-OOD**	**HCT Mode2**	**SSVTP**
GIT-Base	67.32	63.02	46.88	37.86
ViT-GPT2	63.53	60.12	43.12	36.13
CoCa ViT-B-32	62.11	62.11	58.40	49.37
TouchSteer (Ours)	73.6	73.5	62.3	54.7

B. t-SNE Object Categories

Table X lists the 20 representative objects selected for the t-SNE embedding visualisation in Figure 6. Objects span eight tactile categories, selected to maximise diversity of surface properties. Approximately 50 tactile images per object are randomly sampled from the HCT training set, totalling 978 images. The selection includes smooth rigid surfaces such as glass and plastic, coarse textures such as metal mesh and concrete, deformable fabrics, and objects with distinctive contact signatures such as ice and tennis ball.

C. Dataset Samples

Figure 7 presents representative tactile images from the three evaluation datasets used in this work. The RAS dataset captures high-resolution surface deformation patterns through a GelSight sensor, where each 640×480 image encodes micro-geometry as colour-coded normal maps. The HCT dataset contains DIGIT sensor images collected in uncontrolled settings with varying contact angles and pressures, reflecting realistic tactile interaction conditions. The SSVTP dataset provides DIGIT captures of material textures with consistent contact conditions.

TABLE IX: **Training hyperparameters.** Shared settings are common across all fine-tuned models. Per-model settings (learning rate, warmup steps, epochs, and TouchSteer-specific steering) were independently optimised for each dataset via grid search. ZS models are not trained.

	RAS	**HCT**	**SSVTP**
Shared settings (all fine-tuned models)			
Optimiser	AdamW		
Weight decay	10^{-2}		
LR schedule	Linear warmup $\rightarrow$ cosine annealing (to 10^{-7})		
Gradient clipping	Max norm 1.0		
Mixed precision	FP16		
Seed	42		
Image augmentation	Random rotation (0/90/180/270°) & flip (p=1/6 each)		
Text augmentation	None	Keyword shuffle + synonym replace (p=0.3)	
TouchSteer (Ours)			
Learning rate	10^{-5}	2×10^{-5}	10^{-5}
Warmup steps	200	600	200
Epochs / Batch size	10 / 16	10 / 16	15 / 16
Steering strength α	0.02 (P1) / 0.5 (P4)		
Injection position	P1+P4		
CLIP			
Learning rate	10^{-5}	2×10^{-5}	10^{-5}
Warmup steps	200	600	200
Epochs / Batch size	10 / 16	10 / 16	15 / 16
UniTouch, FT			
Learning rate	10^{-5}	2×10^{-5}	10^{-5}
Warmup steps	400	1000	400
Epochs / Batch size	8 / 16	8 / 16	12 / 16
TVL, FT			
Learning rate	5×10^{-6}	10^{-5}	5×10^{-6}
Warmup steps	300	800	300
Epochs / Batch size	5 / 16	5 / 16	10 / 16

TABLE X: **Object categories used in the t-SNE embedding visualisation.** Twenty objects spanning eight tactile categories are selected from the HCT dataset to represent diverse material properties.

Category	**Objects**
Metal	metal mesh, metal chain, metal frame
Fabric	fuzzy blanket, carpet, plaid blanket, knitted sweater, denim fabric
Glass	glass table, glass surface
Wood	wooden desk
Plastic	plastic bottle, plastic container, plastic case
Paper	cardboard
Rubber	sneaker sole
Misc.	tennis ball, coral rock, ice cube, concrete floor

The visual contrast between the three datasets illustrates the sensor-dependent nature of tactile representations: GelSight images emphasise surface relief through photometric stereo, while DIGIT images capture contact patterns through deformable membrane deformation. This diversity motivates the need for a cross-dataset evaluation framework.

D. Qualitative Retrieval Results

Figure 8 presents qualitative examples of language-tactile retrieval for all four models on the HCT dataset. For each text query, we show the top-5 retrieved tactile images ranked by cosine similarity. Red borders highlight correct matches, i.e., retrieved images belonging to the same object category as the query.

TouchSteer consistently retrieves the correct object at rank 1 across all four queries, while the other models show varying performance. For the metal mesh query (a), TouchSteer and UniTouch both achieve rank-1 retrieval, whereas CLIP retrieves a different metal mesh instance first and places the correct one at rank 2; TVL fails to retrieve the correct object within the top-5. For the game controller query (b), TouchSteer and CLIP both achieve rank-1, while TVL locates the correct object at rank 3; its top two results are unrelated objects (sweatshirt fabric, coral rock), indicating weaker cross-modal alignment. UniTouch retrieves a semantically similar "gaming controller" at rank 1 but misses the exact target object. For the foam mat query (c), TouchSteer and CLIP again achieve rank-1, whereas UniTouch retrieves the correct instance at rank 3 behind two other foam mat objects with different identities; TVL misses entirely. The plastic bottle query (d) is the only case where all four models retrieve the correct object within the top-5, yet with markedly different rankings: TouchSteer and CLIP both place the correct instance at rank 1, UniTouch locates it at rank 2, while TVL retrieves it only at rank 5; its top four results are semantically unrelated objects (pliers, knitted

