computation

MDPI

Article

Enhancing Accessibility: Automated Tactile Graphics Generation for Individuals with Visual Impairments

Yehor Dzhurynskyi [1,*], Volodymyr Mayik [2] and Lyudmyla Mayik [3]

[1] SQUAD Ukraine LLC, 3 Korolenkivskya Str., 01033 Kyiv, Ukraine
[2] Department of Printing Technologies and Packaging (PPT), Institute of Printing Art and Media Technologies, Lviv Polytechnic National University, 28a, Stepan Bandera Str., 79013 Lviv, Ukraine; vol.mayik.2015@gmail.com
[3] Department of Multimedia Technologies (MT), Institute of Printing Art and Media Technologies, Lviv Polytechnic National University, 28a, Stepan Bandera Str., 79013 Lviv, Ukraine; ludmyla.maik@gmail.com
* Correspondence: y.a.dzhurynskyi@gmail.com

Abstract: This study addresses the accessibility challenges faced by individuals with visual impairments due to limited access to graphic information, which significantly impacts their educational and social integration. Traditional methods for producing tactile graphics are labor-intensive and require specialized expertise, limiting their availability. Recent advancements in generative models, such as GANs, diffusion models, and VAEs, offer potential solutions to automate the creation of tactile images. In this work, we propose a novel generative model conditioned on text prompts, integrating a Bidirectional and Auto-Regressive Transformer (BART) and Vector Quantized Variational Auto-Encoder (VQ-VAE). This model transforms textual descriptions into tactile graphics, addressing key requirements for legibility and accessibility. The model's performance was evaluated using cross-entropy, perplexity, mean square error, and CLIP Score metrics, demonstrating its ability to generate high-quality, customizable tactile images. Testing with educational and rehabilitation institutions confirmed the practicality and efficiency of the system, which significantly reduces production time and requires minimal operator expertise. The proposed approach enhances the production of inclusive educational materials, enabling improved access to quality education and fostering greater independence for individuals with visual impairments. Future research will focus on expanding the training dataset and refining the model for complex scenarios.

Keywords: accessibility; visual impairment; tactile graphics; generative model; machine learning; artificial intelligence

Citation: Dzhurynskyi, Y.; Mayik, V.; Mayik, L. Enhancing Accessibility: Automated Tactile Graphics Generation for Individuals with Visual Impairments. *Computation* **2024**, *12*, 251. https://doi.org/10.3390/computation12120251

Academic Editors: Dmytro Chumachenko and Sergiy Yakovlev

Received: 20 November 2024
Revised: 17 December 2024
Accepted: 20 December 2024
Published: 23 December 2024

1. Introduction

The modern advancement of an inclusive society underscores the importance of integrating people with visual impairments into all aspects of social life. The challenges of socializing individuals with such disabilities span several factors that hinder their education and active participation in society. Particularly, people with visual impairments face limited access to information, as a substantial portion of materials is available solely in standard printed or digital formats. This challenge is further compounded by the growing amount of information presented in graphic form, which aims to enhance comprehension for the general audience. These barriers restrict opportunities for people with visual impairments to access quality education and pursue professional growth [1–3]. This issue has become particularly pressing, as shown by data from the Vision Loss Expert Group [4]: In 2020, approximately 295 million people worldwide were affected by vision impairment, and this number is projected to rise to 703 million by 2050.

In addition, an analysis [5] of the activities of publishing and printing industry enterprises specializing in educational and instructional literature (such as textbooks and manuals) for people with visual impairments has identified several challenges related to the

creation and adaptation of images and illustrative material, which are especially critical for this type of publication. Key difficulties faced by these enterprises in creating or adapting graphic content include a shortage of qualified specialists with the technical skills necessary for producing tactile graphics, the additional time and financial investment required to train such specialists, and the high labor intensity and costs associated with producing or adapting tactile images. Also, the analysis indicated that production complexity is a major factor contributing to the limited availability of tactile graphics.

2. Related Work

Efforts to automate tactile graphics generation have largely focused on transforming visual content into tactile representations. These approaches can be categorized into methods that rely on direct image-to-tactile transformations and those that incorporate semantic processing.

2.1. Direct Image-to-Tactile Transformations

Models such as those described by Way, Barner, and others [6–9] utilize edge detection algorithms like the Canny edge detector. These techniques aim to extract contours and boundaries from visual content and render them into tactile formats. While straightforward, these methods suffer from challenges such as spurious edges and omitted critical details, which hinder the comprehensibility of the tactile output. Furthermore, these edge-based methods fail to accommodate the semantic requirements of tactile graphics, often including extraneous elements or insufficient simplifications that impede interpretation by touch, which violates the requirements for tactile graphics [10,11].

2.2. Semantic-Based Approaches

Recent advancements in semantic segmentation and object detection have enabled more nuanced methods [12,13]. The Pic2Tac system exemplifies this trend by employing state-of-the-art computer vision techniques such as Mask R-CNN for object detection and PiCANet for saliency detection. Pic2Tac translates salient objects into tactile "icons" and fills background regions with predefined patterns, thus addressing some of the limitations of edge-detection methods. However, its reliance on preexisting visual content (photographs) restricts its applicability when such content is unavailable. Moreover, the patterns used for backgrounds are predefined and may not adequately represent complex scenes.

Another notable approach by Pakenaitė et al. [12,13] combines object detection and salience analysis to create tactile collages. While effective for specific applications, this method often struggles with providing adequate background context, leading to ambiguities in the interpretation of foreground elements.

