

Virtual Lens Try-on System with Dynamic Price Optimization

Ramesh Tamang, Supran Maharjan*, Shishir Shrestha
Himalaya College of Engineering, Tribhuvan University, Nepal
*Corresponding Author: msupran17@gmail.com

Abstract

This study proposes an integrated solution for online eyewear shopping that combines face shape-based eyewear recommendations with dynamic price optimization to improve customer satisfaction and increase revenue. The first objective employs computer vision techniques, starting with Haar Cascade for face detection and Dlib for facial landmark identification and placement. The system then classifies face shapes using a Swin Transformer model trained on a dataset of diverse facial features. To enhance the virtual try-on experience, 3D rendering is achieved through Markerless Augmented Reality (AR) technology and Blender, allowing customers to visualize eyewear options that best fit their face shape. For the second objective, dynamic price optimization leverages machine learning models such as XGBoost, Random Forest, and Linear Regression to analyze historical sales data and determine optimal pricing strategies. By integrating Price Elasticity of Demand (PED) analysis, the project fine-tunes pricing, adjusts discounts, and maximizes profit margins. This dual approach not only enhances the customer experience but also contributes to the overall profitability of the eyewear company.

Keywords: Computer Vision, Dynamic Price Optimization, Eyewear Recommendations, Face Shape Classification, Personalized Shopping, XGBoost

1. Introduction

In the rapidly evolving digital marketplace, consumers now expect personalized experiences and dynamic pricing, especially in visually sensitive industries like eyewear. However, most online platforms still rely on generic recommendations and static pricing models, creating a gap in user satisfaction and business efficiency.

To bridge this gap, the project titled "Virtual Lens Try-On with Dynamic Price Optimization" introduces a dual system: one for personalized eyewear recommendations based on face shape, [6] and another for real-time price adjustment. The try-on system uses computer vision tools like Haar Cascade and Dlib for facial detection, along with a Swin Transformer model to classify face shapes accurately. This is further enhanced through Markerless Augmented Reality and Blender, allowing users to visualize 3D eyewear models on their faces in real time.

The pricing component utilizes machine learning algorithms such as XGBoost, Random Forest, and Linear Regression to analyze customer behavior, sales trends, and Price Elasticity of Demand. This enables the platform to recommend optimal price points that maximize profitability. Together, the two systems offer a seamless and data-driven shopping experience, benefiting both customers and businesses.

1.1 Problem Statement

Lack of personalized shopping experiences and static pricing models often results in customer dissatisfaction and missed revenue opportunities—especially in the eyewear industry. Shoppers face difficulty in selecting suitable products without the ability to try them on, while businesses struggle to set optimal prices that reflect real-time market conditions and consumer demand.

This project addresses these issues through the integration of virtual try-on technology and dynamic price optimization. The

try-on system enhances customer confidence by allowing real-time visualization of eyewear on their faces, while the pricing model uses machine learning and Price Elasticity of Demand (PED) to adjust prices based on factors like visibility, reviews, and market trends. Together, these technologies create a seamless, personalized, and data-driven shopping experience that benefits both consumers and retailers.

1.2 Objectives

- 1) To classify the face shape and provide pre-selected eyewear options to customers.
- 2) To increase the revenue of the eyewear company through dynamic price optimization.

2. Literature Review

The literature on Virtual Try-On (VTO) paints a clear picture of how technology is reshaping the way people shop, making the process more interactive, personal, and convenient. What began in the late 1990s with simple computer-generated try-on images has now evolved into sophisticated systems powered by augmented reality (AR), virtual reality (VR), and machine learning. Researchers like Professor Parry Bedi were among the first to work on generating realistic digital clothing models, paving the way for future developments. Since then, experts like Professor Sarah Johnson have integrated immersive VR environments into VTO, while Dr. David Lee has improved accuracy through advanced 3D scanning techniques. Their work has helped make VTO experiences more lifelike and reliable. VTO has now become mainstream, with companies like Lenskart and Senco Gold offering virtual try-ons for eyewear and jewelry, and even tech giants like Google joining the trend with AI-based solutions. Studies by researchers such as Shuai Yang, Guiyang Xiong, and Mirela-Catrinel Voicu show that this technology is not just a novelty—it significantly influences how people shop online and how they interact with

brands. Still, there's more to explore. Scholars like Hajer Ghodhbbani and Virginie Leroux have pointed out that while the tech is impressive, we still need to better understand how AR in VTO affects customer engagement and satisfaction. The literature also shows how VTO fits into a larger digital shift across industries, from fashion and tourism to retail and hospitality. While the technology has come far, future research needs to focus more on making VTO more personalized, socially engaging, and accessible to everyone. The goal should be to create virtual try-on experiences that are not only realistic but also inclusive, easy to use, and enjoyable—so that shopping becomes more than a transaction; it becomes an experience.