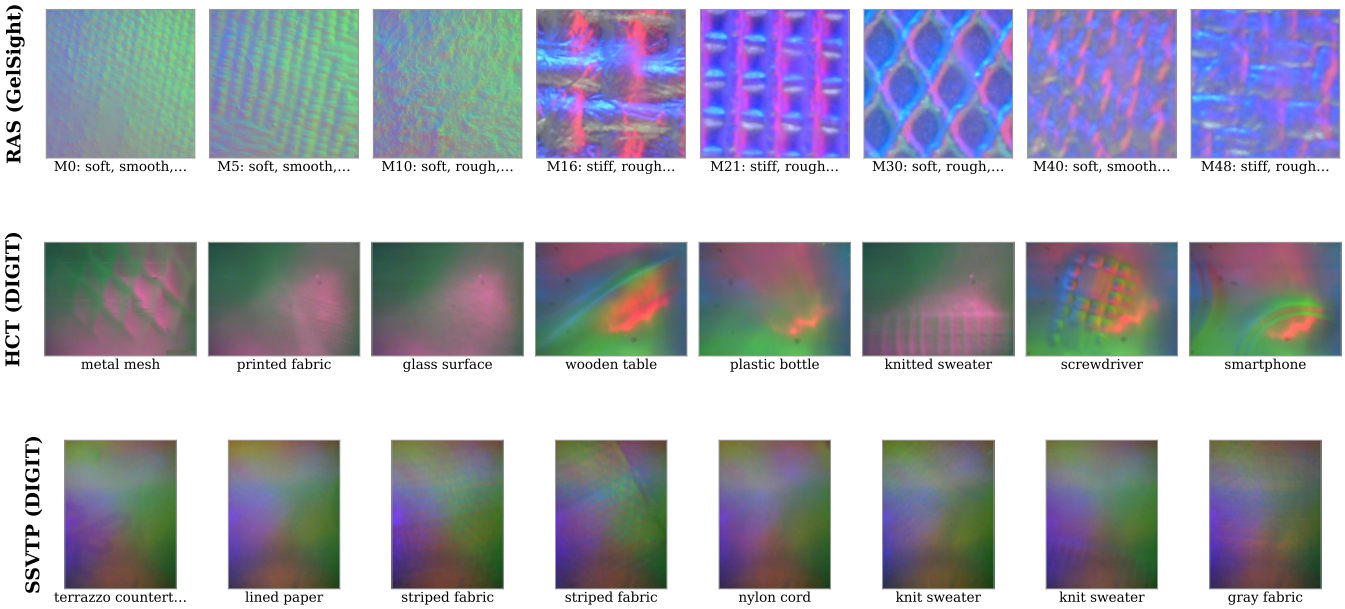


Fig. 7: **Representative tactile images from the three evaluation datasets.** *Top:* RAS (GelSight sensor): fabric materials with high-resolution surface deformation patterns. *Middle:* HCT (DIGIT sensor): everyday objects with diverse contact patterns. *Bottom:* SSVTP (DIGIT sensor): material textures with consistent contact conditions. The datasets span different sensing modalities, object categories, and collection protocols.

hat, book cover, glass bottle). These examples illustrate that TouchSteer's steering mechanism provides the most consistent retrieval accuracy, while baseline models exhibit query-dependent variability in ranking precision.

