RETA3D: Real-Time Animatable 3D Gaussian Head Generation

Shu-Yu Chen[1,2], Chunshuo Qiu[1,2], Feng-Lin Liu[1,2], Yanpei Cao[3], Hongbo Fu[4], Lin Gao[1,5]

[1]Institute of Computing Technology, Chinese Academy of Sciences [2]University of Chinese Academy of Sciences
[3]Vast [4]Hong Kong University of Science and Technology [5]Jinan Zhongke Ubiquitous-Intelligent institute of computing technology

Abstract

3D avatar GANs (generative adversarial networks) learn 3D priors from extensive collections of 2D portrait images. However, existing 3D avatar GANs either struggle with real-time performance or lack 3D consistency. To address these issues, we present RETA3D, a novel 3D GAN framework leveraging the efficiency of 3D Gaussian Splatting (3DGS). Our core contribution is a consecutive mesh-binding 3D Gaussian representation that tightly integrates 3D Gaussians with a FLAME mesh template via a novel local coordinate system defined by surface normals and head pose to ensure consistent animation. We also introduce a dynamic texture generation framework that separates static and dynamic texture components, significantly improving reenactment speed. This framework generates a static texture once and efficiently computes dynamic texture updates per-frame using a compact neural network conditioned on FLAME parameters.

CCS Concepts

• ***Computing methodologies*** → *Animation;* ***3D imaging;*** • ***Theory of computation*** → *Adversarial learning;*

1. Introduction

Real-time animatable 3D avatar synthesis plays an important role in applications such as VR live streaming, video conferencing, and telepresence. An ideal avatar generator must produce high-fidelity, photorealistic facial renderings, exhibit realistic dynamic expressions and movements, and operate at interactive frame rates. Achieving this combination of quality, realism, and speed presents significant challenges.

Previous 2D works [DYC*20, GZL*24, CJF*24] utilize generation models conditioned by 3D Morphable Face Models or latent keypoints to animate avatars. However, due to the lack of a 3D representation foundation, these techniques often cause noticeable artifacts during significant movements and large viewpoint changes. To achieve better geometric consistency, many methods [SWW*23, TZY*23, MZQ*23] train effective 3D avatar generators under 2D image supervision and drive them using mesh-based or parametric models. While these approaches achieve better 3D consistency, they often struggle to achieve real-time performance due to the high computational cost of volumetric rendering.

3D Gaussian Splatting (3DGS) [KKLD23, WYZ*24] enhances modeling and rendering speeds by using a Gaussian ellipsoid representation and rasterization. Although the unstructured representation presents challenges for generation, several studies [BBM*24, KGT*24, HH24, JLL*24] have successfully addressed these issues, thereby advancing real-time 3D avatar generation. However, these methods primarily focus on static avatar generation and lack the ability to create dynamic avatars driven by expression or pose parameters, limiting their applicability for animation. Although GaussianAvatars [QKS*24] performs well in dynamic reconstruction, this method require multi-view data for each individual.

We propose *RETA3D*, a novel 3D GAN framework for learning real-time animatable, 3D-consistent, photorealistic facial avatars from 2D images. Our key design is a mesh-binding Gaussian representation with a consecutive-local coordinate system for accurate motion, coupled with a compact dynamic texture generation module for improved animation quality and speed. Our method achieves significantly faster reenactment speeds compared to state-of-the-art 3D animatable head GANs while maintaining comparable generation quality. As a strong 3D prior, our method facilitates downstream applications such as one-shot reconstruction and real-time animation of 3D facial avatars.

2. Method

As illustrated in Fig. 1, our framework consists of a consecutive mesh-binding 3DGS representation (Sec. 2.1) , a UV feature map generator (Sec. 2.2), and a dynamic texture module (Sec. 2.3). The training strategy is described in Sec. 2.4.