2.3. Advancements in Generative Models

Generative Adversarial Networks (GANs) [14] and Variational Autoencoders (VAEs) [15] have been explored in image synthesis, but their application in tactile graphics remains nascent. GANs, for instance, excel in generating high-resolution images, but often require extensive training datasets and struggle with generating discrete patterns suited for tactile media. Diffusion models [16] offer promising avenues, but have yet to be tailored for accessibility-focused use cases.

2.4. Comparison with Our Approach

The methods described above either focus on transforming existing visual content into tactile formats or utilize limited sets of predefined tactile elements. In contrast, our approach combines the strengths of Bidirectional and Auto-Regressive Transformer (BART) [17,18] and Vector Quantized Variational Auto-Encoder (VQ-VAE) [19] to produce tactile graphics that meet accessibility standards while being computationally efficient and scalable. This integration represents a novel application of generative modeling in the domain of tactile

accessibility. This innovation not only streamlines the production process but also enhances accessibility, as it enables the creation of tactile graphics solely from descriptive text.

Our method integrates a BART with a VQ-VAE, uniquely partitioning the latent space into independent text and graphic embeddings. This architecture ensures that the generated tactile graphics adhere to the specific requirements of accessibility while allowing for greater variability and customization. By leveraging generative models rather than static transformations, our approach introduces a level of adaptability and semantic richness that previous methods lacked.

3. Text-Conditioned Tactile Graphics Generative Model

The transformation of semantic text descriptions into tactile representations builds upon theories of latent space partitioning and generative modeling. Specifically, methods such as BART and VQ-VAE have proven effective for mapping high-dimensional data into interpretable outputs suitable for tactile exploration. The subject of its modeling is the process of converting text information into graphic information. To accomplish this, the embedded space of the transformer, which was formed during language modeling on pretraining task, was divided into two independent embedded spaces—text and graphics—instead of a shared one.

In this study, the developed model was trained using the parameters presented in Tables 1 and 2 for the VQ-VAE and BART models, respectively. At the same time, the parameters of the BART's text-embedded space remained the same as during language modeling.

Table 1. VQ-VAE parameters.

Image Dimension	$256 \times 256 \times 1$
"Codebook" size	512
Latent vectors size	16
Number of hidden layers	5
Hidden layers dimension	16

Table 2. BART parameters.

	Encoder	**Decoder**
Dictionary size	8192	512 + 1 (BOS token)
Sequence size	64	64 + 1 (BOS token)
Number of layers	3	3
Layer dimension	512	512
FFN dimension	1024	1024
Number of attention heads	8	8

Simultaneously, the parameters of the BART's graphic embedded space were adjusted so that the dimension of the embedded space was equal to the size of the VQ-VAE's "codebook" and the dimensionality of the vectors of the graphic embedded space was equal to the dimensionality of the latent space 2d vectors of the VQ-VAE model calculated with the following equation:

$$\dim(z) = (\frac{W}{2^n}, \frac{H}{2^n}), \quad (1)$$

where z is a latent 2d vector of the VQ-VAE; W and H represent the width and height of the image, respectively; and n is the number of hidden layers.

Therefore, the latent vector of the VQ-VAE has dimensions of 8×8, while the BART decoder's sequence size should be equal to this.

Also, it is important to note that the size of the decoder's dictionary and the size of the sequence are each increased by one unit compared to the original values. This adjustment

is necessary to introduce an additional service token (i.e., BOS token), which is added at the beginning of the sequence to facilitate autoregressive image generation.

Before generating text tokens using the BPE [20] tokenization model, the original text components are normalized to achieve a uniform format, including converting all uppercase letters to lowercase.

Formally, the process of converting text tokens into graphic tokens using a text-conditional tactile graphics generation model is described in successive stages. The structural and functional diagram that depicts these stages is shown in Figure 1.