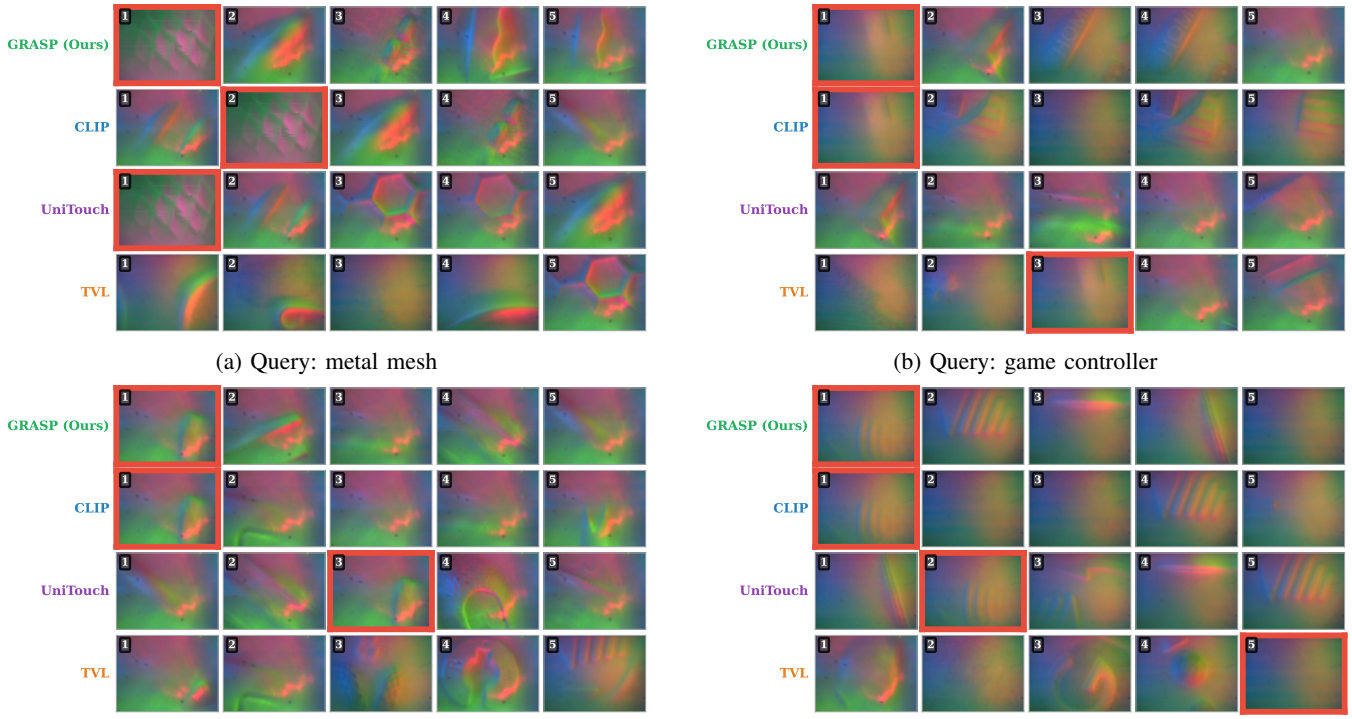
E. Failure Case Analysis

To better understand the limitations of the proposed model, we conduct a systematic analysis of TouchSteer's retrieval failures on the HCT dataset (Full Test, 356 queries, 279 unique objects) under the Mode 2 setting used in Table II. TouchSteer achieves 50.28% R@1 on this setting, corresponding to 177 failed queries. The observed failures are dominated by two recurring patterns: **intra-class instance confusion** and **cross-class material confusion**. Figure 9 illustrates three representative failure cases with paired RGB and tactile images alongside their text descriptions.

Intra-class instance confusion is illustrated by the carpet example in Figure 9(a). The query sample is a coarse loop-pile carpet with a rough, granular texture visible in the RGB image, while the incorrectly retrieved sample is a plush shag carpet with soft, fluffy fibres. Despite their visually distinct appearances, the DIGIT tactile images for both samples are nearly indistinguishable: both exhibit smooth, featureless colour gradients characteristic of the DIGIT sensor's response to soft, deformable surfaces. The sensor membrane conforms to the surface without capturing fine-grained structural differences. Even the text descriptions differ substantially: the query emphasises *grainy, abrasive, coarse*, while the retrieved sample is described as *plush, yielding, cushioned*, yet the model cannot disambiguate the two because the tactile signal itself carries insufficient discriminative information. This error pattern is especially common for *carpet*, where the test set contains 55 samples spanning diverse carpet types that all produce similar DIGIT responses.

Cross-class material confusion arises when physically distinct objects produce overlapping tactile signatures. Figure 9(b) shows a glass surface query incorrectly matched to a glass table. The RGB images reveal two different objects, a glass panel and a tabletop, yet both present flat, smooth, reflective surfaces to the DIGIT sensor. The resulting tactile images are both largely featureless colour fields with no surface relief. Critically, their text descriptions share three of five present attributes (*polished, rigid, cool*) and three of five absent attributes (*coarse, soft, warm*), with the only distinguishing terms being *matte/solid* vs. *lustrous/flat*, a subtle distinction that the embedding space cannot reliably separate. These glass-family confusions remain challenging across models, confirming that the ambiguity is rooted in the data rather than any specific model.

Figure 9(c) illustrates a more fundamental limitation: cross-material confusion between a plastic bottle and a glass bottle. Both objects are rigid, cool, and smooth-surfaced containers. The RGB images clearly distinguish the two (translucent plastic vs. dark tinted glass), but this visual distinction is invisible to tactile sensing. The DIGIT images both show smooth curved surfaces with minimal texture; the plastic bottle shows faint vertical striations, while the glass bottle shows a slight surface deformation pattern, but these differences are too subtle for reliable discrimination. The text descriptions are nearly synonymous: *sleek, rigid, solid, cool, reflective* (plastic) vs. *polished, rigid, cool, glossy, sleek* (glass), sharing 4 of 5 core attributes with only *reflective*↔*glossy* as the distinguishing term, which



(a) Query: metal mesh

(b) Query: game controller

(c) Query: foam mat

(d) Query: plastic bottle

Fig. 8: **Qualitative retrieval comparison on the HCT dataset.** For each query, the top-5 retrieved tactile images are shown for TouchSteer (Ours), CLIP, UniTouch, and TVL. Red borders indicate correct matches, i.e., images of the same object as the query. Rank numbers are shown in the top-left corner. TouchSteer achieves rank-1 retrieval on all four queries; all four models retrieve correctly in (d) with varied rankings; TVL retrieves correctly at rank 3 in (b) and rank 5 in (d).

are effectively interchangeable. This example highlights a fundamental boundary of tactile-only perception: material composition (plastic vs. glass) cannot be reliably inferred from surface contact alone when both materials share similar mechanical and thermal properties at the contact scale.

Table XI reports per-object R@1 for the 10 objects with the most test samples, further confirming these patterns. Objects with flat, uniform surfaces, such as *glass table* at 14% and *glass surface* at 29%, are consistently difficult across all models, while objects with distinctive textures, such as *plastic bottle* at 89% and *metal mesh* at 83%, achieve high accuracy. TouchSteer outperforms CLIP on objects with rich tactile textures, e.g. *plastic bottle* 89% vs. 67% and *metal mesh* 83% vs. 67%, while on smooth, featureless surfaces such as *glass surface*, the two models achieve comparable performance.

These failure cases reveal three key insights for future work: (1) the DIGIT sensor's limited spatial resolution on soft, deformable surfaces results in featureless tactile images that cannot distinguish instances within the same object category, motivating the incorporation of multi-touch or active exploration strategies; (2) objects with smooth, rigid surfaces produce nearly identical tactile signatures regardless of material composition, suggesting that complementary modalities such as thermal sensing or acoustic feedback may be necessary; and (3) the high overlap in text descriptions for tactile synonyms (*glossy*↔*reflective*, *sleek*↔*polished*) calls for attribute-aware contrastive losses that can weight fine-grained lexical distinctions.