2.1. Consecutive-Local Mesh-Binding 3D Gaussians

2.1.1. Mesh-Binding 3D Gaussian Splatting

Inspired by [KGT*24, XGGZ24], we parameterize 3D Gaussians onto a template head mesh (FLAME [LBB*17]) within the UV space, binding them to mesh triangles to inherit the coarse deformation of the facial template. Specially, our generator (detailed in Sec. 2.2 and 2.3) synthesizes a 2D UV feature map $\mathcal{F}^{uv} \in R^{256\times256\times14}$. Each pixel i is translated into a Gaussian by feature splitting, as $f_{u,v} = \{o_i, s_i, q_i, c_i, \sigma_i\}$, where $o \in R^3$ denotes the center offset,

DOI: 10.2312/egs.20261003

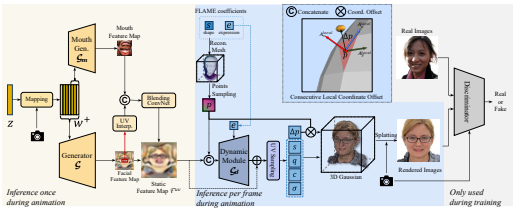


Figure 1: ***Framework of our method.*** *Our approach generates a static feature map using a StyleGAN-based generator $\mathcal{G}$ with a dedicated mouth patch generator $\mathcal{G}_m$. The above process is computed only once during animation. Then a lightweight dynamic module $\mathcal{G}_d$ predicts residual dynamic textures as residual.*

$s \in R^3$ the sacle, $q \in R^4$ the rotation parameterized as quaternion , $c \in R^3$ the color, and $\sigma \in R$ the opacity. To determine its 3D position, we conduct UV sampling to find the corresponding face triangle with vertices (p_i^0, p_i^1, p_i^2). Then, the initial position is obtained using barycentric coordinates (u,v): v):$\mathrm{p}_i = up_i^0 + vp_i^1 + (1-u-v)p_i^2$. To capture personalized geometry, each Gaussian is further displaced from p_i by a small offset. Specially, the offset value $o_i \in R^3$ is queried from the UV feature map. We assume this offset is defined in a local coordinate system that varies per triangle. Thus, a transformation matrix $C_i = [X_i, Y_i, Z_i] \in R^{3\times3}$ is required to convert o_i from local to global coordinates.

The final G aussian p osition i s t hen c omputed a s: $\mu_i = p_i + \gamma^o \tanh(o_i) \cdot C_i$, where tanh acts as an activation function, and $\gamma^o = 0.25$ controls the offset range.

2.1.2. Consecutive Local Coordinate

We propose a consecutive local coordinate system for stable training and effective mesh-guided Gaussian animation, which is an special case in tangent space. For each Gaussian g_i bound to a triangle face (p_i^0, p_i^1, p_i^2), we calculate its normal n_i using barycentric coordinates (u,v) and precomputed vertex normal (n_i^0, n_i^1, n_i^2):

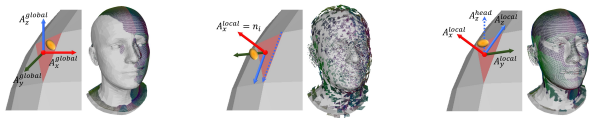
$$n_i = un_i^0 + vn_i^1 + (1-u-v)n_i^2. \tag{1}$$

This normal forms the first axis $A_x^{local} = n_i$ of the local frame. Given the head upward axis A_z^{head} from the FLAME template, we construct the remaining orthogonal axes as $\mathrm{A}_y^{local} = A_z^{head} \times A_x^{local}, A_z^{local} = A_x^{local} \times A_y^{local}$. The resulting local coordinates are continuous across faces and adjacent triangles. By combining mesh normals and the global head orientation, they reliably propagate global head motion and mesh deformations into Gaussian dynamics during pose and expression changes. Degeneracy when n_i aligns with A_z^{head} is negligible for typical human head geometries.

2.2. UV Generation with Mouth Enhancement

As previously discussed, we parameterize 3D Gaussians onto the template head mesh using the UV layout. And introduce a dedicated mouth patch generator $\mathcal{G}_m$ to enhance mouth fidelity.