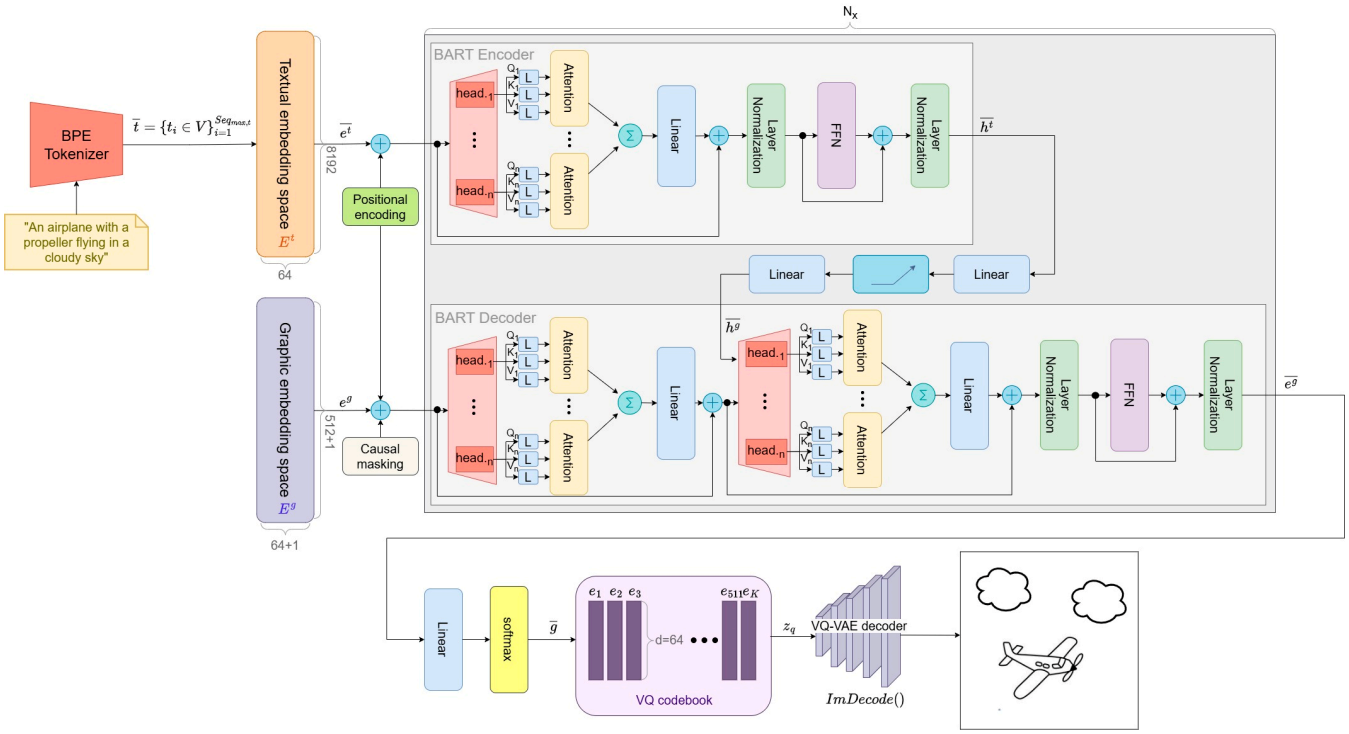


Figure 1. Structural and functional diagram of the text-conditioned tactile graphics generative model.

The first step is to generate a bounded sequence of text tokens based on a text prompt: $\overline{t} = \{t_i \in V\}_{i=1}^{Seq_{max,t}}$, where $\overline{t}$ is a sequence of text tokens of dimensions $Seq_{max,t} = 64$, and V is a dictionary of text tokens. If the size of the generated sequence of text tokens exceeds the value, its size is reduced to the maximum value, discarding the excess tokens. If the size of the generated sequence of text tokens is smaller than the value, its size is increased to the maximum value by adding service $\langle$PAD$\rangle$ tokens that do not affect the simulation result.

In the next step, the text tokens that form the sequence $\overline{t}$ are mapped to the text-embedded space vectors e_k, forming a subset of it:

$$\overline{e^t} = \left\{e_k^t \in E^t \middle| k = t_i \in \overline{t}\right\}_{i=1}^{Seq_{max,t}}; \overline{e^t} \subseteq E^t, \tag{2}$$

where $\overline{t}$ is the sequence of text tokens; E^t is a text-embedded space; e_k^t are elements of the text-embedded space. Elements e_k^t reflect the semantic meaning of text tokens in the embedded space.

Next, the vectors of the text-embedded space E^t are transformed by the transformer's bidirectional encoder, which is formed from several layers, forming hidden states $\overline{h^t}$. The bidirectionality of the encoder means that it analyzes the full context of an individual vector of the embedded space, considering both the previous and the following elements of the sequence:

$$\overline{h^t} = Encode(\overline{e^t}); \overline{h^t} \subseteq E^t, \tag{3}$$

where $\overline{h^t}$ is the hidden state of the encoder and $Encode(\cdot)$ is the transformer's encoding operation, defined within [17].

The hidden state of the encoder $\overline{h^t}$ is then converted by linear layers and a nonlinear activation function to the hidden state of the decoder (i.e., graphic information), forming a subset of the graphic embedded space E^g:

$$\overline{h^g} = Linear_2 \circ ReLU \circ Linear_1(\overline{h^t}), \tag{4}$$

where $\overline{h^t} \subseteq E^t$ is the hidden state of the encoder; $\overline{h^g} \subseteq E^g$ is the hidden state of the decoder; $Linear_i$ is a linear layer; and $ReLU \stackrel{\text{def}}{=} \max(0, x)$ is a non-linear layer activation function.

At the next stage, an autoregressive [18] Transformer's decoder is used. This means that the decoder generates one graphics token per iteration, considering the context of the previously generated graphics tokens. Thus, during the decoding process, the model performs calculations based on the hidden state $\overline{h^t}$ and pre-generated elements of the vector sequence of the graphic embedded space e^g_k or $\overline{h^g}$:

$$e^g_i = Decode(\overline{h^t}, e^g_1, e^g_2, \ldots, e^g_{i-1}); i \leq d_z, \tag{5}$$

where e^g_i is the i-th element of the vector sequence of the graphic embedded space E^g, $e^g_j; j < i$ are previously generated vectors of the graphic embedded space; $\overline{h^g}$ is the hidden state of the decoder; d_z is the size of the final sequence $\overline{e^g} \subseteq E^g$; and $Decode(\cdot)$ is the transformer's decoding operation, defined within [17].

Decoding occurs in an iterative manner until the sequence $\overline{e^g}$ size is equal to d_z (i.e., the size of the latent space vector of the VQ-VAE model). Once the decoding is complete, the resulting sequence of vectors of the graphics-embedded space $\overline{e^g}$ is converted by a linear layer and $Softmax$ function into a sequence of probability distributions from which the element with the highest probability is selected, determining the selected graphics token:

$$\overline{g} = \{g_i\}_{i=1}^{d_z};\ g_i = argmax(Softmax\ \circ\ Linear(e^g_i)), \tag{6}$$

where $\overline{g}$ is the generated sequence of graphic tokens of size d_z and $e^g_i \in \overline{e^g}$ is an element of the vector sequence of the graphics-embedded space E^g.