TABLE XI: **Per-object R@1 (%) for objects with ≥5 test samples on HCT Full Test.** Objects with flat, uniform surfaces are hardest for all models. TouchSteer excels on textured objects; on smooth surfaces the two models achieve comparable performance.

Object	N	TouchSteer	CLIP	UniTouch	TVL
carpet	55	53	51	9	4
wooden table	15	47	40	20	7
shag carpet	12	75	83	8	0
plastic bottle	9	89	67	44	0
wooden guitar	8	75	75	38	0
glass table	7	14	29	14	0
glass surface	7	29	86	14	0
sign plate	6	50	100	50	0
metal mesh	6	83	67	50	0
woven fabric	6	67	50	50	0

F. Human Evaluation of Generated Descriptions

To assess the quality of VLM-generated tactile descriptions beyond automatic metrics, we design a human evaluation protocol. Evaluators are presented with each tactile image alongside its corresponding RGB image and the Qwen-VL-generated description (including both present and absent attribute sets). Each description is rated on five perceptual dimensions using a 5-point Likert scale. Table XII defines the evaluation criteria, and Table XIII provides the full rating scale with anchored descriptors for each dimension.

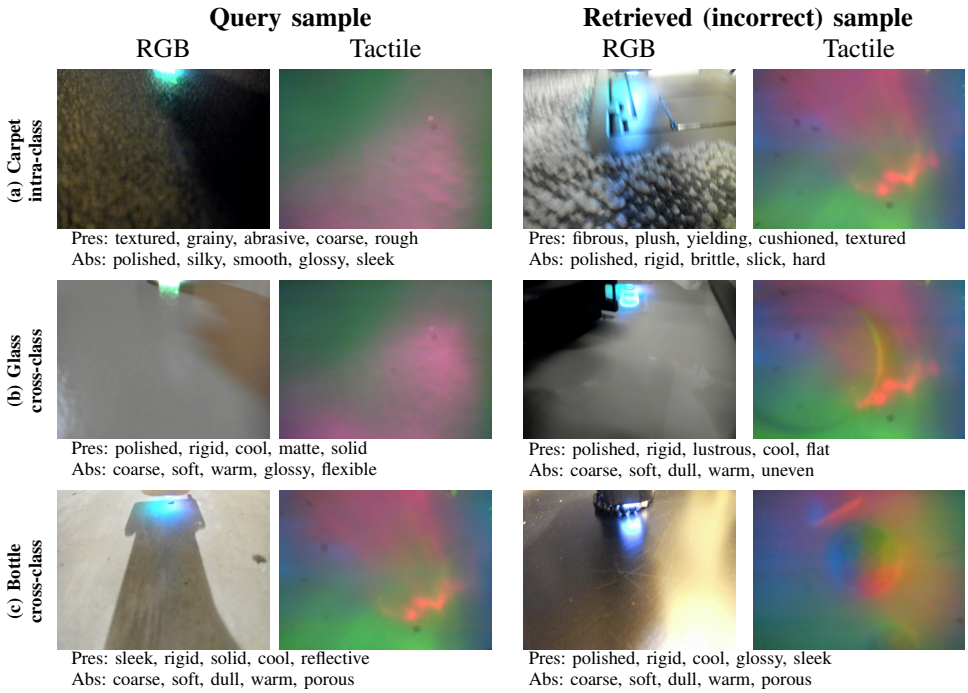


Fig. 9: **Representative failure cases on HCT Full Test.** Each row shows the query sample (left) and the incorrectly retrieved sample (right) with their RGB images, DIGIT tactile images, and text descriptions. **(a)** Intra-class confusion: two carpet instances with visually distinct appearances (coarse loop pile vs. plush shag) produce nearly indistinguishable DIGIT tactile images that both appear as featureless colour gradients. Although their text descriptions differ in most core attributes, they share partial overlap (e.g., both include *textured* as a present attribute and *polished* as an absent attribute), which compounds the retrieval ambiguity. **(b)** Cross-class confusion between glass surface and glass table: both RGB images show flat, smooth, reflective surfaces, and the tactile images are similarly featureless; the text descriptions share 3 of 5 present attributes (*polished, rigid, cool*), differing only in *matte/solid* vs. *lustrous/flat*. **(c)** Cross-class confusion between plastic bottle and glass bottle: both are rigid, cool, sleek containers whose tactile images show smooth curved surfaces with minimal distinguishing texture; the descriptions differ only in synonymous terms (*reflective* vs. *glossy*).

G. Text Description Examples

Table XIV presents representative examples from each evaluation dataset, showing paired RGB and tactile images alongside their original keyword captions and structured attribute descriptions. The structured descriptions consist of present attributes (properties the material possesses) and absent attributes (properties it lacks), further decomposed into anchor-level (shared object/material properties) and additional (per-sample) attributes for HCT and SSVTP. For RAS, descriptions are constructed from human-annotated binary attributes. These examples illustrate the contrast between the sparse original captions and the richer structured representations produced by the refinement pipeline described in Section III-A.