We first map the latent code $z \in R^{512}$ into w^+ using a mapping network. As analyzed in previous works [KLA*20, CLC*22], the earlier channels of w^+ control global facial structure, while the later channels influence fine-grained details. Therefore, we provide the complete w^+ to the main generator $\mathcal{G}$ and select the later channels as input for the mouth patch generator $\mathcal{G}_m$. The two generators produce the UV feature map $\mathcal{F}^{uv} \in R^{256\times256\times14}$ for the entire face, and the local detailed feature map $\mathcal{F}^{mouth} \in R^{64\times64\times14}$ for the mouth, respectively. To ensure consistency, we extract the corresponding mouth region from the full-face feature map, upsample it to match the mouth patch resolution, and concatenate the two feature maps. A lightweight 2D convolutional network, referred to as the Blending ConvNet, is then applied to merge these features. Finally, we sample approximately 256^2 Gaussians from the UV feature map and additional 64^2 Gaussians from the blended mouth feature map to construct the Gaussian Head.



(a) Global Coordinate (b) Naive Local Coord. (c) Ours

Figure 2: *Different definitions of local coordinate system.*

We set $o_i = (1,1,0)$ for UV feature map pixels and visualize Gaussians to evaluate tightness and consecutiveness. (a) Global coordinates [XGGZ24] lose template's geometry and mesh deformations. (b) Naive local coordinates [QKS*24] cause discontinuity/fragmentation. (c) Our consecutive local coordinate system ensures motion consistency for generation.

2.3. Dynamic Texture Generation

With an animatable FLAME template, the mesh-binding Gaussian head models the head pose changes and deformation caused by expressions. However, to address some dynamic details and subtle non-rigid deformations beyond FLAME's expressive capability, we introduce a dynamic generation module $\mathcal{G}_d$ to model these details. This module generates residual features $\Delta\mathcal{F}^{uv}$ based on static Gaussian features $\mathcal{F}^{uv}$, which is added to obtain dynamic features $\mathcal{F}_d^{uv}$. To better capture head information, we back-project the 3D coordinates of Gaussians to the UV plane, and concatenate them with static Gaussian features as input to the dynamic generation module. The module uses FLAME's expression parameters e as control conditions and employs a multi-layer convolutional network to obtain residual features, which are then added to the static Gaussian features to produce the final dynamic features: $\Delta\mathcal{F}^{uv} = \mathcal{G}_d(\{f_{u,v}, xyz\}, e), \mathcal{F}_d^{uv} = \mathcal{F}^{uv} + \Delta\mathcal{F}^{uv}$. During inference, the full generator is executed on the first frame to obtain static textures, while subsequent frames require only the generation of dynamic textures. Thanks to the lightweight design of dynamic module and the efficient rendering of 3D Gaussian splatting, our method enables fast animation and even achieves real-time performance.

2.4. Training Pipeline

We generate facial and mouth UV feature maps at resolutions of 256^2 and 64^2, respectively, and sample Gaussians at the same resolution, resulting in approximately $68k$ Gaussians (excluding pixels originally masked as invalid regions in the UV map). To regularize the predicted Gaussian attributes, we apply the constraints: $\mathrm{L}_{\mathrm{reg}}^{\mathrm{scale}} = \frac{1}{N}\Sigma_{i=1}^{N}||s_i - \gamma^s||_2, \mathcal{L}_{\mathrm{reg}}^{\mathrm{offset}} = \frac{1}{N}\Sigma_{i=1}^{N}||o_i||_2$, where N is the number of Gaussians and $\gamma^s = e^{-5.5}$ is the target scale, corresponding to the average distance between adjacent sampled Gaussians.

To prevent the dynamic module from significantly altering the static feature map and causing severe artifacts, we introduce a regularization term $\mathcal{L}_{\mathrm{reg}}^{\mathcal{G}_d}$ and an identity consistency loss $\mathcal{L}_{\mathrm{ID}}$ specifically for the dynamic module. $\mathrm{L}_{\mathrm{reg}}^{\mathcal{G}_d} = \Sigma_{w_i \in \mathcal{G}_d}||w_i||_2, \mathcal{L}_{\mathrm{ID}} =$

Figure 3: *Qualitative comparisons with Next3D, 3DFaceShop and OTAvatar. Random samples of FFHQ-trained 3D animatable GANs driven by the same video clip (Top). Our method achieves better 3D consistency (e.g., eyeglass details) and faster animation speed.*

$||VGG(I_d) - VGG(I_s)||_1$, where w_i are the parameters of the dynamic module $\mathcal{G}_d$. I_d and I_s are the images generated with and without the dynamic module, respectively.

