🔍 ALTERNATIVAS DE VISION API - GUIA COMPLETO

📋 Visão Geral

Este documento apresenta **todas as alternativas** de Vision API descobertas nos repositórios **xinnan-tech/xiaozhi-esp32-server** e ecossistema relacionado.

🏆 ALTERNATIVAS PRINCIPAIS

1️⃣ **Aliyun Bailian (阿里百炼)** - ⭐ RECOMENDADO PARA PRODUÇÃO

Por que é o melhor:

- ☑️ **Mais rápido**: ~2.5s mais rápido que Zhipu
- ☑️ **Streaming nativo**: Respostas progressivas
- ☑️ **Créditos grátis**: Bom para começar
- ☑️ **Estável**: Infraestrutura da Alibaba Cloud
- 👍 **Recomendado por xinnan-tech**

Modelo: `qwen2.5-vl-3b-instructh`
Endpoint: `https://dashscope.aliyuncs.com/compatible-mode/v1`
Interface: OpenAI-compatible

Configuração:

```
{
  "selected_module": {
    "VLLM": "qwen_vl"
  },
  "VLLM": {
    "qwen_vl": {
      "type": "openai",
      "api_key": "SEU_TOKEN_ALIYUN",
      "model": "qwen2.5-vl-3b-instructh",
      "api_url": "https://dashscope.aliyuncs.com/compatible-mode/v1",
      "base_url": "https://dashscope.aliyuncs.com/compatible-mode/v1",
      "temperature": 0.7,
      "max_tokens": 2048,
      "timeout": 30.0
    }
  }
}
```

Como obter token:

1. 🌐 Acesse: https://bailian.console.aliyun.com/

2. 📝 Crie conta Alibaba Cloud (pode usar email internacional)
3. 🔓 Ative serviço Bailian (百炼)
4. 🔑 Gere API Key: https://dashscope.console.aliyun.com/apiKey
5. 📋 Copie a API Key

Custos (aproximados):

- Texto: ~0.0012 yuan / 1K tokens
- Imagem: ~0.008 yuan / imagem
- **~1000 análises = ~8 yuan (~R$ 5.60)**
- 💰 Créditos grátis para novos usuários

2️⃣ Zhipu AI (智谱) - ☑️ BOA OPÇÃO PARA INICIANTES

Por que usar:

- ☑️ Fácil de configurar
- ☑️ Documentação em chinês clara
- ☑️ Créditos grátis generosos
- ☑️ **Modelo atualizado**: glm-4v-flash (mais rápido)

Modelo: `glm-4v-flash` (não glm-4v-vision!)
Endpoint: `https://open.bigmodel.cn/api/paas/v4/chat/completions`
Interface: OpenAI-compatible

Configuração:

```
{
  "selected_module": {
    "VLLM": "zhipu"
  },
  "VLLM": {
    "zhipu": {
      "type": "zhipu",
      "api_key": "SEU_TOKEN_ZHIPU",
      "model": "glm-4v-flash",
      "api_url": "https://open.bigmodel.cn/api/paas/v4/chat/completions",
      "temperature": 0.7,
      "max_tokens": 2048,
      "timeout": 30.0
    }
  }
}
```

Como obter token:

1. 🌐 Acesse: https://open.bigmodel.cn/
2. 📝 Crie conta (email ou WeChat)

3. 🔑 Gere API Key: https://open.bigmodel.cn/usercenter/apikeys
4. 📋 Copie a chave

Custos:

- Texto: ~0.001 yuan / 1K tokens
- Imagem: ~0.01 yuan / imagem
- **~100 análises = ~1 yuan (~R$ 0.70)**

⚠️ **Diferença importante**:

- ❌ `glm-4v-vision`: Modelo antigo (mais lento)
- ☑ `glm-4v-flash`: Modelo novo (~2.5s mais rápido) ← USE ESTE!

3️⃣ **OpenAI GPT-4 Vision** - 💰 MELHOR QUALIDADE (CARO)

Por que usar:

- ☑ Melhor qualidade de análise
- ☑ Documentação completa em inglês
- ☑ API estável e confiável
- ❌ **Mais caro** que alternativas chinesas

Modelo: `gpt-4o` ou `gpt-4-turbo`
Endpoint: `https://api.openai.com/v1/chat/completions`

Configuração:

```
{
  "selected_module": {
    "VLLM": "openai_vision"
  },
  "VLLM": {
    "openai_vision": {
      "type": "openai",
      "api_key": "SEU_TOKEN_OPENAI",
      "model": "gpt-4o",
      "api_url": "https://api.openai.com/v1/chat/completions",
      "base_url": "https://api.openai.com/v1",
      "temperature": 0.7,
      "max_tokens": 2048,
      "timeout": 30.0
    }
  }
}
```

Como obter token:

1. 🌐 Acesse: https://platform.openai.com/

2. 📝 Crie conta OpenAI
3. 💳 Adicione créditos (mínimo ~$5 USD)
4. 🔑 Gere API Key: https://platform.openai.com/api-keys
5. 📋 Copie a chave (começa com `sk-`)