H. VLM Prompt for Description Augmentation

The structured attribute descriptions described in Section III-A are generated by prompting a vision–language model (Qwen3-VL [31]) with paired RGB and tactile images together with available metadata. Figure 11 presents the unified prompt template used across all datasets. The prompt instructs the VLM to produce four attribute lists organised into two levels of granularity: *anchor attributes* that capture stable, object-level material properties (e.g., "soft" for fabrics, "rigid" for metals), and *additional attributes* that reflect sample-specific observations grounded in the input images. A curated tactile vocabulary constrains the output space and ensures terminological consistency across samples. For datasets with multiple samples per object (HCT), the anchor attributes are first computed as the union of annotations across all samples of the same object; per-sample prompts then perform synonym variation on the anchors and generate additional attributes specific to each contact frame. For datasets with unique samples (SSVTP), both levels are generated in a single pass.

a) When Does Steering Help?: The experimental results reveal a nuanced picture of when steering vectors provide the most benefit. Steering is most effective on datasets with rich, structured text descriptions, namely HCT and SSVTP, where the semantic gap between general-purpose vision–language representations and tactile-specific semantics is largest. On these datasets, TouchSteer outperforms all baselines including fine-tuned CLIP, confirming that explicit semantic guidance through steering vectors provides information beyond what contrastive fine-tuning alone can extract.

In contrast, on the RAS dataset, where descriptions are derived from expert-voted binary attributes and are shorter and

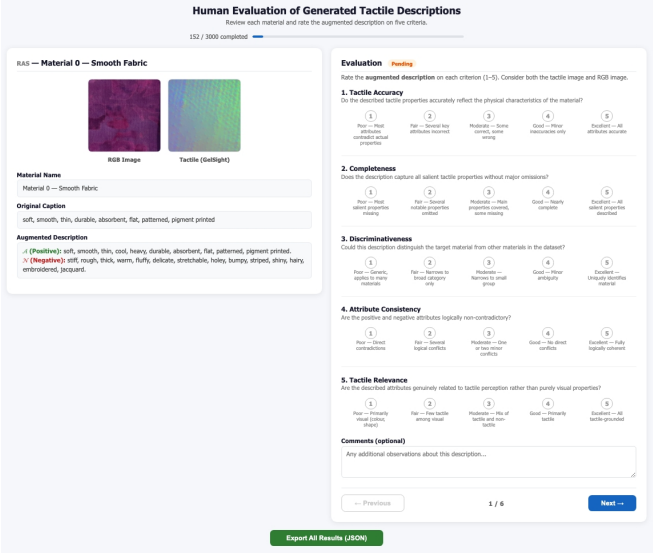


Fig. 10: **Screenshot of the human evaluation interface.** The left panel displays the material's RGB and tactile images together with the original caption and structured attribute description (present/absent attributes). The right panel presents the five evaluation criteria, each rated on a 1–5 Likert scale with anchored descriptors. Evaluators navigate between samples using the Previous/Next buttons.

TABLE XII: **Human evaluation criteria for VLM-generated tactile descriptions.** Five dimensions are designed to assess description quality from complementary perspectives: whether the attributes are correct, complete, distinctive, internally consistent, and grounded in tactile rather than visual perception.

Criterion	Definition
Tactile Accuracy	Do the described tactile properties (e.g., texture, hardness, surface finish) accurately reflect the physical characteristics of the material?
Completeness	Does the description capture all salient tactile properties of the material, without major omissions?
Discriminativeness	Could this description distinguish the target material from other materials in the dataset?
Attribute Consistency	Are the present attributes (properties the material possesses) and absent attributes (properties it lacks) logically non-contradictory?
Tactile Relevance	Are the described attributes genuinely related to tactile perception rather than purely visual or other non-tactile properties?

less semantically complex than the VLM-generated structured descriptions used in HCT and SSVTP, CLIP achieves comparable in-distribution performance. This suggests that when the text modality already conforms to the pretrained model's distributional assumptions, steering provides diminishing returns for in-distribution retrieval. However, steering still improves out-of-distribution generalisation, as shown in Table VII, indicating that the inductive bias toward tactile semantics helps the model extrapolate to unseen materials.

The inverted-U relationship between steering strength and retrieval accuracy, as shown in Figure 3, provides a practical guideline: moderate steering, with α between 0.01 and 0.05 for P1, yields the best trade-off, while excessive steering distorts the embedding geometry and degrades performance. This behaviour is consistent across datasets and evaluation settings.

b) Limitations: Several limitations should be acknowledged. First, the quality of steering vectors depends on the predefined positive prompt $\mathcal{A}$ and negative prompt $\mathcal{N}$. While these prompts were designed to be general-purpose, $\mathcal{A}$ may not capture domain-specific tactile properties (e.g., micro-textures in surgical applications or adhesive properties in manufacturing), and $\mathcal{N}$ may not cover all non-tactile confounders in specialised domains. Extending these prompts or learning them from data is a promising direction.

Second, the current framework employs a single, global steering vector applied uniformly to all material descriptions. In principle, materials with different dominant properties (e.g., fabrics vs. metals) could benefit from category-specific steering directions. The per-material attribute sets in our structured descriptions could serve as a basis for constructing adaptive steering vectors.

Third, our evaluation is limited to three tactile datasets collected with two sensor types (GelSight and DIGIT). While these datasets represent diverse evaluation scenarios, the transferability of the steering mechanism to other sensor modalities (e.g., BioTac, TacTip, magnetic-based sensors) or to significantly larger-scale datasets remains to be validated.

Fourth, the generated descriptions are evaluated using unigram-level metrics, namely BLEU-1 and Attr-F1. While these metrics are appropriate for the bag-of-adjectives structure of tactile descriptions, they do not capture semantic coherence or factual correctness at a deeper level. Human evaluation of generated descriptions would provide a more complete assessment.

c) Broader Applications: The ability to bridge tactile sensing and natural language has direct implications for human–robot interaction (HRI). In collaborative manipulation scenarios, a robot could describe the tactile properties of objects it encounters ("This surface is soft and slightly rough, with a fabric-like texture"), enabling human operators to make informed decisions without direct physical contact. This is particularly relevant in teleoperation, where tactile feedback is typically unavailable.