Finally, we use a standard single discriminator $\mathcal{D}$ in [CLC*22] to supervise the plausibility of the rendered Gaussian head, $\mathrm{L}_{\mathrm{adv}} = \mathrm{softplus}(-\mathcal{D}(I_{\mathcal{G}}|\pi))$. The final optimization target for the generator is as follows:

$$\mathcal{L}_G = \mathcal{L}_{\mathrm{adv}} + \lambda_o \mathcal{L}_{\mathrm{reg}}^{\mathrm{offset}} + \lambda_s \mathcal{L}_{\mathrm{reg}}^{\mathrm{scale}} + \lambda_r \mathcal{L}_{\mathrm{reg}}^{\mathcal{G}_d} + \lambda_{ID} \mathcal{L}_{\mathrm{ID}}, \quad (2)$$

where π is the camera. The regularization weights are $\lambda_o = 0.1$, $\lambda_s = 0.1$, $\lambda_r = 0.001$, and $\lambda_{ID} = 0.1$. The discriminator is trained with R1 gradient regularization, using the weight of 1. We adopt the training hyperparameters (learning rates of generator and discriminator, batch size, etc.) from StyleGAN2. With a batch size of 32, we totally train our model from scratch for $25M$ images following EG3D on the FFHQ dataset (70k images), which takes 6 days on 4 RTX A6000 GPUs(48G VRAM).

3. Experiments

3.1. Comparisons

We compare our method against 3DFaceShop [TZY*23], Next3D [SWW*23], and OTAvatar [MZQ*23].

Qualitative Comparison. Fig. 3 presents a qualitative comparison of reenactment results. It can be seen that our method produces more consistent results and achieves higher rendering speed. While 3DFaceShop achieves high image quality, it lacks control over expression details like blinking. Next3D shows good motion accuracy but suffers from inconsistency issues, such as the appearance of glasses, which is more noticeable in the supplementary video. OTAvatar falls short in both image quality and motion accuracy. Although OTAvatar improves driving speed compared to other NeRF-based methods, it still fails to achieve real-time performance. Overall, our approach maintains advantages in image quality, motion accuracy, and detail expression capability while enabling real-time reenactment.

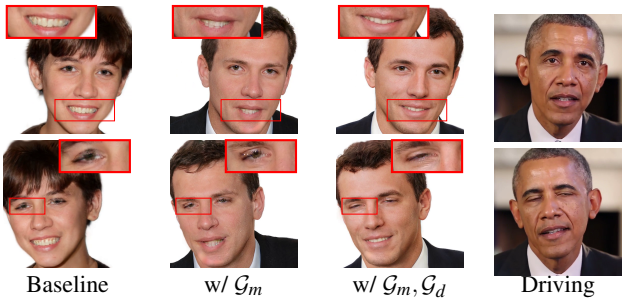


Figure 4: *Qualitative comparison for the ablation study The component $\mathcal{G}_m$ enhances the quality of the generated mouth, while $\mathcal{G}_d$ improves dynamic clarity and expression details.*

Quantitative Evaluation. We evaluate image quality with Fréchet Inception Distance(FID) [HRU*17] on the entire FFHQ dataset with 50,000 generated images generated from randomly sampled latent codes, camera poses, and FLAME parameters. To assess animation accuracy, we follow the methods outlined in [TZY*23, SWW*23]. We calculate three key metrics: the Average Expression Distance (AED), the Average Pose Distance (APD), and Identity Consistency (ID). For each method, we sample 500 identities, animate each with 20 random FLAME expression/pose parameter sets. Then compute the average distance between driving FLAME parameters and reconstructed parameters of these 10,000 generated images. For identity consistency, we generate 500 image pairs (500 identities × 2 random parameter sets) and calculate the average consistency metric via the pre-trained ArcFace model [DGNZ19]. As demonstrated in Tab. 1(a), our method surpasses existing methods across all metrics, except the comparable FID score to Next3D. The 2D render quality outperforms 3DFaceShop and OTAvatar. While Next3D achieves the lowest FID via a super-resolution module (at the cost of 3D consistency), our method maintains similar FID with superior expression/pose accuracy and the highest identity consistency among all compared methods.