Dynamic pricing [7] is transforming how businesses operate by allowing them to adjust prices in real time based on factors like demand, customer behavior, competition, and timing. This strategy not only boosts revenue but also helps companies better understand and segment their customers beyond traditional geographic or demographic factors. Industries such as sports, entertainment, hospitality, and transportation have successfully adopted dynamic pricing to optimize sales and manage capacity more effectively. For instance, sports teams and concert organizers use it to fill seats by adjusting prices according to game popularity or performance schedules, while hotels and airlines rely on it to balance occupancy and maximize profit. However, while dynamic pricing offers clear financial benefits, it also raises concerns about fairness—especially when customers realize they paid more than others for the same service. Transparency, communication, and customer trust are essential for its success. If businesses explain pricing changes clearly and offer safeguards, dynamic pricing can improve both customer satisfaction and company performance.

Overview of Dynamic Price Optimization

Dynamic Pricing, also known as demand based pricing, is widely adopted across industries such as retail, e-commerce, and travel. It involves adjusting prices dynamically by analyzing factors like demand fluctuations, competitor strategies, and inventory levels. Traditional models often fall short in capturing non-linear relationships among pricing factors. XGBoost addresses these challenges with its robust performance and scalability [1]

Applications of XGBoost in Pricing Optimization

Retail Sector

In the retail domain, XGBoost has been effectively used to forecast demand and optimize prices. Brown and Smith [5] demonstrated that XGBoost outperforms traditional regression models by capturing complex relationships between sales data and external factors such as seasonal trends and competitor pricing.

E-commerce Platforms

E-commerce platforms extensively use XGBoost to analyze customer purchase patterns, stock levels, and real-time competitor data. Williams and Taylor [8] implemented

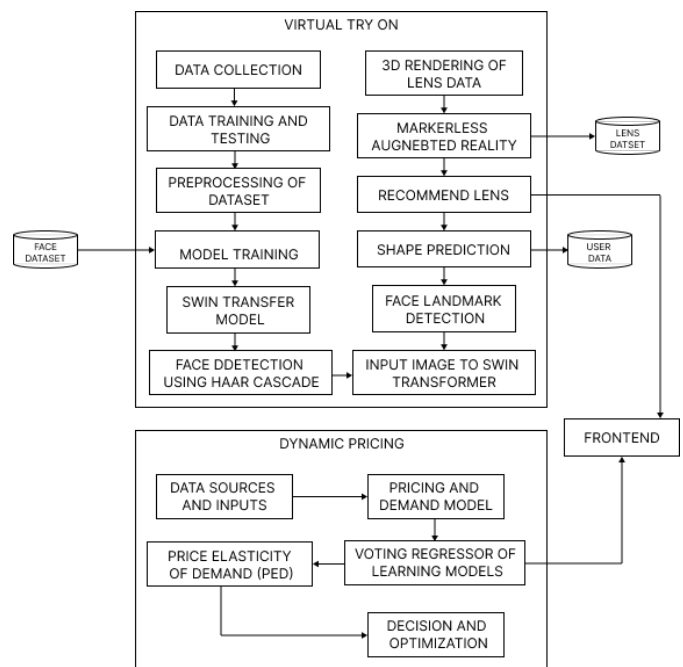


Figure 1. System Architecture

XGBoost for dynamic pricing on an online marketplace, achieving a 15% improvement in revenue through more accurate price predictions.

Travel and Energy Markets

The travel industry has adopted XGBoost for predicting ticket prices and optimizing dynamic fare strategies. Zhao [9] showed that XGBoost reduces forecasting errors significantly compared to Random Forests and Support Vector Machines. Similarly, Zhang and Lee [10] applied XGBoost in energy markets to predict electricity prices, integrating weather and demand data into the pricing models with remarkable accuracy.