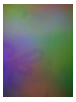
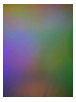
The retrieval capability enables a complementary interaction mode: a human operator could specify desired tactile properties in natural language ("Find a smooth, rigid surface"), and the robot could search its tactile experience database to identify matching materials. This natural language interface is more intuitive than traditional approaches that require specifying numerical thresholds on sensor readings.

Furthermore, the structured attribute representation, the present/absent attribute decomposition in text descriptions, together with the positive/negative prompt distinction ($\mathcal{A}/\mathcal{N}$) in steering, provides an interpretable intermediate layer between raw sensor data and language. This decomposition

TABLE XIII: **Rating scale definitions for human evaluation of VLM-generated tactile descriptions.** Each of the five criteria is rated on a 5-point Likert scale with anchored descriptors to ensure consistent interpretation across evaluators.

Score	1 (Poor)	2 (Fair)	3 (Moderate)	4 (Good)	5 (Excellent)
Tactile Accuracy	Most attributes contradict the material's actual tactile properties	Several key attributes are incorrect or misleading	Some attributes are correct but others are clearly wrong	Minor inaccuracies only; most attributes match the material	All described attributes accurately reflect the material's tactile properties
Completeness	Most salient tactile properties are missing from the description	Several notable properties are omitted	Main properties are covered but some are missing	Nearly complete; only minor details are omitted	All salient tactile properties are described
Discriminativeness	Generic description that could apply to many different materials	Narrows to a broad material category only (e.g., "fabric")	Narrows to a small group of similar materials	Identifies the material with only minor ambiguity	Uniquely identifies the target material among dataset candidates
Attribute Consistency	Present and absent attribute sets contain direct contradictions	Several logical conflicts between present and absent sets	One or two minor conflicts or questionable attribute assignments	No direct conflicts; all assignments are reasonable	Present and absent attributes are fully logically coherent
Tactile Relevance	Description focuses primarily on visual appearance (colour, shape)	Few tactile attributes among predominantly visual descriptions	Roughly equal mix of tactile and non-tactile attributes	Primarily tactile with only minor visual elements	All attributes are grounded in tactile perception (texture, hardness, compliance, temperature)

TABLE XIV: **Representative samples from each evaluation dataset with original and structured attribute descriptions.** Each example shows paired RGB and tactile images alongside the original keyword caption and the structured attribute description. RAS descriptions are derived from human-annotated binary attributes; HCT and SSVTP descriptions are generated by Qwen-VL and organised into present/absent anchor and additional attributes (Pres./Abs.). The structured descriptions provide substantially richer semantic content for cross-modal alignment.

Dataset	Material	RGB	Tactile	Original Caption	Structured Description (Pres./Abs.)
RAS	Material 0 (Smooth fabric)			soft, smooth, thin, durable, absorbent, flat, patterned, pigment printed	Pres: soft, smooth, thin, cool, heavy, durable, absorbent, flat, patterned, pigment printed. Abs: stiff, rough, thick, warm, fluffy, delicate, stretchable, holey, bumpy, striped, shiny, hairy, embroidered, jacquard.
	Material 1 (Stiff jacquard)			stiff, rough, thick, warm, absorbent, flat, patterned, jacquard	Pres: stiff, rough, thick, warm, heavy, durable, absorbent, flat, patterned, jacquard. Abs: soft, smooth, thin, cool, fluffy, delicate, stretchable, holey, bumpy, striped, shiny, hairy, embroidered, pigment printed.
HCT	Printed fabric			textured, patterned	Pres. anchor: textured, soft, flexible, woven, patterned. Abs. anchor: smooth, hard, rigid, glossy, metallic. Pres. add.: fibrous, matte, embossed, thin, plush. Abs. add.: gritty, shiny, stiff, dense, slick.
	Metal ornament			rigid, ridged, extruded	Pres. anchor: sleek, rigid, lustrous, cool, polished. Abs. anchor: coarse, soft, dull, warm, porous. Pres. add.: embossed, ridged, metallic, dense, patterned. Abs. add.: flat, smooth, matte, light, featureless.
SSVTP	Terrazzo countertop			smooth, glossy, patterned	Pres. anchor: hard, smooth, rigid, cool, durable. Abs. anchor: soft, rough, flexible, warm, fragile. Pres. add.: glossy, patterned, polished, flat, dense. Abs. add.: matte, uniform, textured, porous, uneven.
	Lined paper			flat, lined, hard	Pres. anchor: flat, hard, rigid, thin, matte. Abs. anchor: soft, flexible, yielding, thick, elastic. Pres. add.: lined, patterned, smooth, uniform, delicate. Abs. add.: rough, bumpy, gritty, fibrous, porous.

could support compositional queries ("softer than X but rougher than Y") and enable robots to reason about relative material properties, a capability that aligns with how humans naturally discuss and compare tactile experiences.

Beyond human–robot collaboration, the framework extends to scenarios where vision is occluded or degraded and tactile perception becomes the primary sensing modality. Examples include grasping inside opaque containers, exploring confined or cluttered spaces where object surfaces are partially hidden, and operating in darkness, smoke, or underwater environments where camera-based perception degrades significantly. In such settings, the ability to retrieve and describe tactile properties via natural language becomes critical for autonomous decision-making: a robot could identify a grasped object solely from its tactile signature, match it against a language-indexed material database, and generate a human-readable description, all without relying on visual input. The structured attribute representation ensures that the robot can reason about material properties (e.g., compliance, surface roughness, thermal conductivity) even when visual cues are entirely unavailable, making the framework applicable to a broader class of robotic tasks beyond those with co-located cameras.

d) Connection to Human Tactile Perception: The tactile/non-tactile attribute decomposition in our steering vector framework draws a parallel with findings in perceptual psychology. Human tactile perception is organised along a small number of perceptual dimensions (roughness, hardness, stickiness, and warmth) that collectively define a "tactile space". Our positive prompt $\mathcal{A}$ maps directly onto these

System role: You are a tactile perception expert.