Runtime comparison. We compare the inference speed across three phases: Gen. (mapping identity latent codes to 3D representations), Rend. (synthesizing images from 3D representations, including super-resolution for existing methods), and Dri. (combining both phases to generate videos from FLAME sequences and identity codes). We randomly sample 5 identities and 240 FLAME parameter sequences for animation. Tab. 1(b) reports average FPS, showing our method achieves real-time speeds with significant advantages across all phases. The evaluations are conducted on a single RTX 2080 Ti GPU with a batch size of 1.

3.2. Ablation Study

Consecutive Local Coordinate. We compare models with different mesh-binding coordinates, including global offset and naive local coordinates, trained on the FFHQ dataset at a resolution of 256 for 4 million images. Fig. 4 shows that the detailed expressions, such as blinking, are more accurately depicted by the dynamic module. As shown in Tab. 1(c), our proposed Consecutive Local Coordinate, due to its more regular distribution, enhances smoothness in the

Table 1: *Quantitative evaluation including overall performance, runtime, and ablation study.*

$FFHQ@512^2$	FID	AED	APD	ID
3DFaceShop	23.7	0.12	0.036	0.87
Next3D	**3.9**	0.12	0.031	0.86
OTAvatar	38.8	0.17	0.035	0.89
Ours	4.3	**0.07**	**0.018**	**0.97**

(a) Overall quality

Method	3DFaceShop	Next3D	OTAvatar	Ours
Gen.	6.1	14.3	49.3	**203.0**
Rend.	18.1	59.6	37.3	**129.9**
Dri.	4.6	13.8	21.3	**79.2**

(b) Runtime (FPS)

	Res.	FID	AED	FPS
Global offset		9.8	0.15	–
Naive local	256^2	11.9	0.10	–
Ours		**8.1**	**0.08**	–
Baseline		6.8	0.10	31.5
+ $\mathcal{G}_m$	512^2	4.8	0.09	29.8
+ $\mathcal{G}_m + \mathcal{G}_d$		**4.3**	**0.07**	**79.2**

(c) Ablation

generation of UV feature maps, improving the overall quality while maintaining high expression accuracy.

Generation Modules. We conduct an ablation study on our two main modules, mouth patch generator and dynamic module. We add the mouth patch generator $\mathcal{G}_m$ and dynamic module $\mathcal{G}_d$ to the baseline with a single UV feature generator only and train the models on $FFHQ@512$. As reported in Tab. 1(c), the mouth patch generator enhances the visual quality of the generated heads, while the dynamic module further improves dynamic details and expression accuracy. By separating the generation of dynamic textures from whole textures through dynamic modules, the driving speed of our method has been significantly improved.

4. Conclusion

In this work, we present RETA3D, a real-time animatable 3D head generation framework built on 3D Gaussian Splatting. RETA3D adopts a consecutive local mesh-binding Gaussian representation to tightly integrate Gaussians with the FLAME mesh, ensuring consistent animation deformations, while our dynamic texture generation framework decouples static and dynamic components to accelerate animation speed significantly. Our approach enables real-time animation with detailed dynamics, it has limitations including uncontrollable gaze direction, lack of physical dynamics for long hair and suboptimal back-view rendering.

Acknowledgements

This work was supported by National Natural Science Foundation of China (No. 62472407, 62322210, 62561160115), Key Research and Development Program of Hunan Province of China (No. 2025WK2006), Innovation Funding of ICT, CAS (No. E461020), and HNXJ Philanthropy Foundation (No. KY24010, KY25009).