3. Methodology

3.1 System Architecture

3.2 Data Preprocessing

1) For Virtual Lens Try On

a) Data Collection

The dataset includes six basic face shapes: heart, diamond, oval, oblong, round, and square. It was sourced from Baran Bingöl and contains 5000 images. Of these, 4000 were used for training (800 per shape) and 1000 for testing (200 per shape).

b) Defining Glass Styles for Each Face Shape

We identified 8 glasses styles—rectangular, oval, and square, round, large, horn, aviator, cat eye.

2) For Dynamic Price Optimization

a) Data Cleaning

- i) Missing Values: Rows with missing data were removed to maintain data quality.

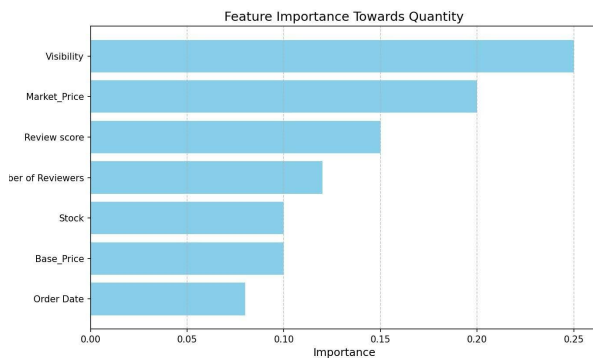


Figure 2. Feature Importance Bar Graph

- ii) **Unnecessary Columns:** Non-essential columns like image URLs, reviews, and dollar-based prices were dropped.
- iii) **Alphanumeric Conversion:** Columns with alphanumeric values (e.g., sock, price) were converted into numeric format for analysis.
- b) **Encoding Categorical Variables**
 - i) **Label Encoding:** Used for gender (e.g., Male = 0, Female = 1, Unisex = 2).
 - ii) **One-Hot Encoding:** Applied to material, lens type, and color to create binary features.
 - iii) **Frequency Encoding:** Replaced categorical values with their frequency counts to support model inputs.
- c) **Feature Selection**
 - i) **Random Forest Model:** Built to rank feature importance.
 - ii) **Selected Key Features:**
 - A) Number of Reviews
 - B) Stock Level
 - C) Pieces Sold
 - D) Market Price
 - E) Base Price
 - F) Visibility
 - G) Review Score
 - H) Order Date

These features were most influential in predicting eyewear prices and ensured model reliability.

- d) **Feature Importance Results**
Top Features by Importance Score:
 - i) Visibility: 0.25
 - ii) Market Price: 0.20
 - iii) Review Score: 0.15
 - iv) No. of Reviews: 0.12
 - v) Stock: 0.10
 - vi) Base Price: 0.10
 - vii) Order Date: 0.08

Visibility and Market Price were most impactful, followed by customer reviews, availability, and seasonal patterns.

- e) **Data Splitting**

The preprocessed dataset was split into training and testing sets using an 80-20 ratio. The training set was used to train the models while the testing set served as an unseen dataset to evaluate the models' performance. The split ensured that the evaluation metrics accurately reflected real world performance.

4. Model Training

4.1 For Virtual Lens Try On

- 1) **Face Detection using the HAAR cascade**
HAAR Cascade, [11] a real-time object detection technique introduced by Viola and Jones, was used for detecting facial regions. It applies rectangular Haar-like features and a cascade classifier trained with AdaBoost to scan image frames effectively. OpenCV's pre-trained models enabled reliable face localization.
- 2) **Swin Transformer Model**
A Swin Transformer model was used to classify face shapes based on facial landmarks. The model processes non-overlapping image patches using shifted window-based self-attention, offering strong local and global feature representation. Input images were resized, embedded into high-dimensional vectors, and passed through multiple Swin blocks with multi-head self-attention to predict face shape categories.
- 3) **Face Landmark Detection using dlib**
Using dlib's [3] 68-point facial landmark detector, features such as eyes, nose, jawline, and mouth were extracted. These landmarks helped preprocess face images and accurately align eyewear in the AR pipeline.
- 4) **Markerless Augmented Reality**
Planar object detection was performed using canny edge detection and ORB-based key point matching. Homography and camera matrices were computed to map 2D video input into a virtual 3D scene, allowing accurate overlay of 3D eyewear models.
- 5) **3D Rendering of Lens Data**
Eyewear models were aligned to facial landmarks and rendered in real time using Blender. Texture mapping, lighting effects, and live camera feed integration allowed users to view eyewear virtually and interactively. Users could adjust position and orientation through gesture or touch input.
- 6) **Performance Evaluation**
The model's performance was evaluated using a confusion matrix along with key classification metrics such as accuracy, precision, recall, and F1-score to assess overall and class-wise effectiveness. For regression-based evaluation, metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) were employed to quantify prediction errors. MAE measured the average magnitude of prediction errors, MSE captured the average squared differences, and RMSE provided error estimates in the same unit as the target variable. Additionally, the coefficient of determination (R^2) was used to evaluate the model's ability to explain variance in the data.