Input:
1) **RGB Image**: visual appearance of the material at the contact point.
2) **Tactile Image**: GelSight sensor output capturing surface micro-texture.
3) **Object Name:** "{object_name}"
4) **Original Caption:** "{original_caption}" *(human-annotated tactile keywords)*

Candidate Tactile Vocabulary:
Texture: smooth, rough, coarse, grainy, bumpy, uneven, flat, polished, abrasive, gritty, silky, velvety, waxy, oily, sticky, slippery. *Hardness:* soft, hard, rigid, stiff, firm, yielding, elastic, stretchy, springy, flexible, pliable, spongy, solid, brittle. *Structure:* patterned, printed, striped, mesh, woven, knitted, embroidered, fibrous, porous, embossed, ridges, grooves, wrinkled. *Temperature:* cool, warm, neutral, cold. *Dimensions:* thin, thick, heavy, light, dense. *Surface:* shiny, matte, glossy, hairy, fuzzy, fluffy, woolly, plush, delicate, sharp. *Quality:* durable, fragile, tough, absorbing, repellent, synthetic, natural, metallic, plastic-like, rubbery, leathery.

Task: Generate four structured attribute lists at two levels of granularity.

Level 1: Anchor attributes (object level):
- **present_anchor** (4–5 attributes): fundamental properties of this material category, derived from the object name, original caption, and general material knowledge.
- **absent_anchor** (same count as present): attributes that are certainly **not** true for this material.

Level 2: Additional attributes (sample-specific observations):
- **present_additional** (3–5 attributes): fine-grained properties observed from the RGB and tactile images for this specific contact, distinct from anchor attributes.
- **absent_additional** (same count as present): contrasting attributes not present in this sample.

Output format (JSON):
```
{''present_anchor'': [...], ''absent_anchor'': [...],
''present_additional'': [...], ''absent_additional'': [...]}
```

Requirements:
1) Present and absent counts must be equal at each level.
2) No attribute may appear in both present and absent lists.
3) Anchor attributes capture material identity; additional attributes capture per-sample details.
4) All attributes must be grounded in the images and captions; do not hallucinate.
5) Use only vocabulary from the candidate tactile vocabulary list.

Fig. 11: **Unified prompt template for structured tactile attribute generation.** The prompt is sent to Qwen3-VL [31] together with paired RGB and tactile images. Placeholders {object_name} and {original_caption} are filled with per-sample metadata. The output is a structured JSON object containing four attribute lists at two levels of granularity, which serve as the structured text input for contrastive retrieval and generation training.

perceptual dimensions, while the negative prompt $\mathcal{N}$ captures visual and categorical confounders (colours, material names, geometric shapes) that lie outside the tactile space. This suggests that the steering mechanism aligns model representations with human perceptual structure by amplifying the tactile dimensions and suppressing the non-tactile ones.

The cross-modal similarity vectors used in our t-SNE analysis, shown in Figure 6, provide a quantitative counterpart to categorical perception in touch: objects with similar tactile properties cluster together, while objects with dissimilar properties are pushed apart. The progressive improvement from unsteered to $\mathcal{A}$-only to $\mathcal{A}-\mathcal{N}$ steering mirrors the cognitive process of tactile categorisation, where both the reinforcement of tactile cues and the suppression of non-tactile confounders contribute to material identification.

e) Scalability: The TouchSteer framework is designed for parameter efficiency. The steering mechanism adds no additional trainable parameters: the steering vector is computed from frozen word embeddings and applied as a fixed bias during inference. The shared ViT backbone serves both retrieval and generation, reducing the total parameter count compared to separate models for each task.

In terms of data requirements, the steering vector can be constructed from a small attribute prompt set of 25–50 words, making the approach applicable even when tactile data is scarce. The fine-tuning itself requires only standard contrastive and language modelling losses, with no specialised training procedures or auxiliary losses. These properties suggest that the framework can scale to larger datasets and additional sensor modalities with minimal architectural changes.

A natural extension would be to combine steering vectors with larger foundation models (e.g., scaling from BLIP-base to BLIP-2 or LLaVA-style architectures), potentially improving both retrieval accuracy and generation fluency. The modular nature of the steering mechanism, which operates on embeddings rather than model internals, makes such integration straightforward.

This section provides a more comprehensive survey of three lines of research that are closely related to the TouchSteer framework: steering vectors and representation engineering in NLP, recent tactile–language multimodal learning, and vision–language pretraining.

f) Steering Vectors and Representation Engineering: The concept of manipulating neural network behaviour through targeted modifications of internal representations has gained significant attention in the NLP and AI safety communities. Subramani et al. [32] first introduced the term *steering vectors* for language models, demonstrating that adding learned vectors to hidden states can steer text generation toward target sentences with near-perfect reconstruction accuracy. This work established the foundational insight that low-dimensional directions in representation space encode meaningful semantic properties.