References

[BBM*24] BARTHEL F., BECKMANN A., MORGENSTERN W., HILSMANN A., EISERT P.: Gaussian splatting decoder for 3d-aware generative adversarial networks. *arXiv preprint arXiv:2404.10625* (2024). 1

[CJF*24] CHEN S.-Y., JIANG Y.-R., FU H., HAN X., LIU Z., LI R., GAO L.: Deepfacereshaping: Interactive deep face reshaping via landmark manipulation. *Computational Visual Media 10*, 5 (2024), 949–963. doi:10.1007/s41095-023-0373-1. 1

[CLC*22] CHAN E. R., LIN C. Z., CHAN M. A., NAGANO K., PAN B., MELLO S. D., GALLO O., GUIBAS L. J., TREMBLAY J., KHAMIS S., KARRAS T., WETZSTEIN G.: Efficient geometry-aware 3D generative adversarial networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR* (2022), IEEE, pp. 16102–16112. 2, 3

[DGNZ19] DENG J., GUO J., NIANNAN X., ZAFEIRIOU S.: ArcFace: Additive angular margin loss for deep face recognition. In *CVPR* (2019). 3

[DYC*20] DENG Y., YANG J., CHEN D., WEN F., TONG X.: Disentangled and controllable face image generation via 3d imitative-contrastive learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 5154–5163. 1

[GZL*24] GUO J., ZHANG D., LIU X., ZHONG Z., ZHANG Y., WAN P., ZHANG D.: Liveportrait: Efficient portrait animation with stitching and retargeting control. *arXiv preprint arXiv:2407.03168* (2024). 1

[HH24] HYUN S., HEO J.-P.: Adversarial generation of hierarchical gaussians for 3d generative model. *arXiv preprint arXiv:2406.02968* (2024). 1

[HRU*17] HEUSEL M., RAMSAUER H., UNTERTHINER T., NESSLER B., HOCHREITER S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems 30* (2017). 3

[JLL*24] JIANG Y., LIAO Q., LI X., MA L., ZHANG Q., ZHANG C., LU Z., SHAN Y.: Uv gaussians: Joint learning of mesh deformation and gaussian textures for human avatar modeling. *arXiv preprint arXiv:2403.11589* (2024). 1

[KGT*24] KIRSCHSTEIN T., GIEBENHAIN S., TANG J., GEORGOPOULOS M., NIESSNER M.: Gghead: Fast and generalizable 3d gaussian heads. In *SIGGRAPH Asia 2024 Conference Papers* (2024), pp. 1–11. 1

[KKLD23] KERBL B., KOPANAS G., LEIMKÜHLER T., DRETTAKIS G.: 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph. 42*, 4 (2023), 139–1. 1

[KLA*20] KARRAS T., LAINE S., AITTALA M., HELLSTEN J., LEHTINEN J., AILA T.: Analyzing and improving the image quality of StyleGAN. In *Proc. CVPR* (2020). 2

[LBB*17] LI T., BOLKART T., BLACK M. J., LI H., ROMERO J.: Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia) 36*, 6 (2017), 194:1–194:17. 1

[MZQ*23] MA Z., ZHU X., QI G.-J., LEI Z., ZHANG L.: OTAvatar: One-shot talking face avatar with controllable tri-plane rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), pp. 16901–16910. 1, 3

[QKS*24] QIAN S., KIRSCHSTEIN T., SCHONEVELD L., DAVOLI D., GIEBENHAIN S., NIESSNER M.: Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), pp. 20299–20309. 1, 2

[SWW*23] SUN J., WANG X., WANG L., LI X., ZHANG Y., ZHANG H., LIU Y.: Next3D: Generative neural texture rasterization for 3D-aware head avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), pp. 20991–21002. 1, 3

[TZY*23] TANG J., ZHANG B., YANG B., ZHANG T., CHEN D., MA L., WEN F.: 3DFaceShop: Explicitly controllable 3D-aware portrait generation. *IEEE Transactions on Visualization and Computer Graphics* (2023). 1, 3

[WYZ*24] WU T., YUAN Y.-J., ZHANG L.-X., YANG J., CAO Y.-P., YAN L.-Q., GAO L.: Recent advances in 3d gaussian splatting. *Computational Visual Media 10*, 4 (2024), 613–642. doi:10.1007/s41095-024-0436-y. 1

[XGGZ24] XIANG J., GAO X., GUO Y., ZHANG J.: Flashavatar: High-fidelity head avatar with efficient gaussian embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), pp. 1802–1812. 1, 2