4.2 For Dynamic Price Optimization

To predict future product demand—a continuous variable—various models were explored, with linear regression chosen as the baseline due to its simplicity and interpretability. This model was appropriate since the data satisfied key assumptions: linearity between variables, feature independence, and approximate normal distribution. Hyper parameters such as learning rate (0.01) and maximum iterations (1000) were optimized using GridSearchCV with 5-fold cross-validation. Elastic Net regularization, which combines L1 and L2 penalties, was employed to prevent overfitting. The final linear model demonstrated the relationship between demand and features such as Product ID, Market Price, Date, and seasonal factor including regression-based models.

In addition to linear models, tree-based approaches like Random Forest and XGBoost were used to improve prediction accuracy. Random Forest utilizes multiple decision trees built on bootstrap samples, averaging their outputs to reduce variance and resist overfitting. XGBoost, a gradient boosting algorithm, sequentially fits trees to minimize loss while incorporating regularization to control complexity. These models handled nonlinear interactions and did not require predefined assumptions about feature relationships. Hyper parameters such as tree depth and number of estimators were fine-tuned, and performance was monitored using gradients and regularized loss functions.

To further enhance prediction robustness, a Multilayer Perceptron (MLP) Regressor and an ensemble [2] approach were implemented. The MLP, a feedforward neural network with one hidden layer (100 neurons) and ReLU activation, used the Adam optimizer and learned by minimizing MSE through backpropagation. Finally, a Voting Regressor ensemble combined predictions from Linear Regression, Random Forest, and XGBoost by averaging their outputs. This ensemble method effectively reduced overfitting and leveraged the strengths of each individual model to improve overall prediction accuracy and generalization.

Price Elasticity of Demand

To analyze the impact of price changes on product demand, the concept of Price Elasticity of Demand (Ed) was applied. It measures how the quantity demanded responds to changes in price. In a business context, base demand is first predicted using a trained model. To maintain pricing stability, a discount guardrail ($+\delta\%$) is enforced, limiting how much the base discount can change. Price elasticity values are updated weekly to reflect current market behavior. Demand is then recalculated at three points: base discount, increased discount ($+\delta\%$), and decreased discount ($-\delta\%$), giving a range of demand responses for different pricing strategies.

To determine the most profitable pricing option, a linear programming approach is used. The objective is to maximize total revenue across all products. For each product, the price point yielding the highest expected revenue is selected from the three demand scenarios. This approach ensures optimized pricing decisions while respecting practical constraints, such as minimum and maximum price limits defined by the busi-

ness.

4.3 Hardware and Software environment

The experiments were conducted on a workstation equipped with an NVIDIA RTX 3060 GPU (12 GB VRAM), an Intel Core i7 processor, and 32 GB RAM, running on Windows 11 (64-bit). GPU acceleration was essential for real-time face landmark detection, 3D mesh rendering, and virtual lens overlay, enabling smooth performance during live try-on interactions. The software stack included Python 3.10, OpenCV 4.8, MediaPipe Face Mesh, NumPy, Pandas, Matplotlib, and Scikit-Learn for data preprocessing, visualization, and model training. For advanced modeling, XGBoost, LightGBM, and TensorFlow were integrated to support both classical and neural network-based demand prediction. The backend for the pricing engine was executed in a Node.js (v18+) environment with Express.js, while the database operations were handled through MongoDB.