In the next step, on the basis of graphic tokens (6), a sequence of latent quantized vectors is formed, z_q, which is defined by the Formula (7). Each graphic token: $1 \leq g_i \leq K$; $i = 1 \ldots d_z$, is the positional number of the quantized vector in the "codebook" of the VQ-VAE model:

$$z_q = \{e_k \in Z | k = g_i \in \overline{g}\}_{i=1}^{d_z};\ z_q \subseteq\ Z, \tag{7}$$

where Z is the set of latent quantized vectors, or "codebook"; $z_q \subseteq Z$ is a sequence of latent quantized vectors; $g_i \in \overline{g}$ is a graphics token; and d_z is the size of the sequence of latent quantized vectors.

The final step is the generation of tactile graphics using a sequence (7) with a VQ-VAE decoder:

$$Y = ImDecode(z_q), \tag{8}$$

where z_q is the sequence of latent quantized vectors; $ImDecode(\cdot)$ is an image-decoding operation based on latent representation defined within [19]; and Y is a generated tactile image.

4. Method

4.1. Dataset

Language modeling was conducted using the BrUK corpus [21], a collection of Ukrainian texts sourced from various domains. Unlike textual datasets (i.e., corpora), which are widely accessible, tactile graphics samples are significantly less common. One of the main challenges in modeling tactile graphics generation using machine learning is

the limited availability of publicly accessible samples, as the tactile graphics production industry is far less developed compared to traditional publishing.

To address this limitation, a collection of plant and animal images from the APH Tactile Graphics Library [22] was selected as the primary dataset for training the model. Additionally, the dataset was augmented with 41 custom tactile image samples, bringing the total to 179 samples. These custom samples, derived from simple illustrations of animals, were previously used by a Ukrainian institution specializing in preschool education for children with visual impairments. The samples were reviewed by field specialists and have been integrated into a range of books for visually impaired children. Tactile graphics samples are effectively represented as grayscale rasterized images, pre-processed to enhance the model's performance. This was achieved by quantizing their pixel values from the [0, 255] range into a finite set of discrete values (i.e., a "palette") using the following formula:

$$\acute{x} = \underset{p \in P}{\operatorname{argmin}} \sqrt{\sum_{i=1}^{n} (x_i - p_i)^2}, \tag{9}$$

where x is the original value of the pixel color in the form of a vector of dimensions from 1 to 4; P is a pre-defined color palette, one of which color values the target pixel acquires; n is the size of the pixel color vector; and $\acute{x}$ is the quantized value of the color of the original pixel.

4.2. Training

Initially, the language-modeling process focused on training the BART encoder to establish the textual embedding space. However, it is important to clarify why we chose not to use an existing pre-trained Transformer model for fine-tuning and instead conducted the modeling independently. The primary reason was the specific requirement to perform modeling for the Ukrainian language, for which no suitable pre-trained models were available from publicly accessible sources. Consequently, this step is not mandatory and can be reproduced using any existing Transformer model trained on generation tasks.

The training process leveraged the pre-trained encoder from BART and the "codebook" and decoder from VQ-VAE, which were excluded from the training graph. Instead, the BART decoder was trained to operate within its own graphics embedding space, while the VQ-VAE encoder was not utilized at this stage.

Given the limited number of samples in the tactile graphics dataset, the training process was conducted over 35,000 iterations. The hyperparameters utilized during training are summarized in Table 3.

Table 3. Hyperparameters of the text-conditioned tactile graphics generative model.

Batch Size	**2**
Learning rate	0.0001
Weight decay	0.001
Optimizer	AdamW
Activation function	ReLU, GeLU

To address potential annotation ambiguities, such as between "a stretching cat" and "a cat", we designed an evaluation framework that prioritizes semantic understanding using the CLIP Score metric, which accounts for partial matches. Additionally, the dataset was revised to include multi-label annotations, allowing for broader descriptions (e.g., both "a stretching cat" and "a cat"). This was complemented by data augmentation with varied textual descriptions to improve model generalization. Finally, a qualitative human evaluation phase ensured that semantically valid predictions were not unfairly penalized, enhancing the robustness and flexibility of the model in real-world scenarios.

4.3. Evaluation Method

The model's performance was evaluated separately for each of its components: BART and VQ-VAE. The Cross-Entropy (10) metric was used to assess how effectively the model converted text prompts into corresponding graphic tokens, while Perplexity (11) measured the level of uncertainty in the model's predictions.

$$\mathrm{H}(\mathrm{P},\mathrm{Q}) = -\sum_{\mathrm{x}} \mathrm{P}(\mathrm{x}) \log \mathrm{Q}(\mathrm{x}); \tag{10}$$

$$PPL(P,\ Q) = 2^{H(P,Q)}, \tag{11}$$

where Q is a corpus distribution and P is an empirical distribution.

The model's evaluation, which assessed its ability to reconstruct the original image and synthesize new ones, was performed by calculating the Mean Square Error (MSE) in both the image space and the low-dimensional latent space, as well as the Fréchet Inception Distance.