Turner et al. [27] proposed *Activation Addition* (ActAdd), which computes steering vectors from the activation difference between contrastive prompt pairs (e.g., "Love" vs. "Hate") and adds them at inference time to shift model behaviour. This approach is methodologically closest to TouchSteer's steering mechanism: both construct a semantic direction from contrastive embeddings (positive vs. negative attributes in our case) and apply it as a fixed bias to shift representations toward a target subspace. However, while ActAdd operates on intermediate hidden states of autoregressive language models, TouchSteer applies steering in the projected retrieval embedding space and the word-embedding space of the decoder, requiring no access to intermediate activations.

Zou et al. [29] formalised *representation engineering* as a top-down paradigm for understanding and controlling neural networks through population-level representation interventions. Their framework provides the theoretical umbrella under which steering vectors operate, demonstrating that concepts such as honesty, fairness, and harmlessness can be identified as linear directions in representation space and manipulated accordingly. Li et al. [33] proposed *Inference-Time Intervention* (ITI), which identifies truthful directions in LLM activations via linear probes and applies activation shifts at inference time to elicit more truthful outputs. Rimsky et al. [28] extended the activation addition approach with *Contrastive Activation Addition* (CAA), demonstrating scalable behavioural steering of Llama 2 using contrastive pairs of behavioural examples.

While these works focus on steering language model behaviour (e.g., truthfulness, sentiment, safety), TouchSteer adapts the core principle to a cross-modal setting: we construct steering vectors from positive and negative prompts to bias vision–language embeddings toward tactile semantics. To our knowledge, this is the first application of steering vectors to cross-modal retrieval and generation in the tactile domain.

g) Recent Tactile–Language Multimodal Learning: The intersection of tactile sensing and language understanding has seen rapid progress. Beyond the baselines evaluated in our experiments, namely UniTouch [14] and TVL [20], several recent works have advanced tactile–language alignment.

Touch100k [21] introduces a large-scale touch–language–vision dataset comprising 100k samples with paired tactile images, natural language descriptions, and visual images, enabling touch-centric multimodal representation learning. While Touch100k demonstrates cross-modal understanding through predicting tactile properties from images, it focuses on indirect tactile–language generation rather than the direct tactile-to-language generation explored in TouchSteer.

TLV-CoRe [34] proposes a collaborative representation learning framework for aligning tactile, language, and vision modalities. It introduces a Sensor-Aware Modulator to unify tactile features across different sensor types and a Unified Bridging Adapter for tri-modal interaction, addressing the challenge of sensor heterogeneity in tactile learning.

AnyTouch [35] learns unified static-dynamic representations across multiple visuo-tactile sensors, addressing the challenge that different tactile sensors produce fundamentally different signal modalities. By learning sensor-agnostic representations, AnyTouch enables cross-sensor transfer, which complements TouchSteer's cross-modal (tactile–language) transfer capability.

Sparsh [13] introduces self-supervised pre-training methods (DINO, IJEPA) for vision-based tactile sensing, trained on over 460k tactile images. It establishes the TacBench benchmark for evaluating tactile representations across multiple downstream tasks. While Sparsh focuses on tactile-only representation learning without language alignment, it demonstrates that large-scale self-supervised pretraining can produce transferable tactile features, suggesting a potential direction for scaling TouchSteer's vision backbone.

Octopi [25] combines tactile encoders with large language models for physical property reasoning, contributing the PhysiCLeAR dataset for evaluating object property reasoning from tactile input. Octopi's focus on reasoning over physical properties is complementary to TouchSteer's focus on retrieval and description generation.

Compared to these works, TouchSteer is distinguished by its steering vector mechanism, which provides explicit, controllable manipulation of the shared embedding space rather than relying solely on end-to-end contrastive or generative training. This mechanism enables both fine-grained retrieval and description generation within a single architecture.

h) Vision–Language Pretraining: TouchSteer builds upon BLIP [18], which unifies vision–language understanding and generation through a multi-task pretraining framework. To contextualise this architectural choice, we survey the broader landscape of vision–language pretraining models.

CLIP [17] pioneered large-scale contrastive pretraining on 400M image–text pairs, learning a shared embedding space for zero-shot image–text retrieval and classification. ALIGN [36] scaled this approach to 1.8B noisy image–alt-text pairs, demonstrating that scale can compensate for noise in training data. Both CLIP and ALIGN are retrieval-only models that lack generation capability, which motivates our use of BLIP's unified architecture.

SigLIP [37] replaces the softmax-based contrastive loss of CLIP with a sigmoid loss that operates on image–text pairs independently, eliminating the need for a global normalisation across the batch. This enables more efficient training at scale and improved performance on fine-grained recognition tasks. EVA-CLIP [38] improves CLIP training

through better initialisation, optimisation, and augmentation strategies, achieving state-of-the-art zero-shot performance.

ALBEF [39] introduced momentum distillation and an align-before-fuse strategy that performs contrastive alignment before cross-modal fusion, a design principle adopted by BLIP. BLIP-2 [19] extends BLIP by bridging frozen image encoders and frozen large language models through a lightweight Querying Transformer (Q-Former), achieving strong performance with significantly fewer trainable parameters. CoCa [40] unifies contrastive and captioning objectives in a single encoder-decoder model, demonstrating that these two training signals are complementary, a principle that TouchSteer also exploits through its joint LTC and LM losses.

TouchSteer inherits BLIP's dual-task architecture (contrastive retrieval + autoregressive generation) and augments it with steering vectors that encode domain-specific (tactile) semantics. The steering mechanism is architecture-agnostic and could be applied to more recent VLP backbones such as BLIP-2 or CoCa to further improve performance, as discussed in Section VI.