5. Results and Discussion

For the Virtual Lens Try-On system, the training process of the face shape classification model was analyzed through accuracy and loss curves plotted over multiple epochs. The accuracy graph reflected how well the model improved its predictions over time, revealing learning patterns and potential overfitting or convergence. Simultaneously, the loss curve—representing prediction error—showed a general decreasing trend across epochs, indicating that the model was progressively minimizing its error and learning from the data effectively. Together, these visualizations offered insights into the model's performance dynamics during training.

The classification results of face shape determination using the Swin Transformer model in conjunction with the HAAR cascade classifier are presented along with their corresponding probabilities. The integration of the Swin Transformer, a hierarchical vision transformer architecture, with the HAAR cascade—used for efficient face detection—demonstrates a robust and accurate approach to face shape classification. The output probabilities reflect the model's confidence in each predicted face shape category, providing insights into the reliability and precision of the classification system.

The model's classification ability was further assessed using a confusion matrix, which provided a detailed breakdown of correct and incorrect predictions for each face shape class. Diagonal elements indicated true positives—correct classifications—while off-diagonal elements captured misclassifications. This matrix enabled a granular understanding of the model's strengths and weaknesses. Precision, recall, and F1-score were also calculated for each class. The "Square" face shape had the best performance (F1-score: 0.79), while "Oval" showed the weakest (F1-score: 0.56). The overall accuracy of the model stood at 70%, demonstrating reasonable performance in identifying facial shapes for virtual try-on applications.

In parallel, a dynamic price optimization (Kirsch, 2023) model was implemented to maximize revenue based on predicted product demand. Price Elasticity of Demand (PED) was