The MSE (12) estimate reflects the model's ability to reproduce the original image. This metric was computed for both the raster image space and the latent space. Formally, the evaluation is expressed as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{x}_i)^2 \tag{12}$$

The Fréchet Inception Distance (13) is a metric used to evaluate the model's ability to synthesize images by comparing them to the original images. It is calculated as follows:

$$FID = \|\mu - \mu_w\|^2 + tr\left(\Sigma + \Sigma_w - 2(\Sigma\Sigma_w)^{\frac{1}{2}}\right), \tag{13}$$

where the distributions of real and synthesized images are defined as $N(\mu, \Sigma)$ and $N(\mu_w, \Sigma_w)$, respectively.

In addition, the overall performance of the model was assessed using the CLIP Score (14) metric [23], which measures the model's ability to translate textual information into graphical representations.

$$CLIPScore(I, C) = \max(100 * \cos(E_I,\ E_C),\ 0), \tag{14}$$

where I is the original image, C is the source text prompt, $\cos(\cdot)$ denotes the cosine similarity, E_I represents the latent embedding of the image generated by the CLIP model, and E_C represents the latent embedding of the text generated by the CLIP model.

5. Results

5.1. Experiment

This section presents the results of the experiment through examples of tactile images generated by the developed model. The generation process was performed using the "greedy search" method, which identifies the shortest path in a directed graph representation of the generation process to synthesize the corresponding image. These examples demonstrate the effectiveness of the variable synthesis capability of the complex tactile graphics generation model, showcasing its ability to produce tactile images based on text prompts of varying descriptions. This text-prompt-based generation approach significantly enhanced the controllability of the process, making it both convenient and accessible, even for individuals without expertise in tactile graphics production.

The results of the experiment include samples of generated tactile graphics images based on various types of text prompts, such as monosyllabic prompts, prompts with numerals, and prompts with epithets. These samples are presented in Figure 2.

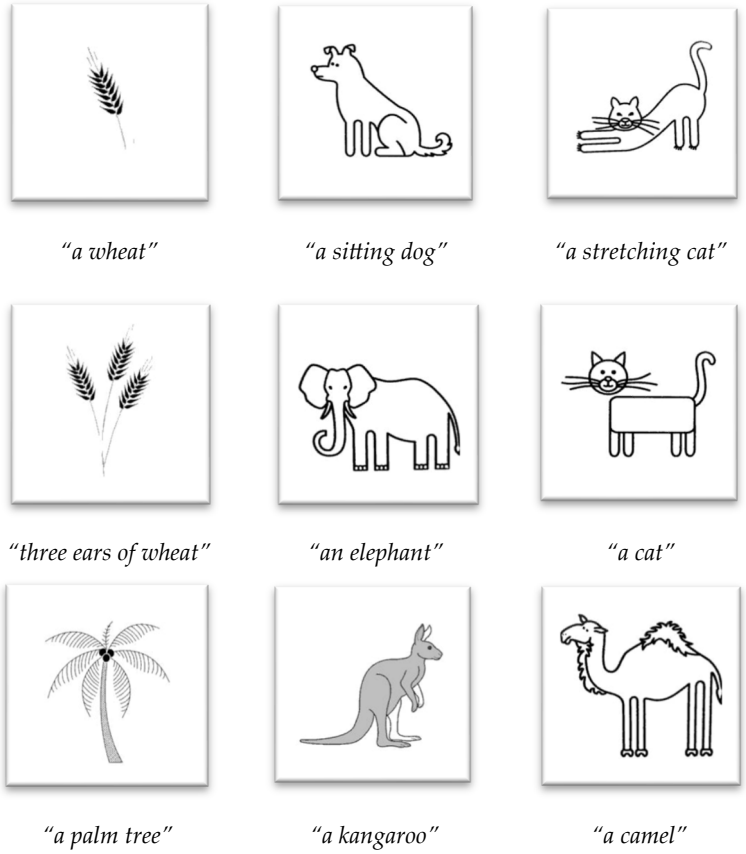


Figure 2. Samples of generated images, each determined by the text prompt provided below the corresponding image (note that, in practice, text prompts were provided in Ukrainian, but they were translated for convenience).

5.2. Comparison with Baseline

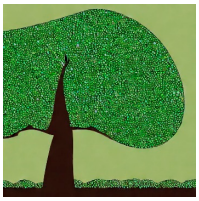
Table 4 presents a comparative evaluation of the developed text-conditioned tactile graphics generative model against other state-of-the-art models, including MidJourney, Stable Diffusion, DALL-E, and DALL-E 2. Each model was assessed based on its text interpretation and image generation mechanisms, as well as the resulting output quality and suitability for tactile graphics.

During the experiment, a consistent text prompt, "coloring page for kids, black outline, white background, tactile graphics, a tree, cartoon style, very low detail, no shading", was used for MidJourney V6.1, Stable Diffusion 3.5, DALL-E, and DALL-E 2. For our model, a simplified prompt, "a tree", was used, as it relies on direct semantic mapping from text to tactile graphics. The following generation parameters were applied: temperature = 0.75 and top-k = 3.

Our proposed model, which integrates a VQ-VAE with a BART Transformer, demonstrated clear advantages in generating simplified and accessible tactile graphics. Unlike the outputs from other models, which prioritize photorealism or complex visual textures, our approach produces clean, interpretable shapes optimized for tactile exploration. Notably, models like MidJourney and Stable Diffusion rely on CLIP-based image generation, which, while effective for general visual synthesis, lacks the precision needed for tactile requirements.

The outputs generated by DALL-E and DALL-E 2, while incorporating VQ-VAE and diffusion techniques, often include extraneous details that complicate tactile interpretation. In contrast, our method prioritizes simplicity and clarity, ensuring that the generated outputs align with accessibility standards for visually impaired users.