Custos (GPT-4o):

- Input: $5.00 / 1M tokens
- Output: $15.00 / 1M tokens
- Imagem: ~$0.01275 / imagem (detalhada)
- **~100 análises = ~$1.50 USD (~R$ 7.50)**

4️⃣ **Google Gemini Vision** - 🆓 OPÇÃO GRATUITA GENEROSA

Por que usar:

- ☑️ **Créditos grátis muito generosos**
- ☑️ Boa qualidade de análise
- ☑️ Multimodal nativo
- ☑️ Fácil integração

Modelo: `gemini-1.5-flash` ou `gemini-1.5-pro`
Endpoint: `https://generativelanguage.googleapis.com/v1beta`

Configuração:

```
{
  "selected_module": {
    "VLLM": "gemini_vision"
  },
  "VLLM": {
    "gemini_vision": {
      "type": "openai",
      "api_key": "SEU_TOKEN_GEMINI",
      "model": "gemini-1.5-flash",
      "api_url": "https://generativelanguage.googleapis.com/v1beta/openai",
      "base_url": "https://generativelanguage.googleapis.com/v1beta/openai",
      "temperature": 0.7,
      "max_tokens": 2048,
      "timeout": 30.0
    }
  }
}
```

Como obter token:

1. 🌐 Acesse: https://aistudio.google.com/app/apikey
2. 📝 Login com conta Google

3. 🔑 Clique "Create API Key"
4. 📋 Copie a chave

Custos:

- **Gemini 1.5 Flash**: GRÁTIS até 15 RPM (requests/min)
- **Gemini 1.5 Pro**: GRÁTIS até 2 RPM
- Acima dos limites: ~$0.075 / 1M tokens
- **Excelente para testes e uso pessoal!**

5️⃣ **Anthropic Claude Vision** - 🔓 QUALIDADE PREMIUM

Por que usar:

- ☑️ Excelente qualidade de análise
- ☑️ Bom com contextos longos
- ☑️ Ética e segurança priorizadas
- ❌ Sem créditos grátis

Modelo: `claude-3-sonnet-20240229` ou `claude-3-opus`
Endpoint: `https://api.anthropic.com/v1/messages`

Configuração (requer adaptação):

```
{
  "selected_module": {
    "VLLM": "anthropic_vision"
  },
  "VLLM": {
    "anthropic_vision": {
      "type": "anthropic",
      "api_key": "SEU_TOKEN_ANTHROPIC",
      "model": "claude-3-sonnet-20240229",
      "api_url": "https://api.anthropic.com/v1/messages",
      "temperature": 0.7,
      "max_tokens": 2048,
      "timeout": 30.0
    }
  }
}
```

Como obter token:

1. 🌐 Acesse: https://console.anthropic.com/
2. 📝 Crie conta
3. 💳 Adicione créditos (mínimo $5 USD)
4. 🔑 Gere API Key
5. 📋 Copie a chave

Custos (Claude 3 Sonnet):

- Input: $3.00 / 1M tokens
- Output: $15.00 / 1M tokens
- **~100 análises = ~$0.50 USD (~R$ 2.50)**

---

6 **Deepseek VL** - 💎 BOA OPÇÃO CHINESA

Por que usar:

- ☑ Modelo open source chinês
- ☑ Preço competitivo
- ☑ OpenAI-compatible
- ⚠ Documentação principalmente em chinês

Modelo: `deepseek-vl`
Endpoint: `https://api.deepseek.com/v1`

Configuração:

```
{
  "selected_module": {
    "VLLM": "deepseek_vision"
  },
  "VLLM": {
    "deepseek_vision": {
      "type": "openai",
      "api_key": "SEU_TOKEN_DEEPSEEK",
      "model": "deepseek-vl",
      "api_url": "https://api.deepseek.com/v1/chat/completions",
      "base_url": "https://api.deepseek.com/v1",
      "temperature": 0.7,
      "max_tokens": 2048,
      "timeout": 30.0
    }
  }
}
```

Como obter token:

1. 🌐 Acesse: https://platform.deepseek.com/
2. 📝 Crie conta
3. 🔑 Gere API Key
4. 📋 Copie a chave

Custos:

- Muito competitivo, similar a Zhipu
- **~100 análises = ~1 yuan (~R$ 0.70)**

TABELA COMPARATIVA

Provider	Modelo	Velocidade	Preço (100 imgs)	Créditos Grátis	Streaming	Recomendação
Aliyun Bailian	qwen2.5-vl-3b-instructh	⚡⚡⚡⚡⚡	~R$ 5.60	☑ Sim	☑ Sim	👍 **Produção**
Zhipu AI	glm-4v-flash	⚡⚡⚡⚡	~R$ 0.70	☑ Generoso	⚠ Não	👍 **Iniciantes**
Google Gemini	gemini-1.5-flash	⚡⚡⚡⚡	🆓 GRÁTIS	☑☑☑ Muito	☑ Sim	👍 **Testes**
OpenAI	gpt-4o	⚡⚡⚡	~R$ 7.50	❌ Não	☑ Sim	💰 **Qualidade**
Anthropic	claude-3-sonnet	⚡⚡⚡	~R$ 2.50	❌ Não	☑ Sim	🔓 **Premium**
Deepseek	deepseek-vl	⚡⚡⚡⚡	~R$ 0.70	⚠ Pouco	⚠ Não	💎 **Alternativa**

QUAL ESCOLHER?