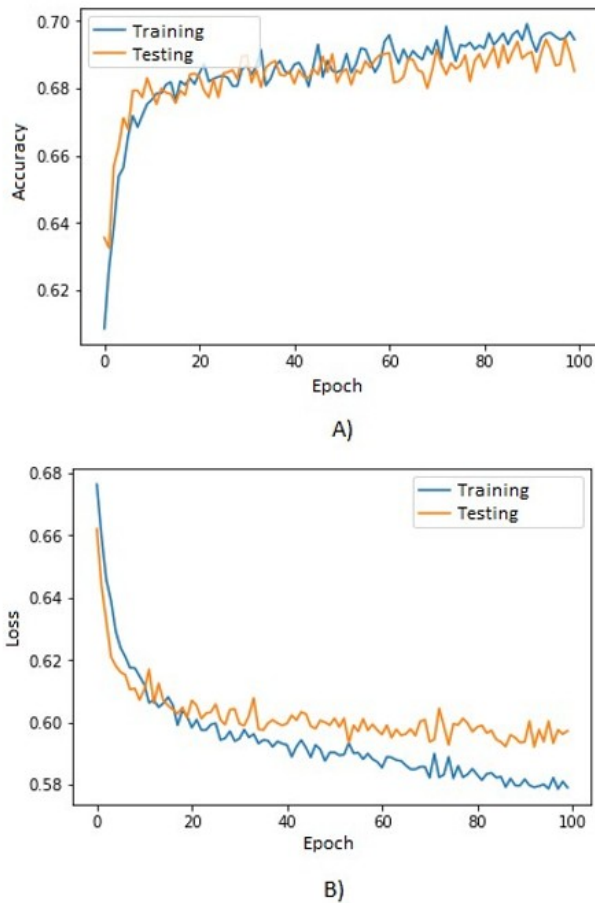


Figure 3. Accuracy and Loss Function

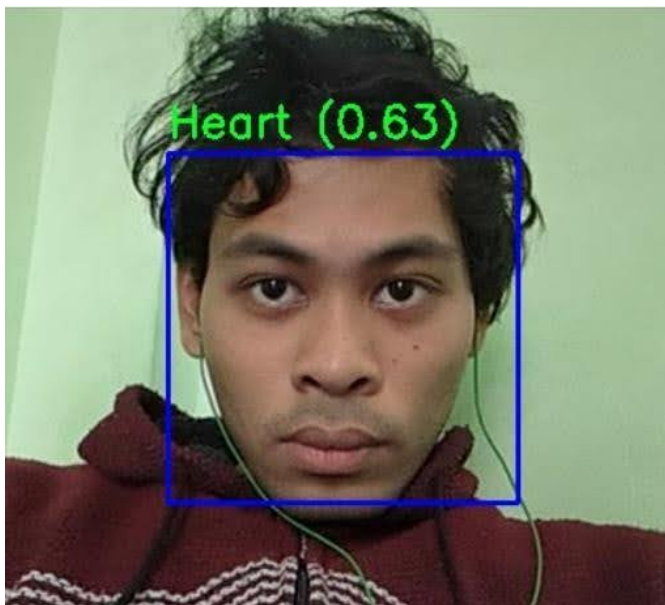


Figure 4. Output Showing Face Shape

applied to estimate changes in quantity demanded under different pricing scenarios (base, $(+\delta\%)$, and $(-\delta\%)$). Demand predictions were generated using multiple machine learning

	Precision	Recall	F1-score	Support
Heart	0.61	0.69	0.65	200
Oblong	0.71	0.80	0.75	200
Oval	0.69	0.47	0.56	200
Round	0.69	0.74	0.71	200
Square	0.79	0.79	0.79	200
Accuracy	0.70			1000

Figure 5. Confusion Matrix Showing the Matched Input Image With the Pre-Trained Model

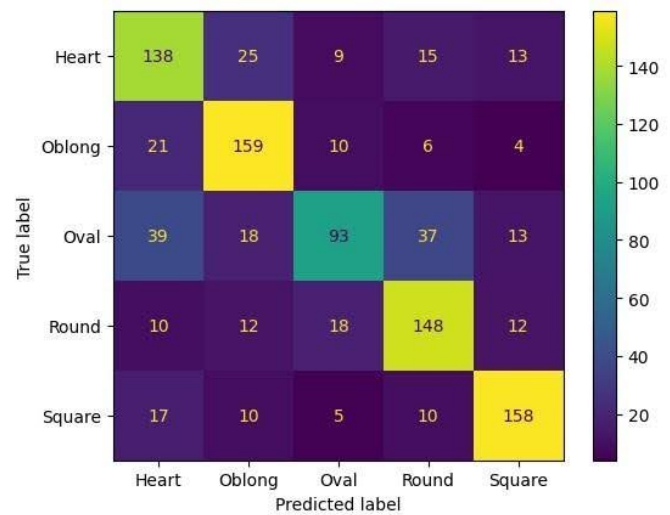


Figure 6. Classification Report

	Precision	Recall	F1-score	Support
Heart	0.61	0.69	0.65	200
Oblong	0.71	0.80	0.75	200
Oval	0.69	0.47	0.56	200
Round	0.69	0.74	0.71	200
Square	0.79	0.79	0.79	200
Accuracy	0.70			1000

Figure 7. Models Performance Table

regressors, among which Random Forest ($MSE = 4.542$, $R^2 = 0.9668$) and XGBoost ($MSE = 4.844$, $R^2 = 0.9646$) demonstrated the highest predictive accuracy, validating the robustness of the optimization pipeline. Linear programming was then employed to select the optimal price that maximized expected revenue for each product. Statistical comparison between pre-optimization and post-optimization pricing for Round Metal Glasses revealed a clear improvement in revenue trends.

As illustrated in Table 3, the average daily revenue increased from NPR 29,600 to NPR 135,200, supported by significant reductions in revenue fluctuations and an overall smoother upward trajectory. This empirical evidence demonstrates that dynamic price optimization, when combined with predictive modelling, substantially enhances revenue performance in a real-world retail environment.