Table 4. Evaluation results of the developed model against other state-of-the-art models.

Model	Text Interpretation	Image Generation			
Midjourney	Diffusion Model	Proprietary (most likely CLIP)			
Stable Diffusion	Latent Diffusion Model	CLIP			
DALL-E	VQ-VAE	GPT			
DALL-E 2	Diffusion Model	CLIP			
Proposed Model [1]	VQ-VAE	BART			

[1] The text-conditioned tactile graphics generative model.

5.3. Expert Assessment

The developed software was tested at two enterprises operating in different fields: the Levenya Educational and Rehabilitation Center (ERC "Levenya") and the Publishing House of Lviv Polytechnic. During the tests, a series of text prompts describing the desired graphical output were provided as input to the program. Upon completion, expert evaluations were conducted by specialists at the respective organizations to assess the quality of the synthesized tactile graphics. The experimental results demonstrated that the synthesized images accurately matched the desired content and exhibited appropriate quality. Based on expert assessments, it was concluded that the software could be recommended for producing tactile illustrations for educational materials and other publications aimed at people with visual impairments.

The process of producing tactile images synthesized by the program on specialized heat-sensitive capsule paper was also evaluated separately. A "PIAF" tactile graphics printer, commonly used by individuals with visual impairments in educational, professional, and personal contexts, was employed to reproduce the synthesized images. According to the printing results (presented in Figure 3), specialists at the Levenya Educational and Rehabilitation Center determined that the synthesized tactile graphics met the re-

quired qualitative and technical standards, making the software suitable for further use in educational materials and similar publications for people with visual impairments.

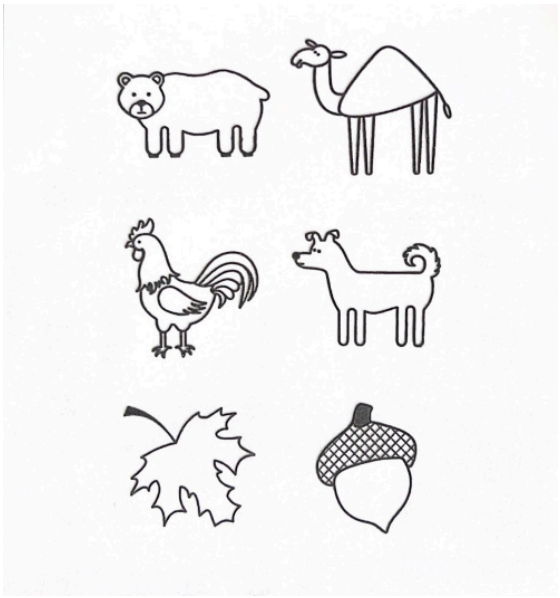


Figure 3. Reproduction of image samples synthesized using the developed software on specialized heat-sensitive capsule paper (scan).

During testing, experts also noted the significantly faster production speed of tactile graphics when using the developed software compared to traditional manual methods. Moreover, the text-prompt-driven synthesis process was praised for its user-friendly approach, as it does not require operators to possess advanced expertise in tactile graphics production. This feature enables the creation of images with diverse content while maintaining ease of use.

Furthermore, experts highlighted the software's advantages over other automated systems for synthesizing tactile graphics. Unlike systems that rely on additional input data (e.g., programs that convert photos into tactile images), the developed software requires only a textual description, significantly simplifying the preparation of materials for production. This convenience, combined with its high efficiency, positions the software as a valuable tool for creating tactile graphics in various domains.

5.4. Quantitative Metrics

The evaluation results are shown in Table 5. The CLIP score value, according to Formula (14), for the obtained model was equal to 23.7. At first glance, the result may seem bad; however, an explanation was found for this. The CLIP model used to calculate the CLIP Score was trained on a diverse range of image samples, including realistic and graphically complex images, as it was designed for general-purpose applications. However, it has been found that the CLIP model—at least in its publicly available versions—is not well-suited for calculating CLIP scores on simple, graphic images such as tactile graphics.

Table 5. Evaluation results of the developed model.

BART	**Cross-Entropy**	**5709**
VQ-VAE	Perplexity	301,662
	MSE (image space)	0.0144
	MSE (latent space)	0.0058
	FID	0.242
Model [1]	CLIP Score	23.7

[1] The text-conditioned tactile graphics generative model.

6. Discussion

The current training dataset consists of relatively simple images, such as animals, plants, and basic objects. However, one limitation of the model lies in its potential difficulty in scaling to more complex images, such as those with intricate details (e.g., architectural blueprints or detailed scientific diagrams). The model's ability to capture fine-grained details is constrained by the size of its latent space and the number of hidden layers in the VQ-VAE architecture. Generating complex tactile graphics may require a more detailed representation, which could introduce inefficiencies or inaccuracies without modifications to the model architecture.

Furthermore, while the model performs well with simpler prompts (e.g., "a cat" or "a tree"), it faces challenges when handling more complex and nuanced prompts (e.g., "a group of children playing soccer with a spotted ball"). As the semantic complexity and length of the prompt increase, the transformer's encoding of textual information becomes more demanding. This can result in difficulties disentangling and representing all elements of a complex scene in tactile graphics form, potentially leading to information loss or oversimplification.

In terms of computational requirements, training the proposed model—which combines the BART Transformer and the VQ-VAE—requires substantial resources. The autoregressive nature of the model and the dual processing of textual and graphical latent spaces make training computationally expensive. This process demands powerful GPUs or TPUs, significant memory capacity, and extended training times, particularly as the dataset size increases. Scaling the model to handle larger datasets or higher-dimensional image outputs poses additional challenges without access to advanced computing infrastructure.