Para **Testes e Aprendizado**:

1. **Google Gemini** 🆓 (grátis e generoso)
2. **Zhipu AI** 💰 (barato e fácil)

Para **Uso Pessoal**:

1. **Aliyun Bailian** ⚡ (rápido e barato)
2. **Zhipu AI** 💰 (simples e funcional)

Para **Produção/Comercial**:

1. **Aliyun Bailian** ⚡ (melhor custo-benefício + streaming)
2. **OpenAI GPT-4o** 💎 (melhor qualidade)

Para **Máxima Qualidade**:

1. **OpenAI GPT-4o** 💎
2. **Anthropic Claude** 🔓

COMO IMPLEMENTAR NOVA ALTERNATIVA

Passo 1: Adicionar Configuração

Abra config/config.json e adicione:

```
{
  "selected_module": {
    "VLLM": "NOME_DO_PROVIDER"
  },
  "VLLM": {
    "NOME_DO_PROVIDER": {
      "type": "openai",
      "api_key": "SEU_TOKEN",
      "model": "nome-do-modelo",
      "api_url": "https://endpoint.com/v1",
      "temperature": 0.7,
      "max_tokens": 2048,
      "timeout": 30.0
    }
  }
}
```

Passo 2: Testar

```
python verify_vision_api.py
```

Passo 3: Executar

```
python src/mcp/tools/providers/vllm_provider.py
```

📚 DOCUMENTAÇÃO DOS PROVIDERS

Aliyun Bailian

- 📖 Docs: https://help.aliyun.com/zh/model-studio/
- 🔑 Console: https://bailian.console.aliyun.com/
- 💰 Preços: https://help.aliyun.com/zh/model-studio/developer-reference/text-generation-billing

Zhipu AI

- 📖 Docs: https://open.bigmodel.cn/dev/api
- 🔑 Console: https://open.bigmodel.cn/usercenter/apikeys
- 💰 Preços: https://open.bigmodel.cn/pricing

Google Gemini

- 📖 Docs: https://ai.google.dev/docs
- 🔑 Console: https://aistudio.google.com/app/apikey
- 💰 Preços: https://ai.google.dev/pricing

OpenAI

- 📖 Docs: https://platform.openai.com/docs/guides/vision
- 🔑 Console: https://platform.openai.com/api-keys
- 💰 Preços: https://openai.com/api/pricing/

Anthropic

- 📖 Docs: https://docs.anthropic.com/claude/docs/vision
- 🔑 Console: https://console.anthropic.com/
- 💰 Preços: https://www.anthropic.com/pricing

Deepseek

- 📖 Docs: https://platform.deepseek.com/api-docs/
- 🔑 Console: https://platform.deepseek.com/
- 💰 Preços: https://platform.deepseek.com/pricing

🔗 REPOSITÓRIOS RELACIONADOS

Ecossistema **xiaozhi** com alternativas de implementação:

Servidores Backend:

- **Python**: https://github.com/xinnan-tech/xiaozhi-esp32-server ⭐ 8.2k
- **Java**: https://github.com/joey-zhou/xiaozhi-esp32-server-java
- **Golang**: https://github.com/AnimeAIChat/xiaozhi-server-go

Clientes:

- **Python**: https://github.com/huangjunsen0406/py-xiaozhi (este projeto!)
- **Android**: https://github.com/TOM88812/xiaozhi-android-client
- **Linux**: http://github.com/100askTeam/xiaozhi-linux

Hardware:

- **ESP32**: https://github.com/78/xiaozhi-esp32 ⭐ 11k+ (firmware principal)
- **SF32 Bluetooth**: https://github.com/78/xiaozhi-sf32

🆘 SUPORTE

Problemas com tokens?

1. Verifique se copiou corretamente (sem espaços)

2. Confirme que o token está ativo no console
3. Aguarde 1-2 minutos após criar (pode demorar)

Erro 401?

- Token expirado ou inválido
- Verifique se salvou o arquivo config.json
- Reinicie o teste

Erro de rede?

- Alguns providers podem estar bloqueados em certas regiões
- Considere usar VPN se necessário

Dúvidas?

- Abra uma issue: https://github.com/MarceloClaro/xiaozhi-ai-assistant/issues
- Consulte FAQ: VISION_API_INTEGRACAO.md

☑ PRÓXIMOS PASSOS

1. **Escolha seu provider** (recomendo Aliyun ou Gemini)
2. **Obtenha o token** (siga o guia do provider escolhido)
3. **Configure** config.json
4. **Teste** com `python verify_vision_api.py`
5. **Use** no seu assistente!

Última atualização: 13 de janeiro de 2026
Baseado em: xinnan-tech/xiaozhi-esp32-server v0.8.11

🎉 **Boa sorte com sua Vision API!**