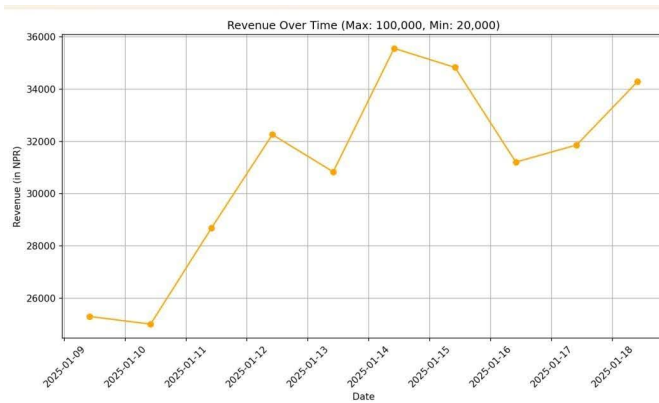


Figure 8. Before Price Optimization for Round Metal Sunglasses

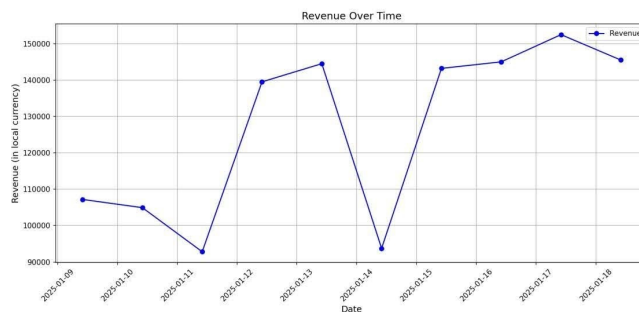


Figure 9. After Price Optimization for Round Metal Sunglasses

Metric	Before Optimization (Orange Line)	After Optimization (Blue Line)
Average Revenue	NPR 29,600	NPR 135,200
Minimum Revenue	25,000	92,000
Maximum Revenue	36,000	152,000
Revenue Variance	High	Moderate
Trend Pattern	Irregular fluctuations	Stable upward trend

Figure 10. Revenue Comparison Before and After Dynamic Price Optimization

6. Conclusion

In conclusion, this research successfully addresses the challenges of personalizing the online eyewear shopping experience and optimizing pricing strategies. By utilizing advanced computer vision techniques, the research provides customers with eyewear options suited to their individual face shapes, ensuring better fit and aesthetic appeal. Simultaneously, dynamic price optimization using machine learning models and PED analysis ensures that the eyewear company can adjust prices effectively, maximizing profitability while meeting market demand. The validation with the competitor price gave effective revenue increase of 20 – 30%. This approach results in a more personalized and engaging shopping experience, which not only enhances customer satisfaction but also drives business growth.

The virtual try-on module struggles with accuracy under poor lighting, extreme head angles, or partial facial occlusions, reducing the precision of lens alignment. The 3D

overlay remains simplified, limiting realism across diverse facial structures, and performance depends heavily on GPU acceleration, causing slowdowns on low-end devices. On the pricing side, the dynamic optimization model is constrained by a limited dataset and simplified assumptions about price elasticity, without accounting for cross-product effects or long-term trends.

Future research could focus on integrating advanced AI-driven features, such as virtual try-on capabilities and more sophisticated price prediction models, to further enhance the system's functionality. Additionally, expanding the personalization aspect by incorporating user preferences and purchase history would create a more tailored shopping journey. As AI and machine learning continue to evolve, the ability to deliver real-time price adjustments and more accurate face shape analysis will further improve the overall customer experience and operational efficiency, ensuring sustainable business success.

7. Acknowledgement

We express our sincere gratitude to Himalaya College of Engineering for providing us with the platform to carry out this research.

References

- [1] Guestrin, T. C. (2016). *XGBoost: A scalable tree boosting system*. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794).
- [2] Kazemi, V., & Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 1867–1874).
- [3] King, D. E. (2009). dlib-ML: A machine learning toolkit. *Journal of Machine Learning Research*, 1755–1758.
- [4] Kirsch, K. (2023, March 2). *Price optimization*. Retrieved from <https://blog.hubspot.com/sales/price-optimization>
- [5] Lee, J. Z. (2021). Energy price forecasting with XGBoost: Applications in dynamic pricing. *IEEE Transactions on Energy Markets*, 7(2), 321–333.
- [6] Patel, R., & Hribar, V. (1997). Recommender systems. *Communications of the ACM*, 56–58.
- [7] Saranagav. (2024, January 26). *Dynamic pricing optimization in e-commerce using predictive analytics*.
- [8] Smith, A. B. (Aug. 2021). Dynamic pricing optimization using gradient boosting in retail. *Journal of Retail Analytics*, 14(3), 34–42.
- [9] Taylor, J. W. (2020). Predictive analytics in e-commerce using machine learning. *IEEE Access*, 8, 14467–14478.
- [10] Tutorialspoint. (2023). Retrieved from https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm
- [11] Viola, P., & Jones, M. J. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 511–518). IEEE.
- [12] Zhao, Y. (2022). Optimizing airline ticket prices with machine learning: A case for XGBoost. *Journal of Transportation Research*, 12(4), 210–217.