The hyperparameters listed in Table 3 were chosen based on a combination of empirical tuning and best practices from prior studies in related fields. The batch size, in particular, was set to 2 due to the high dimensionality of the inputs and the computational constraints of the hardware used during training. While larger batch sizes are often preferred for their ability to improve gradient estimation stability, our experiments showed that the chosen configuration achieved acceptable performance without overloading available resources.

We acknowledge the potential of more advanced hyperparameter optimization techniques, such as Bayesian optimization, to explore and identify more effective configurations. These methods could enable the model to converge faster or achieve improved performance by systematically searching a broader parameter space. While the current study focused on manual tuning to balance resources and performance, integrating automated optimization tools in future work represents a promising direction for further improving the model's effectiveness.

Nevertheless, while the training process of the proposed model is computationally intensive and time-consuming, due to the high dimensionality of the inputs and the complexity of the machine learning process, the inference phase remains efficient. This is achieved through the compact latent space of the VQ-VAE (8×8 dimension), which minimizes the computational load during decoding. Additionally, the autoregressive BART decoder generates graphical tokens iteratively, but the manageable size of the latent space (64 elements) ensures that the process remains lightweight and fast. The model's focus on producing simplified and clear tactile graphics, rather than photorealistic images, further optimizes inference time, making it practical for real-world applications on resource-constrained devices.

Ethical considerations are paramount in the development of tactile graphics to ensure that the generated images accurately represent the intended information. For visually impaired users, tactile graphics serve as a primary means of interpreting visual content. Any distortion or inaccuracy in the generated graphics could lead to misunderstandings. For instance, oversimplification or omission of critical details in a tactile graphic could provide an incomplete or misleading representation. To address this concern, it is essential to rigorously validate model outputs against established standards for tactile graphics.

Additionally, soliciting feedback from visually impaired users is crucial to ensure that the tactile representations are both accurate and comprehensible.

Currently, effective methods for assessing the quality of tactile graphics include expert evaluations, focus groups, and user testing. However, we believe it is crucial to explore the development of an automated tool for assessing the quality of tactile graphics. Such a tool could, for instance, function as a discriminator within a GAN architecture, providing an objective and scalable approach to evaluation. The development of such a tool is a challenging task, as it must address both the technical specifications and the quality standards, which imply that tactile graphics should be compatible with the following properties:

- Legible: The convexity of forms, protrusions of points, signs, lines, and textures defining highlighted surfaces must be easily recognizable by touch;
- Understood: The reader must clearly grasp the idea conveyed by the author through the illustration;
- Substantial: The illustration must correspond to and enhance the text it accompanies;
- Attractive: The illustration should feel pleasant to the touch and evoke interest. Such images encourage visually impaired readers to engage with them, even if it requires effort;
- Continuous: The reader should seamlessly follow the graphic element without losing their place or searching for the next tactile component;
- Useful: Illustrations must convey meaningful information to the user. Decorative elements that merely enhance aesthetic value should be avoided, as they can interfere with readability.

7. Conclusions

Our proposed method introduces a novel approach to generating tactile graphics directly from textual descriptions, leveraging a Bidirectional and Auto-Regressive Transformer (BART) and a Vector Quantized Variational Auto-Encoder (VQ-VAE). By redefining the latent space architecture to separate textual and graphical embeddings, the model ensures precise semantic mapping from text to tactile graphics. This innovation eliminates the reliance on visual input, distinguishing our work from existing methods like Pic2Tac, which depend on photographs or predefined tactile libraries.

Additionally, our adaptation of VQ-VAE optimizes the generated outputs for tactile accessibility by prioritizing legibility and simplicity over photorealistic detail. The end-to-end text-to-tactile pipeline significantly reduces production time and expands accessibility use cases, enabling the automated generation of educational materials for visually impaired users. These contributions establish our approach as a transformative solution in tactile graphics generation, addressing key limitations of prior methods while demonstrating real-world applicability.

This technology has the potential to bridge the accessibility gap in educational materials, enabling visually impaired individuals to better engage with subjects that rely heavily on visual content, such as science, mathematics, and geography. By providing automated tactile graphics, the technology promotes greater independence in learning and supports participation in inclusive classrooms and professional settings.

A key direction for future research involves expanding the size and diversity of the training dataset. This will enhance the model's generalization capabilities and ensure its stable performance across a broader range of scenarios.

Author Contributions: Conceptualization, V.M. and L.M.; methodology, V.M.; software, Y.D.; validation, V.M. and L.M.; formal analysis, Y.D.; investigation, Y.D.; resources, V.M.; data curation, L.M.; writing—original draft preparation, Y.D.; writing—review and editing, Y.D.; visualization, Y.D.; supervision, V.M.; project administration, L.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in the study are openly available in GitLab at https://gitlab.com/YehorDzhurynskyi/text-to-tactile-image (accessed on 19 December 2024). Restrictions apply to the availability of these data. Data were obtained from APH Tactile Graphics Library and are available at https://imagelibrary.aph.org/portals/aphb/#page/welcome (accessed on 19 December 2024) with the permission of APH Tactile Graphics Library.

Acknowledgments: The authors would like to express their gratitude to the Levenya Educational and Rehabilitation Center (ERC "Levenya") and the Publishing House of Lviv Polytechnic for their invaluable support in testing the developed software. The organizations provided essential facilities and expertise, enabling a comprehensive evaluation of the generated tactile graphics.

Conflicts of Interest: Author Yehor Dzhurynskyi was employed by the company SQUAD Ukraine LLC. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Ackland, P.; Resnikoff, S.; Bourne, R. World blindness and visual impairment: Despite many successes, the problem is growing. *Community Eye Health J.* **2018**, *8*, 71–73. [PubMed]
2. Zebehazy, K.; Wilton, A. Graphic Reading Performance of Students with Visual Impairments and Its Implication for Instruction and Assessment. *J. Vis. Impair. Blind.* **2021**, *115*, 215–227. [CrossRef]
3. Mukhiddinov, M.; Kim, S.-Y. A Systematic Literature Review on the Automatic Creation of Tactile Graphics for the Blind and Visually Impaired. *Processes* **2021**, *9*, 1726. [CrossRef]
4. GBD 2019 Blindness and Vision Impairment Collaborators on behalf of the Vision Loss Expert Group of the Global Burden of Disease Study. Trends in prevalence of blindness and distance and near vision impairment over 30 years: An analysis for the Global Burden of Disease Study. *Lancet Glob. Health* **2021**, *9*, e130–e143. [CrossRef]
5. Mayik, V.; Dudok, T.; Mayik, L.; Lotoshynska, N.; Izonin, I.; Kusmierczyk, J. An Approach Towards Vacuum Forming Process Using PostScript for Making Braille. In *Advances in Computer Science for Engineering and Manufacturing*; Springer International Publishing: Berlin/Heidelberg, Germany, 2022; pp. 38–48. [CrossRef]
6. Way, T.; Barner, K. Towards Automatic Generation of Tactile Graphics. *Rehabil. Eng. Assist. Technol. Soc. N. Am.* **1996**, *96*, 161–163.
7. Way, T.; Barner, K. Automatic visual to tactile translation-Part I: Human factors, access methods, and image manipulation. *IEEE Trans. Rehabil. Eng.* **1997**, *5*, 81–94. [CrossRef] [PubMed]
8. Way, T.; Barner, K. Automatic visual to tactile translation. II. Evaluation of the TACTile image creation system. *IEEE Trans. Rehabil. Eng.* **1997**, *5*, 95–105. [CrossRef] [PubMed]
9. Ferro, T.; Pawluk, D. Automatic image conversion to tactile graphic. In Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility, Bellevue, WA, USA, 23 October 2013.
10. Braille Authority of North America & Canadian Braille Authority: Guidelines and Standards for Tactile Graphics. Available online: https://www.brailleauthority.org/guidelines-and-standards-tactile-graphics (accessed on 18 November 2024).
11. Polish Association of the Blind: Instructions for Creating and Adapting Illustrations and Typhlographic Materials for Blind Students. Available online: https://pzn.org.pl/wp-content/uploads/2016/07/instrukcja_tworzenia_i_adaptowania_ilustracji_i_materialow_tyflograficznych_dla_niewidomych.pdf (accessed on 18 November 2024).
12. Pakėnaitė, K.; Nedelev, P.; Kamperou, E.; Proulx, M.; Hall, P. Communicating Photograph Content Through Tactile Images to People with Visual Impairments. *Front. Comput. Sci.* **2022**, *3*, 787735. [CrossRef]
13. Pakenaite, K.; Kamperou, E.; Proulx, M.J.; Sharma, A.; Hall, P. Pic2Tac: Creating Accessible Tactile Images using Semantic Information from Photographs. In Proceedings of the Eighteenth International Conference on Tangible, Embedded, and Embodied Interaction, Cork, Ireland, 11 February 2024. [CrossRef]
14. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks 2014. *arXiv* **2014**. [CrossRef]
15. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. In Proceedings of the International Conference on Learning Representations, London, UK, 8 July 2014. [CrossRef]
16. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-Resolution Image Synthesis with Latent Diffusion Models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 19 June 2022. [CrossRef]
17. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4 December 2017. [CrossRef]
18. Yingchen, Y.; Fangneng, Z.; Rongliang, W.; Jianxiong, P.; Kaiwen, C.; Shijian, L.; Feiying, M.; Xuansong, X.; Chunyan, M. Diverse Image Inpainting with Bidirectional and Autoregressive Transformers. In Proceedings of the 29th ACM International Conference on Multimedia, New York, NY, USA, 20 October 2021. [CrossRef]
19. Van den Oord, A.; Vinyals, O.; Kavukcuoglu, K. Neural Discrete Representation Learning. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4 December 2017. [CrossRef]

20. Zouhar, V.; Meister, C.; Gastaldi, J.; Du, L.; Vieira, T.; Sachan, M.; Cotterell, R. A Formal Perspective on Byte-Pair Encoding. In *Findings of the Association for Computational Linguistics: ACL 2023*, 1st ed.; Rogers, A., Boyd-Graber, J., Okazaki, N., Eds.; Association for Computational Linguistics: Toronto, ON, Canada, 2023; pp. 598–614. [CrossRef]
21. LanguageTool API NLP UK. Available online: https://github.com/brown-uk/nlp_uk (accessed on 18 November 2024).
22. American Printing House: Tactile Graphic Image Library. Available online: https://imagelibrary.aph.org/portals/aphb/#page/welcome (accessed on 18 November 2024).
23. Hessel, J.; Holtzman, A.; Forbes, M.; Le Bras, R.; Choi, Y. CLIPScore: A Reference-free Evaluation Metric for Image Captioning. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Punta Cana, Dominican Republic, 7 November 2021. [CrossRef]